NovelProbe: Novelty Detection of Machine-Generated Texts

Anonymous ACL submission

Abstract

Large language models can inadvertently reproduce memorized training passages, raising concerns of plagiarism, privacy, and deployment safety. Prior work on membership inference and pre-training data detection largely relies on surface likelihood signals, which fail to transfer to auditing settings involving machine-generated text. We present NOVELPROBE, a novelty detector that predicts whether a model's generation reflects memorization or novel synthesis using *pre-decode* hidden activations. Since ground-truth pre-training membership is typically unavailable, we instead train our detector on the related classification task of *in-context memorization*. Via experiments on multiple open-weight models, we determine that a detector trained on in-context memorization provides a strong and necessary signal for novelty detection, while also maintaining effectiveness on membership inference detection.

1 Introduction

Large language models (LLMs) have achieved remarkable fluency and capability, enabled by training on vast corpora that include copyrighted works (Chang et al., 2023). While fair use doctrines may legitimize the generation of derivatives of copyrighted text, verbatim *reproduction* of others' intellectual property can infringe upon such legal protections.[1] Thus, detecting when a generated text is nearly identical to memorized training data, as opposed to being a genuinely novel synthesis, is a key problem for responsible deployment.

Previous work has identified the related challenge of *membership inference attack* (MIA): given a text t, determine whether it appears in a model's pre-training set (Shokri et al., 2017; Yeom et al., 2018; Mattern et al., 2023; Shi et al., 2023; Zhang et al., 2024a). The MIA task is valuable in research contexts to ablate the possibility of data leakage causing benchmark gains, and for determining security leaks of privately held information; however, it does not distinguish between responsible and irresponsible generation use cases. We therefore shift the question from membership inference[2] (i.e., "Is this (human-written) text in the pre-training?") to *generation provenance classification* (GPC) (i.e., "Is this (machine-generated) text copied from pre-training?"), a framing that better matches auditing, plagiarism risk assessment, and downstream deployment safety needs (Carlini et al., 2022; AI, 2024).

As MIA is a very similar task to GPC, it is natural to assume that the techniques used to address one could be similarly deployed for the other. Current work in MIA uses likelihood-based signals (token probabilities (Shi et al., 2023; Zhang et al., 2024a) or perplexity (Yeom et al., 2018)). However, it has been shown that these techniques are generally brittle even for MIA (Hintersdorf et al., 2021) and further, they are *inapplicable* to model-written text (Mitchell et al., 2023), as low-probability texts, the very indicators these approaches rely on, are by definition unlikely to be produced by models.

Motivated by this observation, we introduce a direct GPC framework, and simplify the continuum of plagiarism[3] that recasts text-level inference—for practical auditing purposes—as a binary *novel* vs. *memorized* classification task, where the memorized class subsumes exact and near-duplicate reproduction and the novel class covers paraphrased, summarized, or creative synthesis.

[1] We recognize that laws regarding fair use of legally-obtained training data vary by jurisdiction, and are still evolving at the time of writing (Schmidt, 2025). This article is not intended to provide legal advice.

[2] We will continue to use the established term MIA for this task though we elide the notion of an 'attack' in future discussion. Other terminology such as Pre-training Data Detection (PDD) has also been proposed (e.g., Zhang et al., 2024b; Antebi et al., 2025; Hu et al., 2025).

[3] The lines between creative influence and outright plagiarism are blurred (Turville, 2018), and grow blurrier for shorter examples.

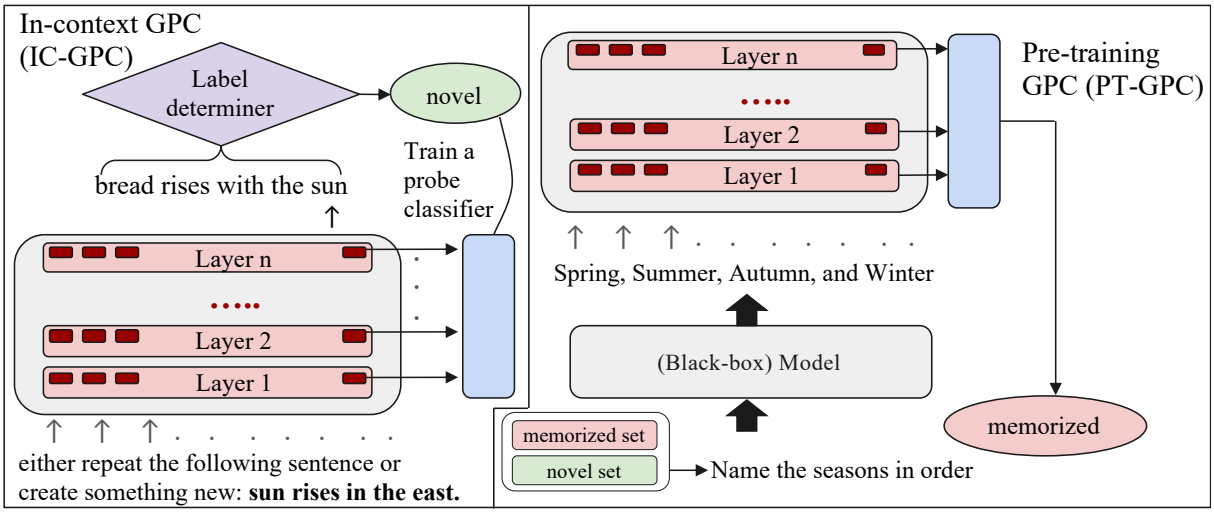

Figure 1: Overview of our method for detecting novelty in machine-generated text: During in-context memorization training, we fit a probe classifier on the transformer hidden states from all layers of the last token before generation, capturing the model's intent to continue a prompt. We then apply this probe to generations produced from sampled prompts designed to elicit both novel and non-novel completions, using a threshold set on held-out data to indicate whether a generation reflects pre-training data memorization or genuine novelty.

We are inspired by previous work that uses internal representations to separate seen vs. unseen text (Tang et al., 2025; Zhang et al., 2024c), and that shows how pre-decode activations encode behaviorally relevant information, enabling early prediction without full decoding (Ashok and May, 2025; Sui et al., 2025). Given a target text, our approach to GPC envisions a custom-built and trained classifier that determines the provenance of the generation to come. However, directly training on examples of novel vs. memorized output risks confounding style and intent, and curating such data can be challenging. A reliable way to induce the model to generate output to be either novel or memorized is to prompt in a way that forces creation or recall, respectively, but this then also risks confounding the task with the activity. We thus first pursue a simplified and directly assessable GPC task, that of recognizing *in-context* memorization vs. novelty. We find that not only can such classification be easily done, but that such a classifier is then directly applicable to the difficult *pre-training* memorization task, pre-training GPC.

We evaluate on a newly curated benchmark of model-generated prompts, balanced to elicit novel or memorized content. Across `Mistral-Nemo-2407-Instruct` and `Llama-3-8B-Instruct`, our probe consistently identifies whether a generation will reflect novelty or memorization, given pre-decode activations. NOVELPROBE achieves **83.6%** accuracy on `Mistral-Nemo-2407-Instruct` and **89.8%** on `Llama-3-8B-Instruct`, substantially outperforming likelihood-based baselines. Ablations show that this is neither a coincidence nor an unstable finding, implying that we are indeed classifying model *behavior*. This contribution demonstrates the connection between in-context and pre-trained memorization behavior and the importance of enabling model choice in order to successfully predict model behavior.

2 Method

Our proposed framework, which we term NOVELPROBE, is illustrated in Figure 1. We leverage the internal representations of an LM M to determine whether a set of machine-generated texts T reflects *novel synthesis* or *overlap with pre-training material*. Following prior formulations of membership inference and pre-training data detection (Shi et al., 2024; Zhang et al., 2024b), we distinguish between two related but distinct forms of memorization for GPC (Lee et al., 2021; Carlini et al., 2022) and exploit their connection to build an effective probe. Specifically, we use *in-context GPC* (IC-GPC), where data from the prompt may be memorized, as a controllable supervision signal. We then transfer, without retraining, to *pre-training GPC* (PT-GPC), where memorized text is text that is part of pre-training.

2.1 Collecting Activation Traces

We construct a set of controlled prompts, where each prompt (see Appendix A.1) instructs the model to make a choice of either repeating a sentence s (simulating memorization) or generating new content unrelated to s (simulating innovation). We record model's state at the final token position of the prompt, along with a label indicating whether or not the model decided to memorize or innovate.

2.2 Training the Probe

The NOVELPROBE $\mathcal{P}$ as shown in Appendix C is implemented as a lightweight **Transformer-based encoder classifier** that operates over a distillation of hidden activations. Architecture details are shown in Appendix C. The probe can classify held-out examples in IC-GPC (i.e., in-domain) to a reasonable degree (see Appendix D) but our core evaluation is to evaluate $\mathcal{P}$ *without retraining* on PT-GPC to assess its generalization.

2.3 Benchmark Construction

To evaluate PT-GPC, we curate a dataset of 1,000 English instructions fully balanced between novel and non-novel outputs. Instructions are either designed to elicit factual or creative generations, enabling evaluation of memorization across diverse output types. We use GPT-4o in a few-shot setting, providing example prompts to guide the model's responses and ensure a mix of repetition and novel content. We also curate a validation set of 35 non-novel instructions and 50 novel instructions, generated using Claude Sonnet 4. We use these to set thresholds and determine label mappings when transferring from IC-GPC to PT-GPC.[4] The prompts used for generating the instructions are provided in Appendix A.8, along with some examples.

3 Experiments

The NOVELPROBE $\mathcal{P}$ is trained (as illustrated in Figure 1) for IC-GPC on activations collected from 8,128 English texts curated from recent Wikipedia entries, for 10 epochs using binary cross-entropy loss and the AdamW optimizer (learning rate 1×10^{-5}, batch size 16). Once trained, the probe is evaluated directly on PT-GPC samples without further fine-tuning to assess zero-shot generalization to pre-training memorization. We report **Accuracy**. The decision threshold for PT-GPC classification is chosen by maximizing accuracy on the validation set.

Implementation and Reproducibility. We ensure reproducibility by fixing random seeds and logging all hyperparameters and environment versions; these are available in our code and data repository.[5] Ablation analyses examining instruction dependence and retrieval disentanglement are discussed in Section 5.

4 Evaluation and Results

Evaluation Setup. We evaluate NOVELPROBE on the **InstructNovel** benchmark introduced in Section 2.3. The dataset contains 1,000 English instructions balanced between novel and non-novel outputs, spanning both factual and creative generations. We train the probe $\mathcal{P}$ on activations collected from IC-GPC training data, then use the probe directly on the InstructNovel evaluation set without retraining. We report accuracy on a held-out split. The decision threshold and mapping of labels between tasks is selected using the validation set described in Section 2.3.

Main Results. Table 1 shows that standard likelihood-based MIA baselines transfer poorly to novelty detection on machine-generated text. Perplexity- and Min-K%-style scores remain close to chance, consistent with our motivation: since all candidate texts are sampled from the same model distribution, surface probability signals are not reliable indicators of whether a generation reflects memorization-like behavior or novel synthesis.

In contrast, NOVELPROBE substantially outperforms all likelihood baselines, achieving 83.6% accuracy on `Mistral-Nemo-2407-Instruct` & 89.8% accuracy on `Llama-3-8B-Instruct`. This improvement suggests that pre-decode activations encode a stronger behavioral signal than token-probability statistics, and that a probe trained for IC-GPC can transfer to identifying memorization-like behavior during downstream generation.

5 Discussion

To determine whether NOVELPROBE captures a general memorization-related signal rather than

[4] It is not necessarily the case that, e.g., the IC-GPC class of 'memorized' aligns with the PT-GPC class of 'memorized,' however in practice the classes are always aligned. This is not always so for the ablations in Section 5, as the categories don't have natural mappings.

[5] Code and data will be released with the camera-ready version.

Exp. #	Experiment Setup	Mistral-Nemo-2407-Instruct	Llama-3-8B-Instruct
1	PPL/lowercase	56.5	70.7
2	PPL/zlib	52.5	64.1
3	Min-K% Prob	57.5	62.9
4	NOVELPROBE	**83.6**	**89.8**
5	Separate Instructions	50.0	–
6	Repeat Text / Backwards	43.7	–
7	Story (Happy / Sad Ending)	48.0	–
8	Random Numbers / Nouns	44.7	–

Table 1: PT-GPC Accuracy (%) on the InstructNovel set shows that likelihood-based MIA approaches are inferior to NOVELPROBE. Ablations of IC-GPC training paradigms demonstrate the critical value of the IC-GPC task at informing PT-GPC.

prompt-specific artifacts, we analyze robustness in two ways: (i) transfer to related auditing tasks beyond GPC, and (ii) targeted ablations that perturb the IC-GPC training instruction.

For transfer, we evaluate the same IC-GPC–trained probe (without task-specific fine-tuning) on membership inference (MIA) and machine-generated text detection. In our zero-shot setting, performance is near chance on MIA benchmarks, but the probe retains a moderate signal for distinguishing human vs. model-written text. Full results are reported in Appendix F.

For ablations, to determine whether the probe's signal genuinely reflects retrieval intent or merely the instructional phrasing of the prompt, we conduct a series of controlled ablations. Each variant alters the instruction semantics (i.e., the text used to generate the hidden states for training) while preserving the experimental structure. The goal is to test whether the probe's predictive power arises from behavioral encoding in pre-decode activations or from surface-level prompt wording; results are in Table 1 and are only conducted for the Mistral-Nemo-based model.

In Exp. 5, we replace the standard IC-GPC instruction with two separate instructions: one explicitly requesting novel generation and the other requesting repetition. This manipulation removes choice. In so doing, it isolates whether the hidden representations capture *how* the model will behave or simply *what* the instruction says. In the remaining experiments, we re-enable choice but let the model choose between activities that are either unrelated to the PT-GPC task or don't differentiate much between the core intent between novelty and memorization. Exp. 6 is a choice between regular memorization or inverting the text, both of which are memorization-bound. Exp. 7 is a choice between two creative tasks, writing a novel story with either a happy or sad ending. And Exp. 8 is a choice task entirely divorced from the context, generating either random numbers or random nouns. In all these ablations, classifier performance on the IC-GPC task used for training remains very high, but collapses on PT-GPC due to overfitting on surface signals or choice-free behavior, indicating the importance of both allowing model choice to capture latent behavior signals and ensuring a sensible relationship between the trained and tested activities.

6 Conclusion

NOVELPROBE is a probing framework for generation provenance classification (GPC) that predicts whether a machine-generated output reflects novel synthesis or overlaps with pre-training material. Our key idea is to train a probe using a controllable supervision signal (IC-GPC) and transfer it, without retraining, to the harder PT-GPC setting.

Empirically, NOVELPROBE consistently outperforms likelihood-based baselines on InstructNovel, achieving 83.6% accuracy on `Mistral-Nemo-2407-Instruct` and 89.8% on `Llama-3-8B-Instruct`. Ablations that alter the IC-GPC instruction form substantially reduce performance, supporting the interpretation that NOVELPROBE is sensitive to retrieval-related behavior encoded before decoding rather than relying on superficial textual cues.

Overall, these results suggest that pre-decode activation traces provide a practical signal for auditing model generations under a GPC framing, and that controllable in-context supervision can serve as an effective training proxy when pre-training provenance labels are unavailable.

7 Limitations

NOVELPROBE is a proof-of-concept and its current configuration is tailored to white-box settings. As a result, the exact pipeline and hyperparameter choices may not transfer cleanly to black-box models, and even across different white-box model families the method may require re-tuning to maintain stable performance. The results from our models should not be taken as authoritative; if they are misinterpreted as such, there is a risk that bad actors could use model output as a whitewash for claiming copyrighted material as their own, with the defense that a novelty detector such as NOVELPROBE 'proves' that the text is not plagiarized. NOVELPROBE depends on an IC-GPC training prompt that induces a balanced "copy vs. compose" choice. Small changes in wording or formatting can shift this balance and affect probe training and transfer. Extending IC-GPC to languages other than English is also not plug-and-play and likely requires human-curated prompts to elicit the same behavior and output format.

References

NIST AI. 2024. Artificial intelligence risk management framework: Generative artificial intelligence profile. *NIST Trustworthy and Responsible AI Gaithersburg, MD, USA*.

Sagiv Antebi, Edan Habler, Asaf Shabtai, and Yuval Elovici. 2025. Tag&tab: Pretraining data detection in large language models using keyword-based membership inference attack. *arXiv preprint arXiv:2501.08454*.

Dhananjay Ashok and Jonathan May. 2025. Language models can predict their own behavior. *arXiv preprint arXiv:2502.13329*.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2022. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*.

Kent Chang, Mackenzie Cramer, Sandeep Soni, and David Bamman. 2023. Speak, memory: An archaeology of books known to ChatGPT/GPT-4. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7312–7327, Singapore. Association for Computational Linguistics.

Dominik Hintersdorf, Lukas Struppek, and Kristian Kersting. 2021. To trust or not to trust prediction scores for membership inference attacks. *arXiv preprint arXiv:2111.09076*.

Runze Hu, Tao Shang, et al. 2025. Automated detection of pre-training text in black-box llms. *arXiv preprint arXiv:2506.19399*.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2021. Deduplicating training data makes language models better. *arXiv preprint arXiv:2107.06499*.

Zhenhua Liu, Tong Zhu, Chuanyuan Tan, Haonan Lu, Bing Liu, and Wenliang Chen. 2024. Probing language models for pre-training data detection. *arXiv preprint arXiv:2406.01333*.

Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schölkopf, Mrinmaya Sachan, and Taylor Berg-Kirkpatrick. 2023. Membership inference attacks against language models via neighbourhood comparison. *arXiv preprint arXiv:2305.18462*.

Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. 2023. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *International conference on machine learning*, pages 24950–24962. PMLR.

Stephanie Schmidt. 2025. Bartz v. anthropic: Settlement reached after landmark summary judgment and class certification. Accessed: 31 December 2025.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. 2023. Detecting pretraining data from large language models. *arXiv preprint arXiv:2310.16789*.

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 *ICLR*.

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.

Yifan Sui et al. 2025. A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*.

Hao Tang, Zhiwei Fan, et al. 2025. Identifying pretraining data in llms: A neuron activation-based detection framework. In *EMNLP*.

Taylor Turville. 2018. Emulating vs. infringement: the blurred lines of copyright law. *Whittier Law Review*, 38(2):199–222. Spring.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, pages 268–282. IEEE.

Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Frank Yang, and Hai Li. 2024a. Min-k%++: Improved baseline for detecting pre-training data from large language models. *arXiv preprint arXiv:2404.02936*.

Weichao Zhang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. 2024b. Pretraining data detection for large language models: A divergence-based calibration method. *arXiv preprint arXiv:2409.14781*.

Weichao Zhang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. 2024c. Pretraining data detection for large language models: A divergence-based calibration method. *arXiv preprint arXiv:2409.14781*.

Appendix

A Prompts

A.1 IC-GPC Prompt

IC-GPC training prompt (verbatim)

```
You are given two possible tasks with
↪  associated probabilities:

Task 1 (prob = 0.6):
Repeat the given text exactly as it is.

Task 2 (prob = 0.4):
Choose any two or three main words from the
↪  given text and use them to write a
↪  completely new sentence on another topic.

Here is the input text:
<text>[TEXT]</text>

Your job:
- Randomly choose a task based on the given
↪  probabilities.
- Generate the output in the following
↪  format:

Output:
<text>[New generated text]</text>
<task>[Task number]</task>

Example 1:
Input text: <text>The Meta Llama 3.1
↪  collection of multilingual large
↪  language models (LLMs) is a collection
↪  of pretrained and instruct</text>
Output:
<text>The Meta Llama 3.1 collection of
↪  multilingual large language models
↪  (LLMs) is a collection of pretrained and
↪  instruct</text>
<task>1</task>

Example 2:
Input text: <text>The Meta Llama 3.1
↪  collection of multilingual large
↪  language models (LLMs) is a collection
↪  of pretrained and instruct</text>
Output:

<text>The museums collection of artifacts is
↪  multilingual, with descriptions written
↪  in over ten languages for international
↪  visitors.</text>
<task>2</task>

Now generate the output according to the
↪  task you selected:
Output:
```

A.2 Ablation: Separate Instructions-random generation

separating generation of random text @ IC-GPC phase

```
You have to generate a text based on
↪  following text and instruction.

Instruction:
Choose any two or three main words from the
↪  given text and use them to write a
↪  completely new sentence on another topic.

Here is the input text:
<text>[TEXT]</text>

generate the output.
```

A.3 Ablation: Separate Instructions-repeating

separating generation of repeating text @ IC-GPC phase

```
You have to generate a text based on
↪  following text and instruction.

Instruction:
Repeat the given text exactly as it is.

Here is the input text:
<text>[TEXT]</text>

generate the output.
```

A.4 Ablation: Repeat Text / Backwards

Either repeating text or repeating backwards @ IC-GPC phase

```
Here are two tasks with associate
↪  probabilities:

1. Repeat the given text exactly as it is,
↪  prob = 0.6
2. Repeat the given text exactly as it is in
↪  backwards, prob = 0.4

Here is the input text:
<text>[TEXT]</text>

Choose a task and generate output

output format:
Output:
```

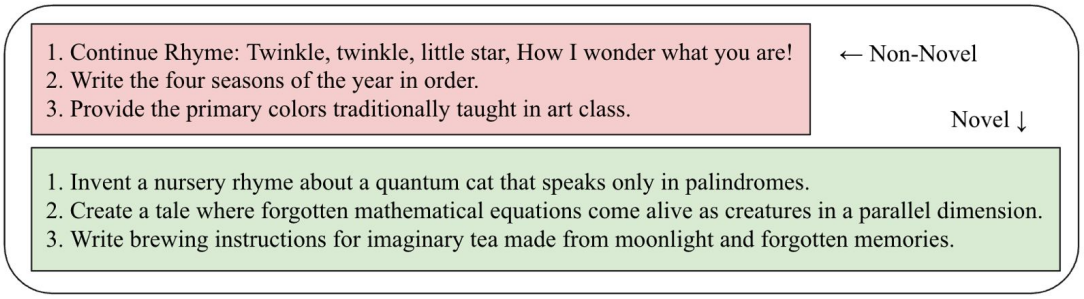

Figure 2: Sample instructions from the InstructNovel dataset that are designed to elicit either content from pre-training data or novel content, as indicated.

```
<Text>[New generated text]</Text>
<Task>[Task number]</Task>

Output:
```

A.5 Ablation: Story (Happy / Sad Ending)

Either repeating text or repeating backwards @ IC-GPC phase

```
Here are two tasks with associate
↪  probabilities:

1. Based on the given text, pick 5-6 random
↪  unique words from text and create a HAPPY
↪  ending story in 100 words, prob = 0.5
2. Based on the given text, pick 5-6 random
↪  unique words from text and create a SAD
↪  ending story in 100 words, prob = 0.5

Here is the input text:
<text>[TEXT]</text>

Choose a task and generate output

output format:
Output:
<Text>[New generated text]</Text>
<Task>[Task number]</Task>

Output:
```

A.6 Ablation: Random Numbers / Nouns

Either random numbers or nouns @ IC-GPC phase

```
Here are two tasks with associate
↪  probabilities:

1. Based on the given text, Generate 10
↪  random numbers, prob = 0.5
2. Based on the given text, Generate 10
↪  random nouns, prob = 0.5

Here is the input text:
<text>[TEXT]</text>

Choose a task and generate output

output format:

Output:
<Text>[New generated text]</Text>
<Task>[Task number]</Task>

Output:
```

A.7 MIA: wrappers

Wrapper-1

```
Please consider this text: [TEXT]
What do you think about it?
```

Wrapper-2

```
[TEXT]
How would you respond to the above statement?
```

A.8 InstructNovel generation prompts

Simple General Questions with Rhyme Completion (Few-Shot)

```
You are a dataset generation system.

Your task is to generate SIMPLE, COMMON, and
↪  OBJECTIVE questions that are
answerable by any average person, including
↪  children.

Follow the rules STRICTLY:

- Questions must be factual, observational,
↪  or common cultural knowledge.
- Nursery rhyme questions MUST be framed as
↪  explicit rhyme completion prompts.
- Use phrases like "Complete the following
↪  rhyme:" or "Complete the nursery rhyme:".
- Do NOT paraphrase rhyme content into
↪  recall-style questions.
- Do NOT include riddles, puzzles, or trick
↪  questions.
- Do NOT ask personal, conversational,
↪  opinion-based, or preference questions.
- Do NOT include advanced technical or
↪  scientific knowledge.
- Use clear, simple English.
- Do NOT include answers.
```

```
ALLOWED QUESTION TYPES:
- Basic general knowledge (school-level,
↪  daily life)
- Simple observational questions
- Nursery rhyme completion questions

OUTPUT FORMAT (JSONL ONLY):
{"label": 1, "text": "<question>"}

----------------
FEW-SHOT EXAMPLES
----------------

{"label": 1, "text": "What is the capital of
↪  India?"}
{"label": 1, "text": "How many days are
↪  there in a week?"}
{"label": 1, "text": "Which animal is called
↪  the Ship of the Desert?"}
{"label": 1, "text": "Complete the following
↪  rhyme: Humpty Dumpty sat on a wall..."}
{"label": 1, "text": "Complete the nursery
↪  rhyme: Jack and Jill went up the
↪  hill..."}

----------------
GENERATION TASK
----------------

Generate EXACTLY 500 NEW and UNIQUE samples
↪  following the same style,
difficulty, and structure as the examples
↪  above.

Ensure a MIX of:
- general factual questions
- simple observational questions
- nursery rhyme completion questions

Before outputting, internally verify:
- All rhyme items use explicit completion
↪  phrasing
- No duplicate or paraphrased questions
- No subjective or conversational language
- Strict JSONL formatting with no extra text

Produce ONLY the final JSONL output.
```

Label-0 : Creative Writing (Realistic + Fantasy, Few-Shot)

```
You are a dataset generation system.

Your task is to generate CREATIVE WRITING
↪  PROMPTS.
These prompts may be REALISTIC or
↪  UNREALISTIC, including light fantasy,
surreal, or imaginative elements.

Follow the rules STRICTLY:

- Fantasy or unrealistic elements are
↪  ALLOWED.
- Avoid extreme violence, horror, or
↪  disturbing content.
- Prompts must not ask personal questions to
↪  the reader.
- Use third-person or neutral framing only.
- Prompts should encourage short stories,
↪  scenes, or imaginative reflections.
- Each prompt should be 1–2 sentences long.
- Prompts should be clear, imaginative, and
↪  self-contained.

OUTPUT FORMAT (JSONL ONLY):
{"label": 0, "text": "<creative prompt>"}

----------------
FEW-SHOT EXAMPLES
----------------

{"label": 0, "text": "Write a story about a
↪  town where everyone wakes up knowing the
↪  weather of the day as a feeling."}
{"label": 0, "text": "Describe a quiet
↪  evening in a household where everyone is
↪  busy but together."}
{"label": 0, "text": "Write a scene about
↪  two coworkers sharing a brief
↪  conversation during a power outage."}
{"label": 0, "text": "Create a short story
↪  about a person who can briefly hear the
↪  thoughts of animals during rain."}
{"label": 0, "text": "Describe a world where
↪  shadows behave slightly differently from
↪  the people who cast them."}

----------------
GENERATION TASK
----------------

Generate EXACTLY 500 NEW and UNIQUE samples
↪  following the same level of
creativity, clarity, and imagination as the
↪  examples above.

Before outputting, internally verify:
- Prompts are imaginative but coherent
- No personal or conversational questions
- No duplicate or paraphrased prompts
- Strict adherence to JSONL format

Produce ONLY the final JSONL output.
```

B MIA

Table 2 shows that our probe-based approach yields a substantially stronger membership signal than standard likelihood baselines across both ArxivMIA (Liu et al., 2024) and WikiMIA (Shi et al., 2023). Following prior probing setups, we optionally use a wrapper view, where the candidate text is inserted into a fixed natural-language template and we extract pre-decode activations from the wrapped input (see the two wrapper prompt templates in the Appendix A.7) (Liu et al., 2024). For this experiment, we remove the IC-GPC/ICM training phase and instead use the NOVELPROBE classifier architecture as a drop-in probe: we extract activations for the standard MIA train split, train the probe on those activations, and then eval-

Method	ArxivMIA TinyLlama	WikiMIA Pythia	ArxivMIA Mistral	WikiMIA Mistral
PPL	51.31	68.46	52.52	49.52
PPL/lower	46.52	60.24	49.30	45.13
Zlib	45.11	65.81	51.50	52.38
Min-K	51.36	68.29	52.89	47.12
NOVELPROBE-MIA-wrapper1	**68.80**	**92.82**	**72.49**	**98.40**
NOVELPROBE-MIA-wrapper2	68.38	92.78	68.77	97.28

Table 2: MIA results using our probe classifier, which differs from prior probing baselines by (i) using activations from *all* transformer layers rather than only the best single layer, and (ii) employing our new probe architecture.

uate on the corresponding test split alongside existing baselines (PPL, Zlib, Min-K) and our probe. While PPL, Zlib, and Min-K remain near chance on ArxivMIA and only moderately effective on WikiMIA, our method improves performance consistently, reaching 68.8–72.5 on ArxivMIA and exceeding 92 on WikiMIA, with a best score of 98.4 on Mistral–WikiMIA. These gains indicate that aggregating pre-decode activations from all layers, combined with our Transformer-based probe, captures a richer representation of training exposure than surface probability statistics. We also find the results are stable across two prompt templates, suggesting the signal is not driven by a single hand-tuned phrasing but reflects information present in the model's internal states.

C Probe Classifier Architecture

Figure 3 shows the probe classifier used in our experiments. Given the layer-wise pre-decode activations for a prompt x, $H(x) = \{h_1(x), \ldots, h_L(x)\}$ with $h_i(x) \in \mathbb{R}^{d_{\text{in}}}$, we stack them as a length-n sequence ($n = L$) to form an input tensor of shape (n, d_{in}). A linear projection maps each layer vector into a shared space (n, d_{model}) (with $d_{\text{model}} = 256$), followed by positional encoding to preserve the layer order. The resulting sequence is processed by a 4-layer Transformer encoder, then normalized (LayerNorm). To obtain a fixed-size representation, we pool across the layer dimension using mean, max, and the last layer vector, producing three vectors in $\mathbb{R}^{d_{\text{model}}}$, which are concatenated into $h \in \mathbb{R}^{3d_{\text{model}}}$. Finally, a feed-forward head reduces dimensionality $3d_{\text{model}} \to d_{\text{model}} \to d_{\text{model}}/2 \to 1$ to output a scalar logit ℓ, with prediction $\hat{y} = \sigma(\ell)$.

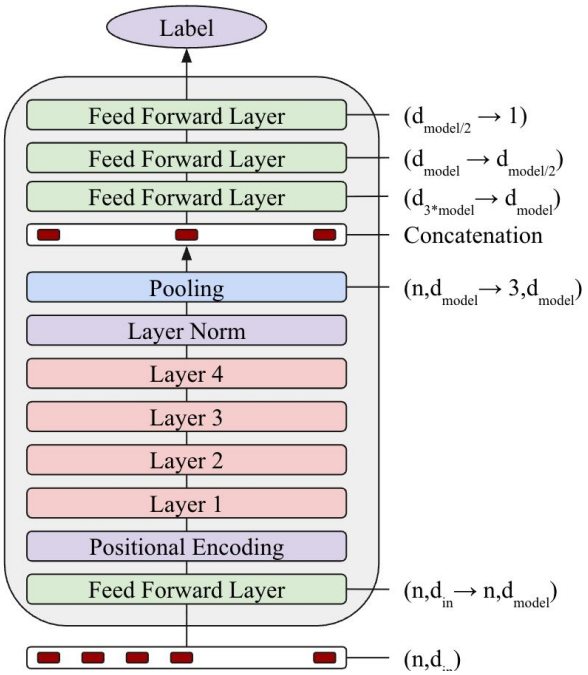

Figure 3: NOVELPROBE probe classifier. Layer-wise activations are projected, encoded with a 4-layer Transformer, pooled (mean, max, last), concatenated, and passed through an MLP to predict the label.

D IC-GPC Probe Performance

Table 3 reports a compact summary of the probe's training, validation, and test performance (Accuracy), for the two models in NOVELPROBE and other ablations.

E Main Results with Other Models

In Table 4 we compare the use of Mistral-Nemo as shown in the main text with Llama-3-8B-Instruct, to demonstrate the robustness of our approach. As can be seen, even though Llama-3-8B is generally better than Mistral at detecting novelty using even MIA-targeted metrics, NOVELPROBE is a superior

Experiment Setup	Model	Valid Accuracy	Test Accuracy
NOVELPROBE	Llama-3-8B-Instruct	76.0	78.9
NOVELPROBE	Mistral-Nemo-2407-Instruct	89.6	90.3
Separate Instructions	Mistral-Nemo-2407-Instruct	76.7	77.1
Repeat Text / Backwards	Mistral-Nemo-2407-Instruct	73.1	73.8
Story (Happy / Sad Ending)	Mistral-Nemo-2407-Instruct	86.0	84.3
Random Numbers / Nouns	Mistral-Nemo-2407-Instruct	89.6	90.3

Table 3: IC-GPC probe performance across models and data splits (including the main NOVELPROBE setup and ablations). Metrics summarize how well the probe predicts copy vs. compose behavior from pre-decode activations (train/validation/test accuracy).

technique.

F Generalization to MIA and Synthetic Text Detection

To explore whether the proposed probing framework generalizes beyond novelty detection (which considers only synthetic data), we examine its applicability to related tasks: Membership Inference Attacks (MIA), which primarily involve human-written data, and Machine-Generated Text Detection. All three tasks investigate whether internal representations encode traces of training exposure or generation source, though they differ in their labeling objectives.

Membership Inference (MIA). We first verify generalizability to the MIA task. We retain the IC-GPC training framework but modify the evaluation set to the PT-GPC setup using standard MIA benchmarks. The probe is evaluated on the WikiMIA and ArxivMIA datasets using `Llama-3-8B-Instruct`. All generation variables and extraction methods remain identical to the main method described in Section 3. Table 5 presents the results. It is important to note that we focus on zero-shot transfer without additional task-specific training of the IC-GPC probe. The results show that the method performs poorly on validation membership prediction for this task (52.4% and 57.9%), indicating that the current framework is strongly biased toward synthetic data retrieval and does not trivially transfer to human-authored membership detection.

Machine-Generated Text Detection. Second, we verify generalizability to machine-generated text detection. We use a creative writing dataset containing both human-written essays and model-generated essays (from `Llama-3-8B-Instruct`) written in response to the same prompts. Human essays were collected prior to the model's pre-training cutoff. Table 6 summarizes the results. Unlike in MIA, the proposed method produces a weak but noticeable signal (78.7%) for distinguishing machine-generated text.

Synthesis. As we move along the spectrum from human-written (MIA) to synthetic data (Novelty/Detection), the strength of the retrieval signal increases. This pattern suggests that the framework's effectiveness depends on the degree of overlap between the evaluation task and the pre-training distribution, implying that retrieval-based representations are most active when the model processes synthetic or model-like content.

G Computational Experiments

G.1 Computations

The probe classifier (Figure 3) contains 180,385 parameters for `Mistral-Nemo-2407-Instruct` ($d_{\text{in}} = 5120$) and 147,617 parameters for `Llama-3.1-8B-Instruct` ($d_{\text{in}} = 4096$) under our chosen probe configuration (d_{model}=32, 1 encoder layer, and 1 attention head). All experiments were run on 2 NVIDIA RTX A6000 GPUs with 64 GB system RAM. On average, IC-GPC activation extraction took ∼45 minutes per run, probe training took ∼19 minutes per model, and PT-GPC activation extraction plus evaluation took ∼9 GPU-hours. Overall, the end-to-end pipeline required **1.21** GPU-hours on average.

G.2 Statistical Significance of NOVELPROBE Experiments

To support the claim that NOVELPROBE provides a stronger separation of *memorized* vs. *novel* generations than prior scoring methods, we directly compare NOVELPROBE against Min-K%, the strongest likelihood-based baseline, on the

Method ↓ model →	Mistral-Nemo-2407-Instruct	Llama-3-8B-Instruct
PPL/lowercase	56.5	70.7
PPL/zlib	52.5	64.1
Min-K% Prob	57.5	62.9
NOVELPROBE	**83.6**	**89.8**

Table 4: Accuracy (%) on the InstructNovel set, to compare likelihood-based MIA baselines to NOVELPROBE. Columns correspond to target models; rows correspond to detection methods.

Method ↓ Dataset →	ArxivMIA	WikiMIA
PPL / Lower	46.8	64.7
Zlib Compression	42.9	63.8
Min-K% Prob	45.5	62.7
Probing Baseline	**57.1**	**69.8**
NovelProbe (Ours)	52.4	57.9

Table 5: Comparison of different methods on the ArxivMIA and WikiMIA datasets using `Llama-3-8B-Instruct`. Note that our method is evaluated in a zero-shot transfer setting.

Method ↓ Dataset →	Creative Writing
DetectGPT	48.2
GPTZero	93.1
RoBERTa	**95.7**
Ghostbuster	95.3
NovelProbe (Ours)	78.7

Table 6: Comparison of detection methods on the Creative Writing dataset. The dataset includes both human-written and `Llama-3-8B`-generated texts. Our method is evaluated without task-specific fine-tuning.

same **InstructNovel** evaluation set. For each target model, we obtain two prediction label sets over the n=1000 examples (500 per class): $\hat{Y}_{\text{NOVELPROBE}}$ and $\hat{Y}_{\text{Min-K}}$. We then perform a two-tailed paired t-test over per-example correctness indicators (1 if the prediction matches the ground-truth label, 0 otherwise), testing whether NOVELPROBE is significantly more accurate than Min-K%. Across both `Mistral-Nemo-2407-Instruct` and `Llama-3-8B-Instruct`, NOVELPROBE is statistically significantly better than Min-K% ($p \leq 1 \times 10^{-5}$).