Lexical Popularity: Quantifying the Impact of Pre-training for LLM Performance

Elena Sofia Ruzzetti[1], **Fabio Massimo Zanzotto**[1,2], **Tommaso Caselli**[3]
[1]Human Centric ART, University of Rome Tor Vergata, Italy
[2]Almawave S.p.A., Rome, Italy
[3]CLCG, University of Groningen, The Netherlands
elena.sofia.ruzzetti@uniroma2.it
fabio.massimo.zanzotto@uniroma2.it
t.caselli@rug.nl

Abstract

Large Language Models (LLMs) excel in numerous and varied tasks. Yet, the mechanisms that underlie this success remain insufficiently understood. In particular, the size and the limited transparency of their pre-training materials make it difficult to state what the properties of the pre-training material are when compared to the test data. In this paper, we investigate whether LLMs learned generalized linguistic abstraction or rely on surface-level features, like lexical patterns, that match their pre-training data. We explore this by examining the relationship between lexical overlap of test data and task performance. We observe that lexical overlap with the pre-training material is mostly beneficial to model performance on tasks requiring functional linguistic knowledge. To further explore the impact of lexical features, we also demonstrate that LLMs are fragile with respect to lexical perturbations that preserve semantics. While we expected models to rely on lexical overlap between test instances and pre-training data for tasks requiring functional knowledge, lexical perturbations reveal that models also exhibit, to a lesser extent, this dependence for tasks requiring formal linguistic knowledge.

1 Introduction

Large Language Models (LLMs) have achieved impressive results across a wide range of linguistic and reasoning tasks. Kaplan et al. (2020) have postulated Scaling Laws of LLMs showing that their performance is a direct effect of scaling up the model's size, the pre-training material, and the compute power. As a direct consequence, we have witnessed the development of increasingly large models in terms of parameters (`Deepseek-3.1` is the largest open-weight LLMs with 670B parameters) and use of pre-training materials which now is in the range of trillions of tokens. A well-known behavior of machine learning models is their dependence on the (pre-)training material: the more this is varied and consistent with the task at hand, the more likely the learning algorithm will be able to *generalize on unseen (test) data*. For this to be observable, a strict separation between the material seen during the learning phase and that seen during test or deployment is essential. Current LLMs' development has put this assumption in jeopardy. On the one hand, the increasingly larger size of data used to develop these models makes it very difficult to properly document its content and determine the statistical properties of the training data. On the other hand, the vast majority of model developers do not release, nor properly document, the pre-training materials. This opacity undermines our ability to assess whether model performance reflects genuine generalization, or even to define generalization as the capability to perform well on unseen test instances.

While previous work has primarily focused on the issue of data contamination (Ranaldi et al., 2023; Sainz et al., 2023; Golchin and Surdeanu, 2023; Deng et al., 2024; Ranaldi et al., 2024), it remains unclear how the *data distribution* in the pre-training material informs LLM behavior. Clarifying this relationship is essential for a deeper understanding of LLM functionalities, especially for disentangling surface-level pattern matching of pre-training data from genuine generalization. For some languages, for example, syntactic perturbations that preserve semantics are demonstrated to impact performance negatively (Ruzzetti et al., 2024). In this paper, we estimate the impact of the pre-training material by assessing LLM performance on a battery of benchmarks requiring either *formal* or *functional* linguistic knowledge (Mahowald et al., 2024). In particular, our analysis investigates the extent to which lexical distributions in pre-training data influence model performance across these types of tasks. To assess the impact of the lexical distribution of the pre-training material, we introduce a lexical invariance test (Ribeiro

Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics
Volume 1: Long Papers, pages 1209–1230
March 24-29, 2026 ©2026 Association for Computational Linguistics

et al., 2020), which compares model performance on original test instances and semantically equivalent, lexically perturbed counterparts. Our working hypothesis is that if LLMs have learned linguistic generalization functionalities during their pre-training, they should exhibit minimal, if not null, performance degradation under lexical perturbation. We demonstrate that LLMs are fragile to these perturbations and that the lexical mismatch between pre-training data and test instances is influencing performance. We also show that current measures of memorization of surface-level lexical patterns like Training Data Extraction (Carlini et al., 2021, 2023; Nasr et al., 2023) and Membership Inference Attacks (Mireshghallah et al., 2022; Mattern et al., 2023), while being effective in quantifying privacy risks and copyright infringement, cannot be used to capture model reliance on lexical features.

Our contributions can be summarized as follows:

- We test LLMs performance against a notion of *n-gram popularity* using models' pre-training data (Section 4).
- We show through an invariance test that LLMs over-rely on surface-level, lexical features observed during the pre-training phase, especially when solving tasks that require *functional* linguistic knowledge (Section 5).
- We show that current definitions and measures of memorization fail to account for LLM performance across tasks requiring both formal and functional linguistic knowledge (Section 6).

2 Related Work

A large number of works have focused on understanding *which capabilities* LLMs acquire from their pre-training data (Bender et al., 2021; Mitchell and Krakauer, 2023; van Dijk et al., 2023) and *how* this data influences their learning.

Mahowald et al. (2024) have shown that LLM capabilities are not uniformly distributed across types of linguistic knowledge. Specifically, LLMs excel on benchmarks targeting knowledge of linguistic rules and patterns, i.e., *formal* competence, but continue to struggle on tasks requiring semantic and pragmatic reasoning, i.e., *functional* competence, where satisfactory performance typically demands fine-tuning. Circuit-level analyses further suggest that these two forms of knowledge are localized in distinct regions of LLMs (Hanna et al., 2025). Building on this distinction, we aim to investigate how the pre-training data impact *formal* and *functional* competence.

Understanding how training data influences the predictions of LLMs remains an open question. A prevailing hypothesis posits that LLMs rely heavily on memorization of pre-training data, with some studies suggesting that such memorization is fundamental to model performance (Feldman, 2020). There are initial studies proposing architectures for building LLMs based on memorizing before performing backpropagation (Zanzotto et al., 2025). However, the definition and scope of memorization in LLMs are still debated within the NLP community. In privacy and copyright contexts, memorization is often operationalized as the verbatim reproduction of training examples, typically measured through Training Data Extraction (TDE) methods (Carlini et al., 2021, 2023; Satvaty et al., 2025; Lasy et al., 2025). Complementary approaches, such as Membership Inference Attacks (MIA), attempt to determine whether a given document was included in the training corpus (Shi et al., 2024; Mireshghallah et al., 2022; Mattern et al., 2023; Duan et al., 2023). Although high extraction success rates raise significant privacy concerns (Nasr et al., 2023), they do not fully account for the breadth of LLM-generated outputs (Duan et al., 2024). In this paper, we further investigate whether the current definitions of memorization explain model performance.

The tension between memorization and generalization has gained increasing attention (Hupkes et al., 2023; Chang and Bergen, 2024; Wei et al., 2025). The influence of pre-training data has been explored during the generation of suffixes given a prefix (McCoy et al., 2023; Merrill et al., 2024), and recent findings suggest that texts generated by LLMs are less novel than those written by humans. To understand how LLMs solve tasks, a positive impact of hight similarity of in-context demonstrations to the pre-training has been discussed by previous work (Chan et al., 2022; Razeghi et al., 2023; Chen et al., 2024). Moreover, recent work demonstrates that knowledge-intensive tasks depend on high similarity of test and pre-training distribution (Hartmann et al., 2023; Wang et al., 2025). In this paper, we will focus on lexical distribution of test and pre-training data for testing LLMs' generalization ability on *functional* and *formal* tasks: in particular, we will focus on understanding how models generalize when a shift in the test data can

be found with respect to their pre-training material (Liang et al., 2023; Srivastava et al., 2023).

3 Task and Models

Our goal is to assess whether and to what extent there is a relationship between LLM performance (on selected benchmarks) and lexical overlap with their pre-training material. We also assume that, if such a relationship is observable, it may be influenced by the linguistic knowledge required to solve the task, and, potentially, by the model architecture and training regime.

Concerning the types of linguistic knowledge, it is known that LLMs show more robust performances on benchmarks designed to test formal linguistic knowledge when compared to those requiring functional one (Mahowald et al., 2024). While the factors controlling this separation remain to be understood, we aim to understand the effects of pre-training lexical distributions on tasks of both types. For this reason, we have selected benchmarks that can, separately, measure *formal* and *functional* linguistic knowledge of an LLM. We have hence selected eight tasks, equally divided for the two types of linguistic knowledge, from six benchmarks. For each task, we have also controlled that: (*a.*) the task can be run in a zero-shot setting since we aim to investigate the effect of pre-training only; and (*b.*) it has a suitable number of instances to investigate the presence of lexical overlap in the pre-training material of the LLMs.

We also want to test whether the size and pre-training regime of an LLM influence the relationship between model performance and lexical overlap with the pre-training. To control for these variables, we have explicitly selected full open-source LLMs where both their weights and their pre-training data are publicly available and accessible. Considering the constraints in terms of availability of benchmarks, access to pre-training materials, and LLMs of different sizes, we selected English as our test language.

Formal Benchmarks To test LLMs' *formal* linguistic competence, we rely on BLiMP (Warstadt et al., 2020), a benchmark composed of minimal pairs that contrast for their acceptability. Among the 12 tasks in the benchmark, we selected three that require purely syntactic and morphological linguistic knowledge (Determiner-Noun agreement, Anaphor Agreement, and Subject-Verb agreement), and one at the syntactic-semantic interface (Argument Structure). In all the tasks, a model should assign a higher probability to the grammatical option. In Determiner-Noun agreement, determiners need to agree in number with their associated noun; for the Anaphor Agreement the requirement is that reflexive pronouns agree with their antecedents. For the Subject-Verb agreement the subject and a present-tense verb agree in number In the Argument Structure tasks, different argument configurations are checked with the verb in the sentence (e.g. whether a verb can take a direct object, undergo alternations, or take inanimate arguments).

Functional Benchmarks For the evaluation of the *functional* competences of LLMs, we selected four benchmarks: LAMBADA, PIQA, COPA and WinoGrande. LAMBADA (Paperno et al., 2016) is used to test how accurately language models (LMs) can generate text through a word prediction challenge. It is composed of narrative passages, and the model is tested to generate the final word: to properly solve the task, an LM must rely on the entire context rather than on potential shortcuts (e.g., use only the last sentence). PIQA (Bisk et al., 2020) is a question answering task that requires world knowledge to correctly answer a question: given a goal, a model must choose the correct completion based on the physical properties of the objects in the sentence. COPA (Roemmele et al., 2011) is intended to test commonsense reasoning: given a premise and two alternatives, the model should choose between the two alternatives the one that is more plausible to have a causal relation with the premise. WinoGrande (Sakaguchi et al., 2020) requires a model to perform pronoun resolution and common sense reasoning. When compared to the Anaphora Resolution benchmark in BLiMP, WinoGrande emphasises access to world knowledge and discourse regularities to correctly resolve the anaphors.

Models and Pre-Training Data We selected four foundational open-source models comparable in size from two different families: `OLMo 1B` and `OLMo 7B` (Groeneveld et al., 2024); `Pythia 2.8B` and `Pythia 6.9B` (Biderman et al., 2023).

The OLMo models are trained on Dolma (Soldaini et al., 2024) and the Pythia family on the Pile (Gao et al., 2020). Both Dolma and the Pile are large-scale datasets (11 TB and 825GB of data, respectively) characterized by a diverse mixture of web-crawled data. The scale of those datasets make them challenging to explore: for this reason, in our

experiments, we employed the inverse indexes – and the corresponding API – made available by Liu et al. (2024). The usage of those indexes allows for the computation of the number of documents containing a given sequence of tokens, and enables a precise quantification of the presence of the target sequence in the pre-training data of the LLMs.

4 The Role of Training Data Distribution in LLM Behavior

Generalization capabilities of a machine learning model is traditionally discussed as the ability of a model to learn from a set of training data and to obtain good performance on *unseen* test data (from the same distribution as the training data). However, this definition does not take into account how strong the similarity between training data and test instances has to be (Ramponi and Plank, 2020).

We are interested in understanding how typical a test instance is compared to the pre-training distribution, and whether this similarity can influence performance. To clarify this relationship, we study the effects of *lexical similarity* of a test instance with respect to the pre-training data, expressed as the overlap of its n-grams with the pre-training material. In particular, we hypothesize that *the more often an LLM is exposed to a sequence of tokens, the more likely it is* ***to recall and reuse*** *it when solving downstream tasks*. Thus, estimating n-gram frequency in pre-training data can offer insights and potentially predict model performance on a given benchmark.

Instance Popularity Score To operationalize this, we introduce the *Instance Popularity Score* (IPS), a metric to assess how popular (i.e, common) a test instance is, based on the cumulative frequency of its constituent n-grams in the pre-training data of a given LLM.

To quantify the popularity of a test instance $x = [x_1, ..., x_L]$ of L tokens, we extract its set of n-grams G_x^n, and, for each $g \in G_x^n$, we record its number of occurrences in the pre-training corpus, denoted $count(g)$. We used the available index over the Pile and Dolma and associated API from Liu et al. (2024) to retrieve this information.

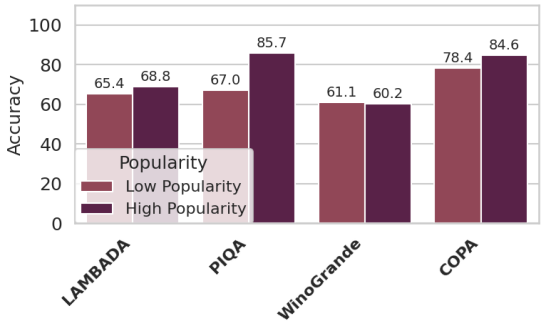
We then estimate the popularity of each g in the pre-training data by situating $count(g)$ within the overall distribution of occurrence counts. Specifically, we compute the deciles of the count distribution across the test set and assign g a popularity score $popularity(g) \in [1, 10]$, corresponding to the decile (bin) in which its count falls. Finally, we define the IPS of the test instance x as the average popularity of its constituent n-grams:

$$IPS(x) = \frac{1}{|G_x^n|} \sum_{g \in G_x^n} popularity(g)$$

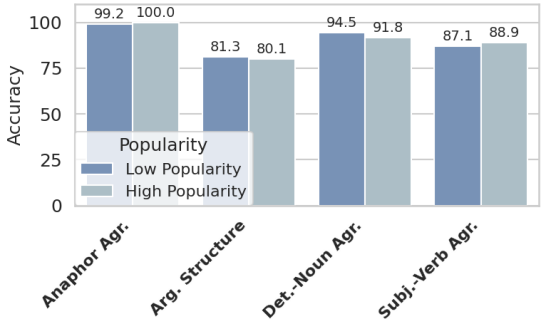
In all tasks that require an LLM to choose between two options, we compare the IPS of the correct option with the IPS of the wrong option: in particular, we measure:

$$\frac{IPS(x_{correct}) - IPS(x_{wrong})}{IPS(x_{correct})}$$

that quantifies how much the correct option is more popular than the wrong one. For LAMBADA, instead, since it requires an LLM to generate a specific word, we considered only the IPS of the entire test instance. Following previous work (Brown et al., 2020; Merrill et al., 2024; Wang et al., 2025), we experimented with n-grams of 7 tokens.



(a) *Functional* tasks



(b) *Formal* tasks

Figure 1: Accuracy of `Pythia 6.9B` for *functional* and *formal* tasks, computed in bins based on *popularity* of test instances.

Popularity mostly impacts *functional* tasks Grouping the test instances into two macro-categories, i.e., those with *low* IPS and those with *high* IPS (based on the median popularity), we observe that on the *high* IPS test instances, all LLMs

tend to obtain higher results, with a prominence for the *functional* tasks. Figure 1 illustrates the impact of the popularity of test instances over model performance for `Pythia-6.9B`. The remaining LLMs are reported in Figure A in Appendix A.1.

This behavior suggests that lexical overlap between the pre-training materials and test sets is indeed beneficial to model performance, with differences in performance of up to 20% between the two groups on the PIQA dataset. However, the marginal performance gains of the highly popular test instances in the *formal* tasks highlight an asymmetry which confirms that LLMs rely on different features to solve these tasks. In particular, it appears that the structural interpretation of a sentence, necessary for *formal* tasks, is less affected by frequent exposure to certain lexical patterns.

As for WinoGrande, we do not observe this trend, with performance remaining similar between the two groups in line with the *formal* benchmarks. This behavior can be explained by considering the role of the pronoun anaphora resolution subtask, which expresses *formal* linguistic knowledge.

5 Shifting Lexical Distribution Affects LLMs' Performance

To further validate the role that lexical similarity between pre-training and test instances has on model performance, we have designed an **invariance test** inspired by CHECKLIST approach (Ribeiro et al., 2020). In particular, we have applied label-preserving perturbations with the expectation that the model's predictions must be the same.

We focused on lexical perturbations that do not alter the semantics of a sentence, but rather the statistical properties of the test instance. Indeed, a model that does not rely on surface-level pattern matching with its pre-training data and has learned to generalize should be marginally affected by this kind of perturbation.

Perturbation Strategy Perturbations are obtained by substituting the nouns and verbs in a selected benchmark test instance with corresponding synonyms. To do this extensively, eligible synonyms have been retrieved from WordNet (Miller, 1995), a lexical database where nouns, verbs (and adjectives) with the same meaning are grouped into *synsets*. Every synset identifies a concept, and every lexical item belonging to the same synset shares the same meaning. Lexical items belonging to the same synset can substitute for one another without altering the overall meaning of a sentence, as they express the same underlying concept (Lyons, 1968). Although synonyms maintain the syntactic and semantic structure of benchmark sentences, their usage frequencies may vary (Palmer, 1981). This variability is what allows us to evaluate the influence of the lexical distribution of the test instance with respect to pre-training material. If a model has truly learned semantic representations, rather than relying on surface-level lexical features, it should produce consistent predictions when synonyms are used, even in the case of less frequent ones. Details of the perturbation pipeline are in Appendix A.2.

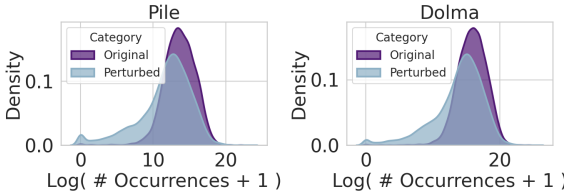


Figure 2: Distribution of the occurrences of original words and their synonyms (with a logarithmic scale) on Pile and Dolma.

Figure 2 shows the frequencies of the original words and their synonyms for the Pile and Dolma. As expected, the distribution of the synonyms presents a larger variance, and it is more skewed towards zero: overall, the synonyms are less frequent in the pre-training datasets than the original words.

Since the perturbations should influence only the lexical properties of the sentences without modifying their semantics, we also measure the semantic similarity of the original input with respect to the perturbed input. We quantify the similarity with BERTscore (Zhang et al., 2020) and ROUGE-L: high BERT scores testify that there is a high semantic similarity, while lower ROUGE-L scores are expected, due to the lexical perturbations. As illustrated in Figure 3, the proposed approach to generate semantically equivalent perturbed instances is valid: a high semantic similarity is kept in both *formal* and *functional* datasets allowing us to test LLMs both the original and the perturbed input for their generalization abilities. We also further validate our perturbation strategy and we implemented an LLM-as-judge annotation of a subset of the dataset. We prompted Llama 3.1 8B on 50 examples from each subtask to judge whether the original and the perturbed sentence convey the same meaning. Details of the evaluation are pro-

vided in Appendix A.3. Across the different tasks, the model evaluates the original and perturbed sentences positively in 94.67%, i.e., they convey the same meaning. This further confirms the validity of our implementation of exclusively lexical perturbation.

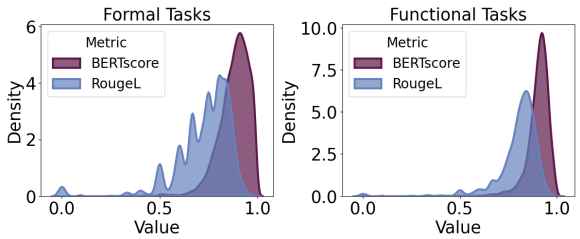


Figure 3: Similarity scores distributions of the perturbed sentences and the original ones, in functional and formal tasks. The high BERT scores testify that there is a high semantic similarity, while lower ROUGE-L scores are expected due to the lexical perturbations.

LLMs over-rely on lexical features especially for *functional* tasks Results in Table 1 and 2 compare the Original to the *Lexically Perturbed* versions of the datasets. We discuss performance in terms of accuracy on all tasks. We also report the perplexity for LAMBADA, which is frequently used as a task-specific performance metric. To quantify the effect of the perturbation, we also report the Δ between the Original results and the *Lexically Perturbed* ones, rescaled on the Original to obtain a percentage change. In Appendix A.4, we also discuss noun-only and verb-only perturbations.

The results indicate that LLMs have a drop in performance across all tasks, notably for the *functional* tasks. Indeed, on the LAMBADA, PIQA and COPA tasks (Table 1) models tend to lose between the 16.25% and the 27.5% of accuracy on *Lexically Perturbed* data, and the perplexity on LAMBADA even quadruples. `OLMo 1B` exhibits substantial accuracy degradation under perturbation, with drops of up to 26.17% on LAMBADA, accompanied by a correspondingly high perplexity on the same benchmark. While `OLMo 7B` consistently outperforms `OLMo 1B` across all Original benchmark versions, it follows a comparable trend in relative performance decline. Nonetheless, under perturbation, `OLMo 7B` demonstrates greater robustness, sustaining higher overall scores. `Pythia 2.8B` exhibits moderate performance relative to the other models: its accuracy and robustness are generally lower than those of `OLMo 7B`, yet higher than `OLMo 1B`. Although `Pythia 6.9B` achieves higher absolute accuracy than `Pythia 2.8B`, it experiences a more pronounced degradation in robustness across all tasks. Notably, the increased model size does not translate into improved stability, as the accuracy drops observed on LAMBADA, PIQA, and COPA are larger than those recorded for `Pythia 2.8B` on the corresponding benchmarks.

The performance drops on *functional* tasks does not equally affect WinoGrande: this is in line with the observed behavior in Section 4, and can be explained by the additional *formal* linguistic knowledge necessary to solve it.

As for the *formal* tasks the degradation of performance is less severe (Table 2), ranging between 6.95% and 15.64%. The magnitude of the loss in performance is smaller than on *functional* tasks, but not minimal (drops are statistically significant with $p < 0.05$). `OLMo 1B` performs reasonably well on Original tasks, and demonstrates limited vulnerability to perturbations. The larger model in the same family, `OLMo 7B`, has a similar performance across all tasks. For `Pythia 2.8B`, the accuracy peaks in the Anaphor Agreement, and despite its small size, it is slightly more robust than `OLMo 7B` on all tasks. `Pythia 6.9B` also obtains high scores on the Original, and has a comparable drop in performance across all tasks.

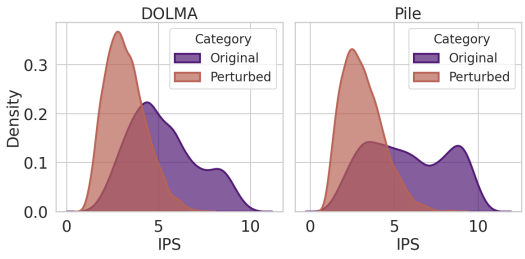


Figure 4: Distribution of the *popularity* scores of the test instances of LAMBADA on Dolma and Pile.

The lexical perturbation hurt the popularity of the test instance Lexical perturbations not only lead to systematic performance degradation on *functional* and *formal* benchmarks, but they also substantially decrease the popularity of the corresponding test instances. Figure 4 visualizes these differences for the LAMBADA benchmark.

We further analyze the subset of instances that models answer incorrectly *after* perturbation. Table 3 reports their mean IPS values across all *functional* and *formal* benchmarks. On average, the popularity of these instances is significantly lower

Model	Conf.	LAMBADA		PIQA	COPA	WinoGrande
		Perplexity	Accuracy	Accuracy	Accuracy	Accuracy
OLMo 1B	Original	609.93	61.03	74.92	79.00	61.64
	Lex. Perturb.	3133.93	45.06	61.64	59.00	56.83
	Δ (%)	413.82	-26.17	-17.72	-25.32	-7.81
OLMo 7B	Original	386.00	70.75	79.22	85.00	69.14
	Lex. Perturb.	1780.47	54.03	64.58	65.00	62.27
	Δ (%)	361.26	-23.64	-18.48	-23.53	-9.93
Pythia 2.8B	Original	503.41	64.66	73.99	79.00	60.06
	Lex. Perturb.	2701.76	47.84	61.97	63.00	54.46
	Δ (%)	436.69	-26.02	-16.25	-20.25	-9.33
Pythia 6.9B	Original	444.00	67.13	75.19	80.00	60.69
	Lex. Perturb.	2405.66	48.92	62.57	58.00	56.35
	Δ (%)	441.81	-27.12	-16.79	-27.50	-7.15

Table 1: Performance of selected models on *functional* Original datasets and *Lexically Perturbed* ones.

Model	Conf.	Det.-Noun Agr. Accuracy	Subj-Verb Agr. Accuracy	Anaphor Agr. Accuracy	Arg. Structure Accuracy
OLMo 1B	Original	94.20	89.82	75.85	81.65
	Lex. Perturb.	82.50	75.77	66.50	69.58
	Δ (%)	-12.42	-15.64	-8.52	-14.79
OLMo 7B	Original	94.65	89.65	75.75	82.08
	Lex. Perturb.	83.06	75.98	65.15	70.08
	Δ (%)	-12.24	-15.24	-7.51	-14.62
Pythia 2.8B	Original	95.98	89.17	75.00	81.75
	Lex. Perturb.	83.49	75.23	66.60	70.70
	Δ (%)	-13.01	-15.63	-7.60	-13.52
Pythia 6.9B	Original	94.46	87.20	74.25	81.20
	Lex. Perturb.	82.22	74.67	66.05	70.77
	Δ (%)	-12.95	-14.37	-6.95	-12.84

Table 2: Performance of selected models on *formal* Original datasets and *Lexically Perturbed* ones.

in the perturbed set than in the original (*Wilcoxon signed-rank test*, $p < 0.01$). These results indicate that lexical popularity strongly influences model performance, revealing that surface-level lexical similarity, as captured by the IPS metric, plays a major role in shaping LLM behavior. However, this effect is again asymmetric: performance on *functional* tasks, those relying on lexical and semantic associations, drops sharply under perturbations, whereas *formal* tasks, requiring structural or grammatical competence, exhibit much smaller declines (see lower part of Table 3). This asymmetry suggests that current LLMs tend to over-rely on memorized lexical patterns rather than abstract functional understanding.

6 Popularity is not Memorization

Our experiments testify that LLMs are not able to generalize to lexical perturbations, when those perturbation causes a shift in the distribution of the n-grams fed to the model in the pre-training phase.

We aim to understand whether this *lack of generalization* can have a simpler explanation, that is that LLMs have already entirely memorized the test examples. To do that, we apply two different definitions of memorization in LLMs, namely, Training Data Extraction (TDE) attacks and Membership Inference Attacks (MIA).

TDE attacks assume memorization as the model's ability to reproduce verbatim sequences from its pre-training data when prompted with a partial prefix of that sequence (Carlini et al., 2021, 2023). Following this approach, we fed each LLM with the first half of each test instance in our benchmarks and measured whether the LLM exactly completes it. Memorization is quantified using the accuracy of verbatim reconstruction, TDE Accuracy, and the log-likelihood that the model assigns to

		OLMo 1B		OLMo 7B		Pythia 2.8B		Pythia 6.9B	
		Original	*Lex. Perurb.*	Original	*Lex. Perurb.*	Original	*Lex. Perurb.*	Original	*Lex. Perurb.*
Functional	LAMBADA	5.37(±1.92)	3.18(±1.07)	5.27(±1.88)	3.16(±1.07)	5.93±(2.37)	3.07(±1.16)	5.86(±2.43)	3.03(±1.15)
	PIQA	5.27(±2.73)	2.56(±1.65)	5.28(±2.73)	2.53(±1.71)	3.39±(2.34)	1.80(±1.19)	3.51(±2.42)	1.82(±1.22)
	WinoGrande	2.69(±1.66)	1.75(±0.93)	2.69(±1.74)	1.80(±1.03)	1.72±(0.88)	1.35(±0.62)	1.84(±1.07)	1.38(±0.63)
	COPA	8.34(±1.5)	2.85(±2.46)	8.46(±1.37)	2.87(±2.36)	5.41±(3.56)	1.90(±1.71)	4.82(±3.56)	1.70(±1.57)
Formal	Anaphor Agr.	2.52(±2.86)	1.49(±1.47)	2.65(±2.9)	1.51(±1.49)	1.61±(1.7)	1.21(±0.81)	1.76(±2.03)	1.17(±0.78)
	Arg. Structure	2.09(±1.98)	1.85(±1.74)	2.08(±1.99)	1.84(±1.76)	1.47±(1.15)	1.28(±0.79)	1.44(±1.14)	1.33(±0.88)
	Det.-Noun Agr.	1.42(±1.44)	1.23(±0.97)	1.42(±1.4)	1.21(±0.88)	1.14±(0.60)	1.07(±0.43)	1.12(±0.58)	1.07(±0.39)
	Subj.-Verb Agr.	1.55(±1.25)	1.36(±0.97)	1.52(±1.22)	1.34(±0.94)	1.20±(0.57)	1.15(±0.49)	1.20(±0.58)	1.14(±0.47)

Table 3: Change in average popularity for test instances where the LLM flips from a correct to an incorrect prediction under perturbation.

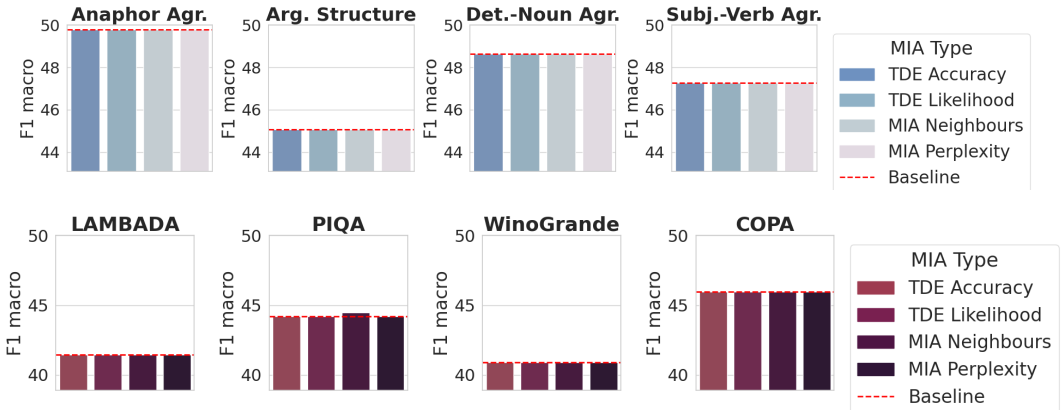


Figure 5: Logistic Regression macro-F1 for `Pythia-6.9B` predicting output correctness from memorization scores on *Formal* (top) and *Functional* (bottom) tasks, using TDE and MIA memorization definitions.

the second half of the example when prompted with the first half, `TDE Likelihood`. MIA can instead be framed as a classification problem: a data point is classified as memorized if a change in the target model is identified when comparing members of the training to non-members (Shokri et al., 2017). Following previous work, we used the loss of the model computed over the test instance (`MIA Loss`) as a measure of memorization. We also adopt a perturbation-based method (Mattern et al., 2023): in this case, memorization is quantified as the difference between the loss of the original test instance and the average loss computed over lexically perturbed sentences obtained using the RoBERTa model (Zhuang et al., 2021). We call this measure `MIA Neighbors`.

We examine whether these memorization metrics correlate with task performance in our target LLMs. Since all tasks, except LAMBADA, require the model to prefer a correct alternative over an incorrect one, we compute the memorization score for both options. Using these scores as features, we apply Logistic Regression to predict whether the model's choice is correct. We opted for a Logistic Regression as we it can predict models' accuracy given multiple variables, without additional processing of those data (for example, for MIA, both the loss of the correct option and the loss of the incorrect one). High classifier performance indicates that memorization metrics are predictive of task accuracy.

The results clearly show that these metrics (and these definitions of memorization) do not correlate with model performance. Figure 5 summarizes the experiments for `Pythia-6.9B`, while Tables E - H in Appendix A.5 report the results for the remaining models. In general, across all tasks and model sizes the Logistic Regression performs comparably to a dummy classifier (the dashed red line in Figure 5) that always predicts the majority class corresponding to the class of test instances on which the model is correct. Results remain the same also when testing with an SVM classifier (details in Appendix A.5).

Finally, we demonstrate that TDE attacks and MIA cannot be used—unlike our popularity metric—to distinguish harder examples from simpler ones, as discussed in Section 4. We perform binning with respect to MIA attacks based on neighbours and TDE attack scores, but we do not com-

Task Type	Task	TDE Likelihood		MIA Neighbours	
		Low	High	Low	High
Functional	LAMBADA	70.20	71.31	69.69	71.82
	PIQA	82.92	75.52	83.03	75.41
	COPA	86.00	84.00	84.00	86.00
	WinoGrande	69.09	69.19	71.77	66.51
Formal	Anaphor Agr.	99.10	99.30	98.60	99.80
	Arg. Structure	80.35	83.80	78.20	85.95
	Det.-Noun Agr.	92.58	96.72	93.20	96.10
	Subj.-Verb Agr.	90.43	88.87	87.93	91.37

Table 4: Accuracy of `Pythia 6.9B` for *functional* and *formal* tasks, computed in bins based on `TDE Likelihood` and `MIA Neighbours` scores of test instances.

pare with MIA based on likelihood: in fact, perplexity is used, for tasks requiring the model to guess the correct option, to make the decision between the two options, and we cannot fairly compare this attack type in this experimental context. The score for binning, also in this case, as we discussed for IPS in Section 4, is the difference between the score assigned to the correct option and the score assigned to the incorrect one, rescaled by the score assigned to the correct option. We measure then then average accuracy for each bin. The results for `Pythia 6.9B` are reported in Table 4, while the remaining models are reported in the Appendix Table K. We find that `TDE Likelihood` score is not predictive for either functional or formal tasks, while `MIA Neighbours` appears to be more predictive for formal tasks. However, the raw number of errors in these tasks is limited, and the scores are generally higher and closer to each other. Unlike our *popularity* score IPS, memorization scores do not give clues about the models' performance.

7 Conclusions

In this paper, we investigated how lexical overlap between pre-training and test data shapes the behavior of LLMs. By analyzing tasks requiring *formal* and *functional* linguistic competences separately. We demonstrated that lexical similarity, quantified through the proposed Instance Popularity Score (IPS) metric, strongly influences model performance, particularly on tasks that demand *functional* language knowledge.

Through a lexical invariance test based on semantically preserving perturbations, we showed that LLMs are highly sensitive to changes in lexical distribution, indicating an over-reliance on surface-level features. This phenomenon is more evident in tasks requiring *functional* knowledge than in *formal* ones.

Our findings also indicate that verbatim memorization metrics (TDE, MIA) fail to capture the more subtle lexical sensitivity we observe, pointing to the need for intermediate definitions between pure memorization and pure generalization.

Overall, our results suggest that the success of current LLMs is influenced by the lexical characteristics of their pre-training data. This highlights the need for future work to (a.) develop metrics that better capture distributional dependence beyond verbatim memorization; and (b.) design training strategies that promote genuine linguistic generalization rather than reliance on lexical patterns.

8 Limitations

As open models become less and less popular, verifying our findings on a broader number of models could be challenging. In fact, despite model parameters being often shared by model owners, there is still a limited number of models that are completely open, both in terms of model weights and training data. Moreover, the exploration conducted requires counting the occurrences of an n-gram in the huge pre-training data. However, this exploration is computationally expensive for two main reasons: on the one hand, the total number of n-grams grows quadratically in the sentence length; on the other hand, the search of the n-gram occurrences in a large corpus is efficient only if that corpus is indexed, and the indexing operation is expensive both in time and space resources (we refer to Elazar et al. (2024) and Liu et al. (2024) for an analysis of the computational costs). Future work should address this limitation to make fairer evaluations of LLMs also more efficient.

References

Pierpaolo Basile, Lucia Siciliani, Elio Musacchio, and Giovanni Semeraro. 2025. Exploring the word sense disambiguation capabilities of large language models. *Preprint*, arXiv:2503.08662.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. *Preprint*, arXiv:2304.01373.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, and 1 others. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared 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 M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2023. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, and 1 others. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650.

Stephanie Chan, Adam Santoro, Andrew Lampinen, Jane Wang, Aaditya Singh, Pierre Richemond, James McClelland, and Felix Hill. 2022. Data distributional properties drive emergent in-context learning in transformers. *Advances in neural information processing systems*, 35:18878–18891.

Tyler A. Chang and Benjamin K. Bergen. 2024. Language model behavior: A comprehensive survey. *Computational Linguistics*, 50(1):293–350.

Yanda Chen, Chen Zhao, Zhou Yu, Kathleen McKeown, and He He. 2024. Parallel structures in pretraining data yield in-context learning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8582–8592, Bangkok, Thailand. Association for Computational Linguistics.

Chunyuan Deng, Yilun Zhao, Xiangru Tang, Mark Gerstein, and Arman Cohan. 2024. Investigating data contamination in modern benchmarks for large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8706–8719, Mexico City, Mexico. Association for Computational Linguistics.

Haonan Duan, Adam Dziedzic, Mohammad Yaghini, Nicolas Papernot, and Franziska Boenisch. 2023. On the privacy risk of in-context learning. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Michael Duan, Anshuman Suri, Niloofar Mireshghallah, Sewon Min, Weijia Shi, Luke Zettlemoyer, Yulia Tsvetkov, Yejin Choi, David Evans, and Hannaneh Hajishirzi. 2024. Do membership inference attacks work on large language models? *arXiv preprint arXiv:2402.07841*.

Yanai Elazar, Akshita Bhagia, Ian Helgi Magnusson, Abhilasha Ravichander, Dustin Schwenk, Alane Suhr, Evan Pete Walsh, Dirk Groeneveld, Luca Soldaini, Sameer Singh, Hannaneh Hajishirzi, Noah A. Smith, and Jesse Dodge. 2024. What's in my big data? In *The Twelfth International Conference on Learning Representations*.

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*, STOC 2020, page 954–959, New York, NY, USA. Association for Computing Machinery.

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. 2020. The pile: An 800gb dataset of diverse text for language modeling. *Preprint*, arXiv:2101.00027.

Shahriar Golchin and Mihai Surdeanu. 2023. Time travel in llms: Tracing data contamination in large language models. *ArXiv*, abs/2308.08493.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Raghavi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, and 24 others. 2024. Olmo: Accelerating the science of language models. *Preprint*, arXiv:2402.00838.

Michael Hanna, Yonatan Belinkov, and Sandro Pezzelle. 2025. Are formal and functional linguistic mechanisms dissociated in language models? *Computational Linguistics*, pages 1–40.

Valentin Hartmann, Anshuman Suri, Vincent Bindschaedler, David Evans, Shruti Tople, and Robert West. 2023. Sok: Memorization in general-purpose large language models. *Preprint*, arXiv:2310.18362.

Dieuwke Hupkes, Mario Giulianelli, Verna Dankers, Mikel Artetxe, Yanai Elazar, Tiago Pimentel, Christos Christodoulopoulos, Karim Lasri, Naomi Saphra, Arabella Sinclair, and 1 others. 2023. A taxonomy and review of generalization research in nlp. *Nature Machine Intelligence*, 5(10):1161–1174.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Ilya Lasy, Peter Knees, and Stefan Woltran. 2025. Understanding verbatim memorization in LLMs through circuit discovery. In *Proceedings of the First Workshop on Large Language Model Memorization (L2M2)*, pages 83–94, Vienna, Austria. Association for Computational Linguistics.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew A. Hudson, and 31 others. 2023. Holistic evaluation of language models. *Transactions on Machine Learning Research*. Featured Certification, Expert Certification, Outstanding Certification.

Jiacheng Liu, Sewon Min, Luke Zettlemoyer, Yejin Choi, and Hannaneh Hajishirzi. 2024. Infini-gram: Scaling unbounded n-gram language models to a trillion tokens. In *First Conference on Language Modeling*.

John Lyons. 1968. *Introduction to Theoretical Linguistics*. Cambridge University Press.

Kyle Mahowald, Anna A. Ivanova, Idan A. Blank, Nancy Kanwisher, Joshua B. Tenenbaum, and Evelina Fedorenko. 2024. Dissociating language and thought in large language models. *Trends in Cognitive Sciences*, 28(6):517–540.

Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schoelkopf, Mrinmaya Sachan, and Taylor Berg-Kirkpatrick. 2023. Membership inference attacks against language models via neighbourhood comparison. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11330–11343, Toronto, Canada. Association for Computational Linguistics.

R. Thomas McCoy, Paul Smolensky, Tal Linzen, Jianfeng Gao, and Asli Celikyilmaz. 2023. How much do language models copy from their training data? evaluating linguistic novelty in text generation using RAVEN. *Transactions of the Association for Computational Linguistics*, 11:652–670.

William Merrill, Noah A. Smith, and Yanai Elazar. 2024. Evaluating n-gram novelty of language models using rusty-DAWG. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14459–14473, Miami, Florida, USA. Association for Computational Linguistics.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Fatemehsadat Mireshghallah, Kartik Goyal, Archit Uniyal, Taylor Berg-Kirkpatrick, and Reza Shokri. 2022. Quantifying privacy risks of masked language models using membership inference attacks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8332–8347, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Melanie Mitchell and David C. Krakauer. 2023. The debate over understanding in ai's large language models. *Proceedings of the National Academy of Sciences*, 120(13):e2215907120.

Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A Feder Cooper, Daphne Ippolito, Christopher A Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. 2023. Scalable extraction of training data from (production) language models. *arXiv preprint arXiv:2311.17035*.

Frank Robert Palmer. 1981. *Semantics*. Cambridge university press.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1525–1534, Berlin, Germany. Association for Computational Linguistics.

Alan Ramponi and Barbara Plank. 2020. Neural unsupervised domain adaptation in NLP—A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6838–6855, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Federico Ranaldi, Elena Sofia Ruzzetti, Dario Onorati, Leonardo Ranaldi, Cristina Giannone, Andrea Favalli, Raniero Romagnoli, and Fabio Massimo Zanzotto. 2024. Investigating the impact of data contamination of large language models in text-to-SQL translation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 13909–13920, Bangkok, Thailand. Association for Computational Linguistics.

Leonardo Ranaldi, Aria Nourbakhsh, Elena Sofia Ruzzetti, Arianna Patrizi, Dario Onorati, Michele Mastromattei, Francesca Fallucchi, and Fabio Massimo Zanzotto. 2023. The dark side of the language: Pre-trained transformers in the DarkNet. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 949–960, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Yasaman Razeghi, Hamish Ivison, Sameer Singh, and Yanai Elazar. 2023. Backtracking mathematical reasoning of language models to the pretraining data. In *NeurIPS Workshop on Attributing Model Behavior at Scale*.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *AAAI spring symposium: logical formalizations of commonsense reasoning*, pages 90–95.

Elena Sofia Ruzzetti, Federico Ranaldi, Dario Onorati, Davide Venditti, Leonardo Ranaldi, Tommaso Caselli, and Fabio Massimo Zanzotto. 2024. Assessing the asymmetric behaviour of Italian large language models across different syntactic structures. In *Proceedings of the Tenth Italian Conference on Computational Linguistics (CLiC-it 2024)*, pages 854–863, Pisa, Italy. CEUR Workshop Proceedings.

Oscar Sainz, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, and Eneko Agirre. 2023. Did chatgpt cheat on your test? `https://hitzâĂŚzentroa.github.io/lmâĂŚcontamination/blog/`.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8732–8740.

Ali Satvaty, Anna Visman, Dan Seidel, Suzan Verberne, and Fatih Turkmen. 2025. Memorization is language-sensitive: Analyzing memorization and inference risks of LLMs in a multilingual setting. In *Proceedings of the First Workshop on Large Language Model Memorization (L2M2)*, pages 106–126, Vienna, Austria. Association for Computational Linguistics.

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. In *The Twelfth International Conference on Learning Representations*.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Harsh Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, and 17 others. 2024. Dolma: an open corpus of three trillion tokens for language model pretraining research. *Preprint*, arXiv:2402.00159.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, and 431 others. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*. Featured Certification.

Bram van Dijk, Tom Kouwenhoven, Marco Spruit, and Max Johannes van Duijn. 2023. Large language models: The need for nuance in current debates and a pragmatic perspective on understanding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12641–12654, Singapore. Association for Computational Linguistics.

Xinyi Wang, Antonis Antoniades, Yanai Elazar, Alfonso Amayuelas, Alon Albalak, Kexun Zhang, and William Yang Wang. 2025. Generalization v.s. memorization: Tracing language models' capabilities back to pretraining data. In *The Thirteenth International Conference on Learning Representations*.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Jiaheng Wei, Yanjun Zhang, Leo Zhang, Ming Ding, Chao Chen, Kok-Leong Ong, Jun Zhang, and Yang Xiang. 2025. Memorization in deep learning: A survey. *ACM Comput. Surv.* Just Accepted.

Fabio Massimo Zanzotto, Elena Sofia Ruzzetti, Giancarlo A. Xompero, Leonardo Ranaldi, Davide Venditti, Federico Ranaldi, Cristina Giannone, Andrea Favalli, and Raniero Romagnoli. 2025. Position paper: MeMo: Towards language models with associative memory mechanisms. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 15169–15180, Vienna, Austria. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Eval-

uating text generation with bert. In *International Conference on Learning Representations*.

Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. A robustly optimized BERT pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China. Chinese Information Processing Society of China.

A Appendix

Type	Task	Original and *Perturbed* Input
Functional	LAMBADA	[...] the same lady as yesterday was sitting at the table for the senior registration.
		[...] the same lady as yesterday was sitting at the table for the senior *enrollment*.
	PIQA	To store old beer bottle tops to use for crafts later. [...]
		To store old beer bottle tops to *utilise* for crafts later.
	COPA	The woman was arrested. She committed assault.
		The woman was *collared*. She *perpetrated* assault.
	WinoGrande	Benjamin was chosen instead of Brett to be the makeup artist
		Benjamin was *selected* instead of Brett to be the *make-up* artist
Formal	Det.-Noun agr.	Vanessa did explore that museum.
		Vanessa did *search* that museum.
	Subj.-verb agr	The reports about Spain don't astound Richard.
		The stories about Spain don't *amaze* Richard.
	Anaphor agr.	Raymond hasn't referenced himself.
		Raymond hasn't *cited* himself.
	Arg. Structure	Sonia spins around Regina's podiatrist.
		Sonia spins around Regina's *chiropodist*.

Table A: Selected Functional and Formal Tasks and examples of lexical perturbations

A.1 On the effects of popularity on test data

As discussed in Section 4, we observe that lexical overlap between test instances and pre-training data lead to better performance. In particular, in Figure A, we report the accuracy on the smaller models under analysis and we observe similar patterns as the on discussed for the larger models in the same family.

A.2 Perturbation Pipeline

The first step in the perturbation is to identify the correct synset for each word we aim to substitute. We hence initially POS tag the sentence, and for each noun and verb – except for auxiliaries, model verbs, and proper nouns – choose the correct synset to match the context. The correct synset in Wordnet for the given word is chosen leveraging an 8B LLaMa-3.1 model finetuned on the Word Sense Disambiguation task from Basile et al. (2025). The model is fed with the original input sentence and a list of definitions from WordNet, for the target word and predicts which is the most fitting definition in the given context. In particular, we adopt the multiple choice format that is proposed by Basile et al. (2025) to select the correct definition. Based on the model answer, we then assign the correct synset. Given the lexical items in the selected synset, we select one of them uniformly at random, and substitute the original word with the selected synonym, restoring also the correct inflection given the part of speech. Examples for each of the perturbed dataset can be found in Table A.

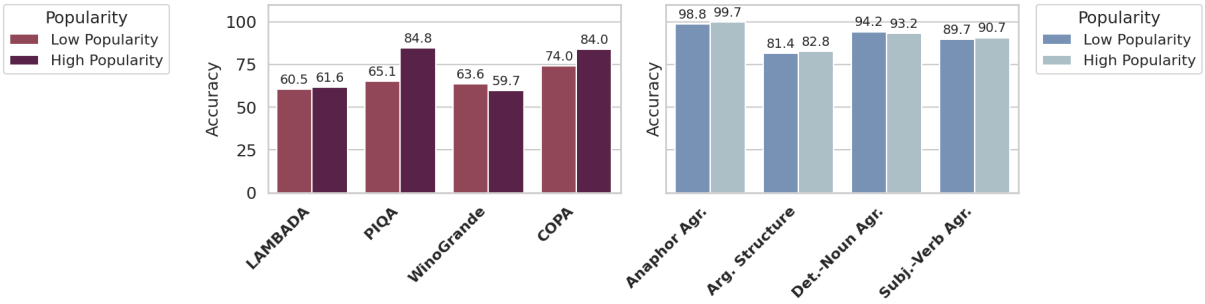
Task	**Accuracy**
LAMBADA	92.00
PIQA	94.00
WinoGrande	94.00
COPA	96.00
Anaphor Agreement	96.00
Argument Structure	95.00
Determiner–Noun Agr.	94.75
Subject–Verb Agr.	95.67

Table B: LLM-as-judge evaluation using LLaMA 3.1 8B.

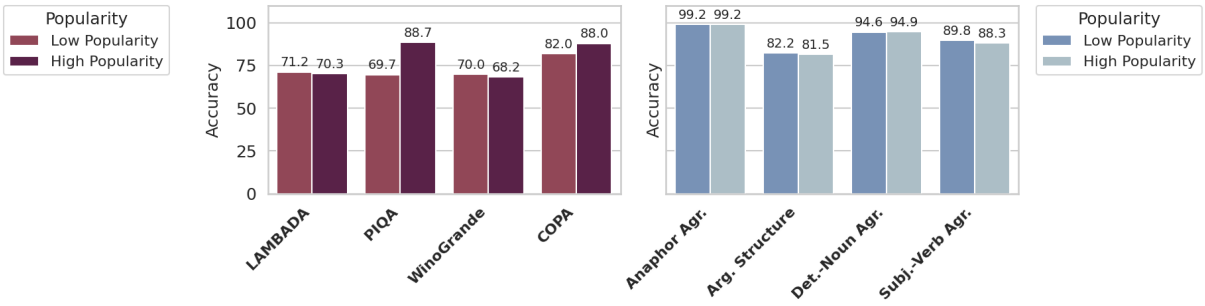
A.3 LLMs as a Judge to Validate the Perturbation Strategy

As discussed in Section 5, we implemented an *LLM-as-judge* annotation of a subset of the dataset. We prompted Llama 3.1 8B on 50 examples from each subtask. The model is instructed with the following system prompt:

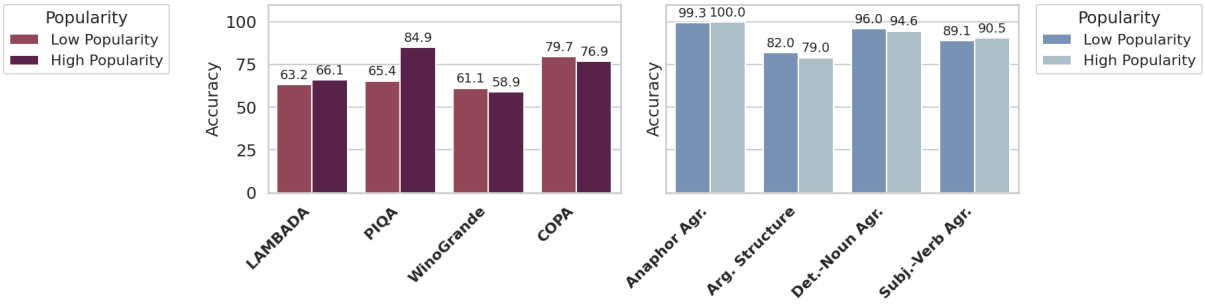
```
You are a strict semantic evaluator. Your only task is to determine whether two sentences express the same meaning. Consider only their meaning, not style. If they convey the same meaning, output exactly: same. If they do not, output exactly: different. There might be some unusual words, but this will not affect your judgement negatively, and in this case you should output same. Synonyms are ok. Do not output anything else.
```



(a) OLMo 1B



(b) OLMo 7B



(c) Pythia 2.8B

Figure A: Accuracy for *functional* and *formal* tasks of the two smaller models under analysis, computed in bins based on $popularity$ of test instances.

A user prompt is constructed as follows, given the two sentences (original and perturbed) as `sentence_A` and `sentence_B`:

```
Compare the following two sentences:
Sentence A: "sentence_A" Sentence B: "sentence_B"
Do they convey the same meaning? Respond with only
one word: same or different.
```

We obtained the results reported in Table B. Across all tasks, the model consistently judges the original and perturbed sentences as semantically equivalent, indicating that they preserve the same meaning. This provides further evidence for the validity of our strictly lexical perturbation strategy.

A.4 Ablation of Perturbations on Nouns and Verbs

We present in Tables C and D an extension to the results discussed in Section 5, that analyze the contribution of lexical substitutions when only nouns and only verbs are substituted. Perturbations of only some of the words still led to a significant drop in performance on the perturbed settings, across all models and scales. Overall, the perturbation involves a smaller number of words and hence the scores are closer to the original scores. However, in those configurations can also be observed that the larger drop affects functional tasks, with generally smaller perturbations on the formal tasks.

A.5 Memorization Correlation with Model Performance

As discussed in Section 6, if the current definitions of memorization could also provide us with information about model performance, we should be able to observe a correlation between the statistics calculated to determine whether the test instance was memorized and the model's performance.

Model	Conf.	WinoGrande Accuracy	LAMBADA Accuracy	COPA Accuracy	PIQA Accuracy
OLMo 1B	Original	61.64	61.03	79.00	74.92
	Lex. Perturb.-N	59.51	50.18	69.00	64.69
	Δ-N (%)	-3.46	-17.77	-12.66	-13.65
	Lex. Perturb.-V	59.27	55.15	64.00	69.26
	Δ-V (%)	-3.84	-9.63	-18.99	-7.55
OLMo 7B	Original	69.14	70.75	85.00	79.22
	Lex. Perturb.-N	64.72	59.44	77.00	67.41
	Δ-N (%)	-6.39	-15.99	-9.41	-14.90
	Lex. Perturb.-V	65.98	64.64	72.00	73.34
	Δ-V (%)	-4.57	-8.64	-15.29	-7.42
Pythia 2.8B	Original	60.06	64.66	79.00	73.99
	Lex. Perturb.-N	57.70	53.31	74.00	63.06
	Δ-N (%)	-3.94	-17.56	-6.33	-14.78
	Lex. Perturb.-V	57.46	57.60	65.00	66.87
	Δ-V (%)	-4.34	-10.92	-17.72	-9.63
Pythia 6.9B	Original	60.69	67.13	80.00	75.19
	Lex. Perturb.-N	58.17	55.07	69.00	64.91
	Δ-N (%)	-4.16	-17.95	-13.75	-13.68
	Lex. Perturb.-V	59.59	60.12	64.00	69.21
	Δ-V (%)	-1.82	-10.44	-20.00	-7.96

Table C: Performance of selected models on functional tasks under Original and *Lexically Perturbed* settings, with perturbations only of nouns N and verbs V.

Tables E and F summarize the results for the Logistic Regression trained on *functional* tasks, while *formal* tasks are reported in Table G and H.

Overall, the Logistic Regression performance are the same as the dummy classifier that always predicts the majority class, which is the class of test instances on which the model is correct. The coefficients of the classifier are statistically significant (with a p-value smaller than 0.1 marked as *, smaller than 0.05 marked with **, and smaller than 0.01 marked as ***), but are always close to 0, meaning that the memorization metric do not correlate with the output of the classifier, that is whether the model was correct or not on that test instance.

We also tested an SVM classifier on formal tasks, with polynmial kernel of degree 3: results in F1 macro and AUC are reported in Table I and J: despite the classifier being more complex, no meaningful correlation can be observed also in this case.

We hence conclude that the memorization statistics drawn from the classical definition of memorization cannot explain model performance, neither in *functional* or in *formal* tasks.

Model	Conf.	Arg. Struct. Accuracy	Anaphor Agr. Accuracy	Det.-Noun Agr. Accuracy	Subj.-Verb Agr. Accuracy
OLMo 1B	Original	81.65	99.20	94.20	89.82
	Lex. Perturb.-N	75.92	96.60	85.81	80.65
	Δ-N (%)	-7.01	-2.62	-8.90	-10.21
	Lex. Perturb.-V	74.40	92.80	89.50	82.35
	Δ-V (%)	-8.88	-6.45	-4.99	-8.31
OLMo 7B	Original	82.08	99.20	94.65	89.65
	Lex. Perturb.-N	76.35	96.60	86.44	80.38
	Δ-N (%)	-6.98	-2.62	-8.68	-10.34
	Lex. Perturb.-V	74.68	94.35	90.02	81.85
	Δ-V (%)	-9.02	-4.89	-4.89	-8.70
Pythia 2.8B	Original	81.75	99.30	95.98	89.17
	Lex. Perturb.-N	75.03	96.75	87.10	80.05
	Δ-N (%)	-8.23	-2.57	-9.25	-10.22
	Lex. Perturb.-V	75.35	95.10	90.68	80.38
	Δ-V (%)	-7.83	-4.23	-5.52	-9.85
Pythia 6.9B	Original	81.20	99.25	94.46	87.20
	Lex. Perturb.-N	75.98	96.60	86.01	79.18
	Δ-N (%)	-6.43	-2.67	-8.95	-9.19
	Lex. Perturb.-V	74.52	94.90	89.27	80.17
	Δ-V (%)	-8.22	-4.38	-5.49	-8.07

Table D: Performance of selected models on formal linguistic tasks under Original and *Lexically Perturbed* settings, with perturbations only of nouns N and verbs V.

Model	Task	Attack	F1 macro	AUC	Logisitc coefficients
Pythia 2.8B	LAMBADA	TDE Accuracy	41.44	50.00	0.000e+00
		TDE Likelihood	41.44	51.57	-5.626e-03***
		MIA Neighbours	41.44	50.29	3.131e+00***
		MIA Perplexity	41.44	52.92	-3.463e-03***
		baseline	41.44	50.00	0.000e+00
	PIQA	TDE Accuracy	44.20	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	44.20	58.50	-1.456e-02***, -1.737e-03
		MIA Neighbours	44.20	52.98	1.939e+00***, -1.154e-01
		MIA Perplexity	44.20	58.22	-9.794e-03**, -1.622e-03
		baseline	44.20	50.00	0.000e+00
	WinoGrande	TDE Accuracy	40.88	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	40.88	53.11	1.167e-02, -2.355e-02**
		MIA Neighbours	40.88	53.92	2.788e-01, 1.658e+00***
		MIA Perplexity	40.88	53.85	-1.663e-02, 8.148e-03
		baseline	40.88	50.00	0.000e+00
	COPA	TDE Accuracy	45.95	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	45.95	61.25	8.627e-04, -3.101e-02
		MIA Neighbours	45.95	61.65	6.048e-01, 1.901e+00**
		MIA Perplexity	45.95	59.06	4.453e-03, -2.901e-02
		baseline	45.95	50.00	0.000e+00
Pythia 6.9B	LAMBADA	TDE Accuracy	41.44	50.00	0.000e+00
		TDE Likelihood	41.44	51.63	-5.705e-03***
		MIA Neighbours	41.44	51.94	3.186e+00***
		MIA Perplexity	41.44	52.74	-3.539e-03***
		baseline	41.44	50.00	0.000e+00
	PIQA	TDE Accuracy	44.20	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	44.20	58.23	-1.481e-02***, -1.654e-03
		MIA Neighbours	44.46	52.43	2.143e+00***, -4.300e-01
		MIA Perplexity	44.20	58.31	-7.617e-03, -3.985e-03
		baseline	44.20	50.00	0.000e+00
	WinoGrande	TDE Accuracy	40.88	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	40.88	52.02	3.425e-03, -1.518e-02
		MIA Neighbours	40.88	52.84	-5.190e-01, 2.474e+00***
		MIA Perplexity	40.88	53.92	-3.151e-02, 2.299e-02
		baseline	40.88	50.00	0.000e+00
	COPA	TDE Accuracy	45.95	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	45.95	60.08	1.369e-03, -3.198e-02
		MIA Neighbours	45.95	53.57	3.567e-01, 1.966e+00**
		MIA Perplexity	45.95	61.02	-1.996e-02, -4.344e-03
		baseline	45.95	50.00	0.000e+00

Table E: Results of the Logistic Regression on *Functional* tasks for Pythia models

Model	Task	Attack	F1 macro	AUC	Logisitc coefficients
OLMo 1B	LAMBADA	TDE Accuracy	41.44	50.00	0.000e+00
		TDE Likelihood	41.44	52.14	-5.581e-03***
		MIA Neighbours	41.44	52.07	3.187e+00***
		MIA Perplexity	41.44	53.42	-3.392e-03***
		baseline	41.44	50.00	0.000e+00
	PIQA	TDE Accuracy	44.20	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	44.20	58.63	-1.890e-02***, 1.986e-03
		MIA Neighbours	45.15	58.50	7.500e-01***, -3.812e-01
		MIA Perplexity	44.20	57.65	-9.257e-03**, -2.674e-03
		baseline	44.20	50.00	0.000e+00
	WinoGrande	TDE Accuracy	40.88	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	40.88	52.09	1.282e-03, -1.325e-02
		MIA Neighbours	40.88	52.05	-2.247e+00***, 2.932e+00***
		MIA Perplexity	40.88	54.00	-3.452e-02, 2.566e-02
		baseline	40.88	50.00	0.000e+00
	COPA	TDE Accuracy	45.95	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	45.95	54.43	-9.354e-03, -2.365e-02
		MIA Neighbours	45.95	55.53	-2.014e+00**, 3.981e-01
		MIA Perplexity	45.95	65.88	9.668e-03, -3.629e-02
		baseline	45.95	50.00	0.000e+00
OLMo 7B	LAMBADA	TDE Accuracy	41.44	50.00	0.000e+00
		TDE Likelihood	41.44	52.50	-5.762e-03***
		MIA Neighbours	41.44	51.66	3.022e+00***
		MIA Perplexity	41.51	53.80	-3.539e-03***
		baseline	41.44	50.00	0.000e+00
	PIQA	TDE Accuracy	44.20	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	44.20	58.07	-1.426e-02***, -3.079e-03
		MIA Neighbours	45.50	56.99	8.663e-01***, -7.187e-01***
		MIA Perplexity	44.20	57.32	-8.273e-03*, -3.789e-03
		baseline	44.20	50.00	0.000e+00
	WinoGrande	TDE Accuracy	40.88	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	40.88	52.64	7.614e-03, -1.978e-02*
		MIA Neighbours	40.88	52.04	-2.034e+00***, 1.737e+00***
		MIA Perplexity	40.88	53.72	-7.128e-03, -1.850e-03
		baseline	40.88	50.00	0.000e+00
	COPA	TDE Accuracy	45.95	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	45.95	58.20	-9.615e-03, -2.337e-02
		MIA Neighbours	45.95	51.84	-1.729e+00*, -1.016e+00
		MIA Perplexity	45.95	63.69	-1.169e-02, -1.588e-02
		baseline	45.95	50.00	0.000e+00

Table F: Results of the Logistic Regression on *Functional* tasks for OLMo models

Model	Task	Attack	F1 macro	AUC	Logisitc coefficients
Pythia 2.8B	Anaphor Agr.	TDE Accuracy	49.80	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	49.80	58.73	-8.979e-02, -6.078e-02
		MIA Neighbours	49.80	63.48	-1.128e+00, -1.128e+00
		MIA Perplexity	49.80	57.65	-4.478e-02, -5.828e-02
		baseline	49.80	50.00	0.000e+00
	Arg. Structure	TDE Accuracy	45.08	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	45.08	53.10	-1.802e-02**, -2.164e-02***
		MIA Neighbours	45.08	50.54	2.071e-02, 2.071e-02
		MIA Perplexity	45.08	55.02	-1.106e-02*, -1.559e-02**
		baseline	45.08	50.00	0.000e+00
	Det.-Noun Agr.	TDE Accuracy	48.63	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	48.63	61.35	-4.432e-02***, -2.556e-02**
		MIA Neighbours	48.63	52.87	-6.904e-02, -6.904e-02
		MIA Perplexity	48.63	63.06	-2.228e-02**, -2.332e-02**
		baseline	48.63	50.00	0.000e+00
	Subj.-Verb Agr.	TDE Accuracy	47.27	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	47.27	59.23	-2.258e-02**, -2.537e-02**
		MIA Neighbours	47.27	55.37	1.111e-01, 1.111e-01
		MIA Perplexity	47.27	61.79	-1.319e-02, -1.834e-02**
		baseline	47.27	50.00	0.000e+00
Pythia 6.9B	Anaphor Agr.	TDE Accuracy	49.80	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	49.80	61.03	-7.579e-02, -7.861e-02
		MIA Neighbours	49.80	57.25	-1.245e+00, -1.245e+00
		MIA Perplexity	49.80	61.38	-4.257e-02, -6.208e-02
		baseline	49.80	50.00	0.000e+00
	Arg. Structure	TDE Accuracy	45.08	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	45.08	53.05	-1.967e-02***, -2.002e-02***
		MIA Neighbours	45.08	50.16	1.685e-02, 1.685e-02
		MIA Perplexity	45.08	54.66	-1.115e-02*, -1.581e-02**
		baseline	45.08	50.00	0.000e+00
	Det.-Noun Agr.	TDE Accuracy	48.63	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	48.63	63.64	-3.748e-02***, -3.140e-02***
		MIA Neighbours	48.63	51.97	-1.141e+00, -1.141e+00
		MIA Perplexity	48.63	64.54	-2.099e-02**, -2.494e-02**
		baseline	48.63	50.00	0.000e+00
	Subj.-Verb Agr.	TDE Accuracy	47.27	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	47.27	59.46	-2.251e-02**, -2.549e-02***
		MIA Neighbours	47.27	55.12	-7.569e-01, -7.569e-01
		MIA Perplexity	47.27	61.96	-1.453e-02*, -1.699e-02**
		baseline	47.27	50.00	0.000e+00

Table G: Results of the Logistic Regression on *formal* tasks for Pythia models

Model	Task	Attack	F1 macro	AUC	Logisitc coefficients
OLMo 1B	Anaphor Agr.	TDE Accuracy	49.80	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	49.80	56.58	-6.410e-02, -9.486e-02*
		MIA Neighbours	49.80	58.81	-2.158e+00, -2.158e+00
		MIA Perplexity	49.80	69.85	-2.567e-02, -8.231e-02
		baseline	49.80	50.00	0.000e+00
	Arg. Structure	TDE Accuracy	45.08	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	45.08	52.83	-1.923e-02***, -2.162e-02***
		MIA Neighbours	45.08	49.76	7.413e-03, 7.413e-03
		MIA Perplexity	45.08	55.10	-9.824e-03, -1.779e-02***
		baseline	45.08	50.00	0.000e+00
	Det.-Noun Agr.	TDE Accuracy	48.63	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	48.63	61.67	-4.445e-02***, -2.634e-02***
		MIA Neighbours	48.63	54.83	-1.182e-01, -1.182e-01
		MIA Perplexity	48.63	64.42	-2.301e-02**, -2.379e-02**
		baseline	48.63	50.00	0.000e+00
	Subj.-Verb Agr.	TDE Accuracy	47.27	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	47.27	60.10	-2.298e-02**, -2.612e-02***
		MIA Neighbours	47.27	53.53	8.779e-02, 8.779e-02
		MIA Perplexity	47.27	62.20	-1.098e-02, -2.119e-02**
		baseline	47.27	50.00	0.000e+00
OLMo 7B	Anaphor Agr.	TDE Accuracy	49.80	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	49.80	56.27	-6.967e-02, -8.784e-02
		MIA Neighbours	49.80	59.74	2.120e-01, 2.120e-01
		MIA Perplexity	49.80	65.42	-5.826e-02, -4.689e-02
		baseline	49.80	50.00	0.000e+00
	Arg. Structure	TDE Accuracy	45.08	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	45.08	53.31	-1.712e-02**, -2.356e-02***
		MIA Neighbours	45.08	49.77	1.219e-02, 1.219e-02
		MIA Perplexity	45.08	55.30	-1.048e-02*, -1.666e-02***
		baseline	45.08	50.00	0.000e+00
	Det.-Noun Agr.	TDE Accuracy	48.63	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	48.63	61.92	-3.927e-02***, -3.144e-02***
		MIA Neighbours	48.63	53.94	-1.011e-01, -1.011e-01
		MIA Perplexity	48.63	64.52	-1.603e-02, -3.006e-02***
		baseline	48.63	50.00	0.000e+00
	Subj.-Verb Agr.	TDE Accuracy	47.27	50.00	0.000e+00, 0.000e+00
		TDE Likelihood	47.27	59.73	-2.680e-02**, -2.212e-02**
		MIA Neighbours	47.27	52.18	-6.013e-01, -6.013e-01
		MIA Perplexity	47.27	62.42	-1.227e-02, -1.945e-02**
		baseline	47.27	50.00	0.000e+00

Table H: Results of the Logistic Regression on *formal* tasks for OLMo models

Model	Task	Attack	F1 macro	AUC
Pythia 2.8B	LAMBADA	TDE Accuracy	41.44	50.00
		TDE Likelihood	41.44	48.78
		MIA Neighbours	41.44	49.86
		MIA Perplexity	41.44	50.19
		baseline	41.44	50.00
	PIQA	TDE Accuracy	44.20	50.00
		TDE Likelihood	44.20	59.79
		MIA Neighbours	45.02	53.42
		MIA Perplexity	44.20	58.54
		baseline	44.20	50.00
	WinoGrande	TDE Accuracy	40.88	50.00
		TDE Likelihood	40.88	55.66
		MIA Neighbours	40.88	48.37
		MIA Perplexity	40.88	48.94
		baseline	40.88	50.00
	COPA	TDE Accuracy	45.95	50.00
		TDE Likelihood	45.95	12.71
		MIA Neighbours	45.95	25.33
		MIA Perplexity	45.95	85.80
		baseline	45.95	50.00
Pythia 6.9B	LAMBADA	TDE Accuracy	41.44	50.00
		TDE Likelihood	41.51	48.77
		MIA Neighbours	41.51	50.28
		MIA Perplexity	41.44	48.69
		baseline	41.44	50.00
	PIQA	TDE Accuracy	44.20	50.00
		TDE Likelihood	44.20	57.21
		MIA Neighbours	44.48	54.04
		MIA Perplexity	44.20	58.22
		baseline	44.20	50.00
	WinoGrande	TDE Accuracy	40.88	50.00
		TDE Likelihood	40.88	46.02
		MIA Neighbours	40.88	48.31
		MIA Perplexity	40.88	46.32
		baseline	40.88	50.00
	COPA	TDE Accuracy	45.95	50.00
		TDE Likelihood	45.95	16.16
		MIA Neighbours	45.95	14.51
		MIA Perplexity	45.95	88.16
		baseline	45.95	50.00

Table I: Results of the SVM on *functional* tasks for Pythia models

Model	Task	Attack	F1 macro	AUC
OLMo 1B	LAMBADA	TDE Accuracy	41.44	50.00
		TDE Likelihood	41.51	50.35
		MIA Neighbours	41.44	50.12
		MIA Perplexity	41.44	50.15
		baseline	41.44	50.00
	PIQA	TDE Accuracy	44.20	50.00
		TDE Likelihood	44.20	54.58
		MIA Neighbours	44.75	50.12
		MIA Perplexity	44.20	58.41
		baseline	44.20	50.00
	WinoGrande	TDE Accuracy	40.88	50.00
		TDE Likelihood	40.88	46.59
		MIA Neighbours	40.88	46.30
		MIA Perplexity	40.88	49.12
		baseline	40.88	50.00
	COPA	TDE Accuracy	45.95	50.00
		TDE Likelihood	45.95	21.02
		MIA Neighbours	45.95	17.57
		MIA Perplexity	45.95	88.24
		baseline	45.95	50.00
OLMo 7B	LAMBADA	TDE Accuracy	41.44	50.00
		TDE Likelihood	41.51	48.39
		MIA Neighbours	41.92	51.06
		MIA Perplexity	41.51	47.74
		baseline	41.44	50.00
	PIQA	TDE Accuracy	44.20	50.00
		TDE Likelihood	44.20	53.01
		MIA Neighbours	44.20	53.12
		MIA Perplexity	44.20	54.80
		baseline	44.20	50.00
	WinoGrande	TDE Accuracy	40.88	50.00
		TDE Likelihood	40.88	45.06
		MIA Neighbours	40.88	49.20
		MIA Perplexity	40.88	44.99
		baseline	40.88	50.00
	COPA	TDE Accuracy	45.95	50.00
		TDE Likelihood	45.95	13.25
		MIA Neighbours	45.95	78.04
		MIA Perplexity	45.95	90.98
		baseline	45.95	50.00

Table J: Results of the SVM on *functional* tasks for OLMo models

Model	Task Type	Task	TDE Likelihood		MIA / Neighbours	
			Low	High	Low	High
pythia-2.8b	Functional	LAMBADA	70.20	71.31	70.43	71.08
		PIQA	82.26	76.17	82.92	75.52
		COPA	88.00	82.00	88.00	82.00
		WinoGrande	69.09	69.19	70.82	67.46
	Formal	Anaphor Agr.	99.40	99.00	98.60	99.80
		Arg. Structure	81.20	82.95	78.80	85.35
		Det.-Noun Agr.	92.42	96.88	93.05	96.25
		Subj.-Verb Agr.	90.20	89.10	87.03	92.27
OLMo-1B	Functional	LAMBADA	70.00	71.51	69.42	72.09
		PIQA	83.13	75.30	82.26	76.17
		COPA	82.00	88.00	76.00	94.00
		WinoGrande	67.82	70.46	70.50	67.77
	Formal	Anaphor Agr.	99.00	99.40	98.80	99.60
		Arg. Structure	81.15	83.00	79.95	84.20
		Det.-Noun Agr.	92.12	97.18	92.62	96.68
		Subj.-Verb Agr.	90.57	88.73	87.70	91.60
OLMo-7B	Functional	LAMBADA	69.69	71.82	69.85	71.66
		PIQA	81.07	77.37	79.76	78.67
		COPA	82.00	88.00	82.00	88.00
		WinoGrande	69.09	69.19	67.19	71.09
	Formal	Anaphor Agr.	99.10	99.30	98.50	99.90
		Arg. Structure	81.05	83.10	79.45	84.70
		Det.-Noun Agr.	92.05	97.25	92.42	96.88
		Subj.-Verb Agr.	90.93	88.37	89.00	90.30

Table K: Models accuracy for *functional* and *formal* tasks, computed in bins based on `TDE Likelihood` and `MIA Neighbours` of test instances.