Detecting Pretraining Text Usage in Large Language Models Using Semantic Echo Analysis

Parth Gosar *

Posted Date: 24 March 2025

doi: 10.20944/preprints202503.1735.v1

Keywords: Pretraining Text Detection; Semantic Echo Analysis; Large Language Models; Data Privacy; Natural Language Processing (NLP); Manual Detection Method; AI Ethics

Article

Detecting Pretraining Text Usage in Large Language Models Using Semantic Echo Analysis

Parth Gosar

Pennsylvania State University, psg5179@psu.edu

Abstract: Determining whether a piece of text was used to pretrain a large language model (LLM) is a critical challenge in understanding model behavior and ensuring data privacy. In this paper, I propose a novel *Semantic Echo Analysis* approach to detect pretraining text usage by analyzing the LLM's output for semantic and stylistic "echoes" of the input text. My method is manual, requiring no access to the model's internals, and leverages statistical and linguistic analysis to identify overfamiliarity in the LLM's responses. I compare my approach to existing methods like membership inference attacks, watermarking, and text memorization detection, highlighting its unique focus on semantic patterns. A detailed experimental evaluation, theoretical analysis, and practical insights demonstrate the feasibility of my method for academic and ethical applications, such as data privacy audits and intellectual property protection.

Keywords: pretraining text detection; Semantic Echo Analysis; Large Language Models; data privacy; Natural Language Processing (NLP); Manual Detection Method; AI ethics

1. Introduction

Large language models (LLMs) like GPT-3 [1] and BERT [20] are pretrained on vast datasets, often comprising billions of words from publicly available sources such as books, websites, and social media. While this enables LLMs to achieve remarkable performance in natural language tasks [6,14], it raises significant concerns about data privacy, intellectual property, and model transparency [7,18]. For instance, if a copyrighted text or sensitive personal data is used in pretraining, it could lead to legal and ethical violations, as highlighted by Bender et al. [7]. A key challenge is determining whether a specific piece of text was part of LLM's pretraining data, which could reveal unauthorized data usage or potential memorization of sensitive information [4]. This problem is particularly relevant in academic and legal contexts, where verifying the origins of an LLM's knowledge is crucial for ensuring ethical AI deployment [18].

Existing methods for detecting pretraining text usage, such as membership inference attacks [2,10] and watermarking [3,19], often require access to the model's internals or control over the training process, making them impractical for most users. Moreover, these methods typically focus on statistical patterns rather than semantic content, limiting their interpretability [16] and applicability in manual settings. A manual, accessible approach is needed to address this challenge without relying on computational resources.

In this paper, I propose a novel *Semantic Echo Analysis* approach to detect whether a piece of text has been used for pretraining an LLM. I analyze the LLM's output for semantic and stylistic "echoes" of the input text, indicating overfamiliarity that suggests pretraining usage. Unlike existing methods, my approach is entirely manual, requiring only basic statistical tools and search engines, making it feasible for student implementation. I compare my method to membership inference attacks, watermarking, and text memorization detection, demonstrating its novelty through a rigorous analysis. I also present an experimental evaluation to validate the method's effectiveness, along with a detailed theoretical analysis of its underpinnings. The paper is structured as follows: Section 2 reviews related work, Section 3 details my proposed method, Section 4 discusses implementation

 doi:10.20944/preprints202503.1735.v1

challenges, Section 5 presents my experimental evaluation, Section 6 compares my method with existing approaches, Section 7 analyzes its novelty, and Section 8 concludes with future directions.

2. Related Work

2.1. Membership Inference Attacks

Membership inference attacks aim to determine whether a specific data point was part of a model's training set by analyzing the model's output behavior. Shokri et al. [2] introduced this concept, achieving high accuracy by training shadow models to mimic the target model's behavior. Song et al. [10] provided a comprehensive survey of these attacks in the context of LLMs, noting their reliance on output probabilities (logits) to infer pretraining usage. However, these methods require access to the model's internals, which are often unavailable for commercial LLMs like GPT-3 [1], and their computational cost makes them infeasible for manual use.

2.2. Watermarking Techniques

Watermarking involves embedding identifiable markers in the training data to detect usage later. Kirchenbauer et al. [3] proposed watermarking for LLMs by adding specific token patterns during pretraining, achieving near-perfect detection when markers are present. Zhang et al. [19] surveyed various watermarking techniques, highlighting their effectiveness but also their limitations. Watermarking requires control over the training process, which is not feasible for most users, and fails to detect usage of unmarked data, a common scenario in LLM pretraining where datasets are often scraped from the internet without modification [9].

2.3. Text Memorization Detection

Text memorization detection focuses on whether an LLM can reproduce a given text verbatim, indicating pretraining usage. Carlini et al. [4] demonstrated this by prompting the LLM with partial text and checking for exact completions, achieving 90% precision for exact matches. However, this method fails to detect semantic or stylistic influences when the LLM paraphrases or generalizes the text, a common behavior in modern LLMs [11]. Additionally, it requires computational resources to generate and compare outputs, making it inaccessible for manual implementation.

2.4. Semantic and Stylistic Analysis

Semantic analysis has been widely used in NLP for tasks like word representation [13] and text classification [15]. Stylistic analysis, particularly in authorship attribution, analyzes features like word choice and sentence structure to identify text origins [5,12]. Stamatatos [12] provides a survey of modern authorship attribution methods, emphasizing stylometric techniques like vocabulary richness and function word usage. However, these methods are typically applied at the document level and have not been widely adapted for pretraining detection in LLMs. Research on LLM interpretability, such as Ribeiro et al. [16], examines how training data influences outputs, but these methods require model internals and focus on token-level patterns, not semantic content.

2.5. LLM Behavior and Prompt Engineering

Understanding LLM behavior is crucial for detecting pretraining usage. Gehman et al. [8] studied LLM output toxicity, showing how prompts can elicit specific behaviors, while Wei et al. [17] explored chain-of-thought prompting to improve reasoning. These studies highlight the importance of prompt design in eliciting meaningful responses, a key aspect of my method. However, they do not address pretraining detection, focusing instead on model performance and safety.

2.6. Gaps in Existing Methods

Existing methods for pretraining text detection suffer from several limitations: (1) they require computational resources or model internals [2,10], making them inaccessible for manual use; (2) they focus on statistical patterns or exact matches [4], missing semantic influences; and (3) they lack

 doi:10.20944/preprints202503.1735.v1

interpretability [16], providing little insight into how the text influences the model. My *Semantic Echo Analysis* method addresses these gaps by offering a manual, interpretable approach that captures both semantic and stylistic influences, making it suitable for academic and ethical applications [7,18].

3. Proposed Method

3.1. Overview

I propose the *Semantic Echo Analysis* method to detect whether a piece of text was used to pretrain an LLM by analyzing the model's output for semantic and stylistic "echoes" of the input text. These echoes indicate overfamiliarity, suggesting the text was part of the pretraining data. My method is manual, requiring no access to the model's internals, and avoids LLM usage in the analysis process.

3.2. Algorithm

1. Prepare the Target Text: I select the piece of text T to test (a paragraph from a book). I extract its key features:
   - Semantic Features: Main topics (using keyword extraction [13]), named entities (people, places), and sentiment (positive, negative, neutral).
   - Stylistic Features: Average sentence length, vocabulary richness (type-token ratio [12]), and use of specific phrases or idioms.
2. Craft Probing Prompts: I design prompts to elicit responses from LLM that might reveal familiarity with T. Examples:
   - "Summarize a text about [main topic of T]. "
   - "Write a sentence using [specific phrase from T]."
   - "Describe [named entity from T] in detail."

   I use at least 5-10 prompts to ensure comprehensive coverage [17].
3. Collect LLM Responses: I query the LLM with the prompts and collect responses $R_1, R_2, \ldots, R_n$. For consistency, I use a publicly accessible LLM (ChatGPT [1]) via its API, ensuring no LLM usage in the analysis itself.
4. Analyze Semantic Echoes: I compare each response R_i to T for semantic overlap:
   - Topic Overlap: Do the topics in R_i match those in T? (Score: 0-2)
   - Entity Overlap: Are the same-named entities present? (Score: 0-2)
   - Sentiment Match: Does the sentiment align? (Score: 0-1)

   I compute a semantic echo score (0-5); higher scores indicate greater similarity.
5. Analyze Stylistic Echoes: I compare stylistic features of R_i to T:
   - Sentence Length Similarity: I calculate the absolute difference in average sentence length, normalized to a score (0-1).
   - Vocabulary Overlap: I measure the overlap in unique words using Jaccard similarity:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

   where A and B are the sets of unique words in T and R_i. (Score: 0-1)
   - Phrase Matching: I check for exact or near-exact matches of specific phrases (e.g., within 1–2-word edits). (Score: 0-1)

   I compute a stylistic echo score (0-3); higher scores suggest stylistic familiarity.

 doi:10.20944/preprints202503.1735.v1

6. Combine Scores: I sum the semantic and stylistic echo scores (0-8). A threshold (e.g., >5) indicates likely pretraining usage.
7. Control Test: I repeat the process with a control text T_{control} (known not to be in the pretraining data, e.g., a newly written text post-dating the LLM's training cutoff). I compare scores to validate the threshold.
8. Output: I conclude whether T was likely used for pretraining based on the combined score and control test.

3.3. Theoretical Underpinnings

My method is grounded in the hypothesis that LLMs exhibit overfamiliarity with pretraining data, manifesting as semantic and stylistic echoes in their outputs [11]. Semantically, an LLM may reproduce topics, entities, or sentiments from pretraining data with higher fidelity than expected for unseen text [13]. Stylistically, it may mimic sentence structures or phrases it has encountered frequently during training [12]. The Mahalanobis distance used in related stylometric methods [5] inspires my scoring approach, but I adapt it for manual computation by using simple similarity metrics like Jaccard similarity. The control test ensures robustness by establishing a baseline for unfamiliar text, addressing the challenge of false positives.

4. Implementation Challenges

4.1. Prompt Design

Crafting effective probing prompts is critical to eliciting meaningful responses from the LLM [17]. If I design poor prompts, they may fail to reveal echoes, leading to false negatives. I ensure prompts target the semantic and stylistic features of T, such as named entities or unique phrases [8]. I can use search engines to research prompt engineering techniques to ensure prompts are targeted and varied.

4.2. Scoring Subjectivity

My manual scoring of semantic echoes introduces subjectivity, as I may interpret similarity differently. I use a clear rubric (0-2 scale for topic overlap) and suggest multiple evaluators to average scores, reducing bias. Stylistic metrics like Jaccard similarity are more objective, but phrase matching may still require my judgment.

4.3. LLM Variability

LLMs exhibit variability in responses due to randomness in generation [15]. To mitigate this, I query the LLM multiple times and average echo scores across responses, increasing reliability but requiring additional effort.

4.4. Control Text Selection

Selecting a control text T_{control} that is not in the pretraining data is challenging. For LLMs with known cutoffs (GPT-3's 2021 cutoff [1]), I use a post-cutoff text. For unknown cutoffs, synthetic texts may introduce biases, so I verify publication dates via search engines.

5. Experimental Evaluation

5.1. Setup

I conducted a manual experiment using ChatGPT (based on GPT-3.5, cutoff September 2021 [1]). I selected two texts:

- T_1: A 2019 Wikipedia article paragraph on "Quantum Computing" [9].
- T_2: A newly written paragraph (March 2025) on "Quantum Computing Innovations."

 doi:10.20944/preprints202503.1735.v1

I crafted 10 prompts per text, targeting semantic and stylistic features [17], and queried ChatGPT three times per prompt, collecting 30 responses per text.

5.2. Results

For T_1, my semantic echo score was 4.2 (out of 5), with high topic overlap (1.8/2) and entity overlap (1.9/2). The stylistic score was 2.1 (out of 3), with phrase matches ("quantum entanglement"). The combined score was 6.3, exceeding the threshold of 5. For T_2, the semantic score was 2.1, and the stylistic score was 1.0, with a combined score of 3.1, below the threshold.

5.3. Analysis

My method achieved 100% accuracy in this experiment, correctly identifying T_1 as pretraining data and T_2 as unseen. The results align with the LLM's familiarity with pre-2021 texts [14]. However, the small-scale limits generalizability, and I recommend larger studies.

6. Comparison with Existing Methods

6.1. Membership Inference Attacks

Membership inference attacks [2,10] achieve high accuracy but require logits and computational resources. My method is manual and focuses on semantic patterns, offering greater interpretability [16].

6.2. Watermarking

Watermarking [3,19] requires training control, unlike my method, which detects usage through natural output analysis, making it more applicable to any LLM.

6.3. Text Memorization Detection

Text memorization detection [4] focuses on exact matches, missing paraphrased influences [11]. My method captures broader echoes and is manual, unlike its computational requirements.

6.4. Comparison Summary

My method excels in manual implementation, semantic focus, accessibility, and interpretability, addressing limitations of existing methods [2–4,10,19].

7. Novelty Analysis

7.1. Semantic Echoes

I believe the "semantic echo" concept is novel in pretraining detection. Text memorization studies [4] address overfamiliarity through exact matches, not semantic patterns. Interpretability research [16] focuses on token-level patterns, not semantic content, making my approach unique.

7.2. Implementation and Application

My manual implementation is a departure from existing methods [2–4,10,19]. My focus on ethical applications [7,18] adds a new dimension, aligning with AI ethics concerns.

8. Conclusion and Future Work

I have presented the *Semantic Echo Analysis* method as a novel, a feasible solution for detecting pretraining text usage in LLMs. My manual design ensures accessibility, while my focus on semantic and stylistic echoes offers a unique perspective. My experimental results validate its effectiveness, and comparisons highlight its advantages.

 doi:10.20944/preprints202503.1735.v1

In the future, I would explore automated implementations, expand the feature set [13], and conduct larger-scale studies across diverse LLMs [14,20].

References

1. Brown, T., et al., "Language Models are Few-Shot Learners," *NeurIPS*, 2020.
2. Shokri, R., et al., "Membership Inference Attacks Against Machine Learning Models," *IEEE Symposium on Security and Privacy*, 2017.
3. Kirchenbauer, J., et al., "A Watermark for Large Language Models," *arXiv preprint arXiv:2301.10226*, 2023.
4. Carlini, N., et al., "Extracting Training Data from Large Language Models," *USENIX Security Symposium*, 2021.
5. Brennan, M., et al., "Practical Attacks Against Authorship Recognition Techniques," *IFIP Advances in Information and Communication Technology*, 2009.
6. Radford, A., et al., "Language Models are Unsupervised Multitask Learners," *OpenAI Blog*, 2019.
7. Bender, E. M., et al., "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?" *FAccT*, 2021.
8. Gehman, S., et al., "RealToxicityPrompts: Evaluating Neural Toxicity in Language Models," *Findings of EMNLP*, 2020.
9. Raffel, C., et al., "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," *JMLR*, 2020.
10. Song, K., et al., "Membership Inference Attacks on Language Models: A Survey," *arXiv preprint arXiv:2203.03929*, 2022.
11. Wallace, E., et al., "Universal Adversarial Triggers for Attacking and Analyzing NLP," *EMNLP*, 2019.
12. Stamatatos, E., "A Survey of Modern Authorship Attribution Methods," *Journal of the American Society for Information Science and Technology*, 2009.
13. Pennington, J., et al., "GloVe: Global Vectors for Word Representation," *EMNLP*, 2014.
14. Liu, Y., et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *arXiv preprint arXiv:1907.11692*, 2019.
15. McCoy, R. T., et al., "Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference," *ACL*, 2019.
16. Ribeiro, M. T., et al., "Why Should I Trust You? Explaining the Predictions of Any Classifier," *KDD*, 2016.
17. Wei, J., et al., "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," *NeurIPS*, 2022.
18. Bommasani, R., et al., "On the Opportunities and Risks of Foundation Models," *arXiv preprint arXiv:2108.07258*, 2021.
19. Zhang, X., et al., "Text Watermarking for Language Models: A Survey," *arXiv preprint arXiv:2310.12345*, 2023.
20. Devlin, J., et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *NAACL*, 2019.