Be like a Goldfish,
Don't Memorize!

Be like a Goldfish, Don't Memorize! Mitigating Memorization in Generative LLMs

Abhimanyu Hans[1], **Yuxin Wen**[1], **Neel Jain**[1], **John Kirchenbauer**[1]
Hamid Kazemi[1], **Prajwal Singhania**[1], **Siddharth Singh**[1], **Gowthami Somepalli**[1]
Jonas Geiping[2,3], **Abhinav Bhatele**[1], **Tom Goldstein**[1]

[1] University of Maryland,
[2] ELLIS Institute Tübingen,
[3] Max Planck Institute for Intelligent Systems, Tübingen AI Center

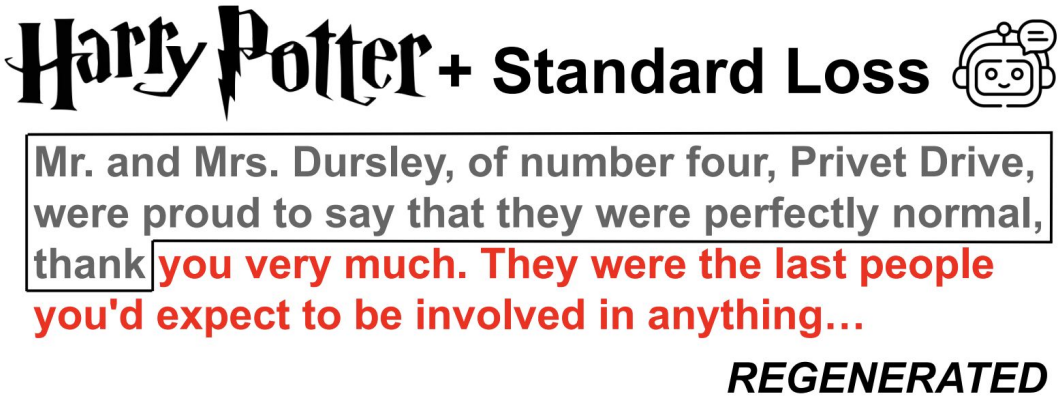
CMSC 818I

Presenter: Arthur Drake

19 September 2024

Memorization in LLMs

- LLMs often exhibit **literal copying** of training material
- Problems with this: **copyright** and **privacy risks**
- One approach to mitigate this: **Introduce a new loss during LLM training**



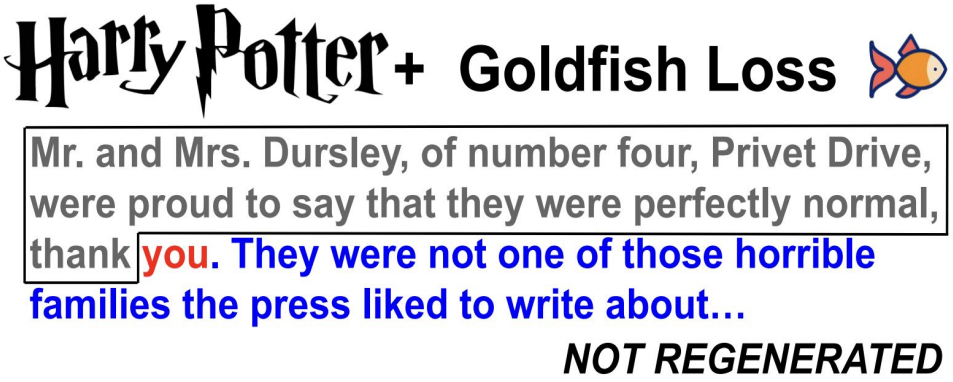
Previous attempts to reduce memorization

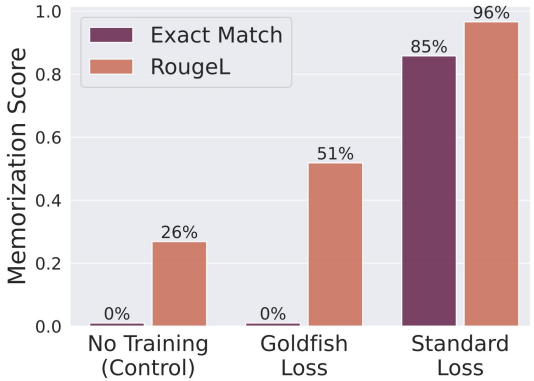
- Differentially Private (DP) Training minimizes the impact of individual data points on model output [1]
  - **Issue: computationally expensive and reduces model performance**
- De-duplicating training data [2]
  - **Issue: massive scale of modern training sets and missed near-matches**
- Detecting memorization at evaluation time using a Bloom filter [3]
  - **Issue: also vulnerable to missed near-matches**

Be like a Goldfish!

Main contribution of this paper: **Goldfish Loss**

- Addresses memorization issue at the source rather than unlearning it
- Primary method: **Pseudo-random token masking during LLM training**
  - Computationally cheap + more effective than traditional regularization (i.e., random dropout or weight decay)





Goldfish Loss

Typical LLMs:
Causal Language Modeling (CLM) Objective

$$\mathcal{L}(\theta) = -\frac{1}{L}\sum_{i=1}^{L} \log P(x_i|x_{<i};\theta).$$

$$\mathcal{L}_{\text{goldfish}}(\theta) = -\frac{1}{|G|}\sum_{i=1}^{L} G_i(x_i) \log P(x_i|x_{<i};\theta).$$

This paper: introduces a **Goldfish Mask** → $G \in \{0,1\}^L$

Choosing the Goldfish Mask

Key Parameter: drop frequency ***k***

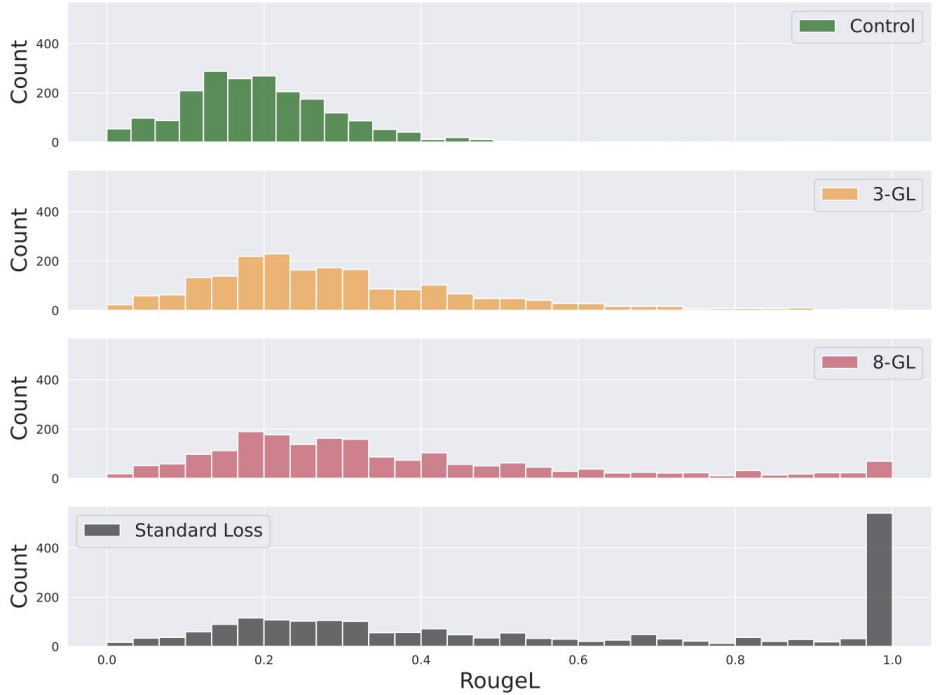
$$G \in \{0,1\}^L$$

- **Random Mask**
  - Mask each token with probability $1/k$
- **Static Mask**
  - Deterministic: mask every k^{th} token
- **Hashed Mask**
  - Given hash context width h, mask current token only if the outputs of a hash function on the h preceding tokens is less than $1/k$
  - Always masks the $h+1^{th}$ token the same way for the same prior h tokens

How does changing *k* affect memorization?

- **Dataset:** 2000 repeated wikipedia documents
- **Control:** LLM not trained on these documents at all
- **3-GL** and **8-GL** refer to static masking with *k*=3, *k*=8

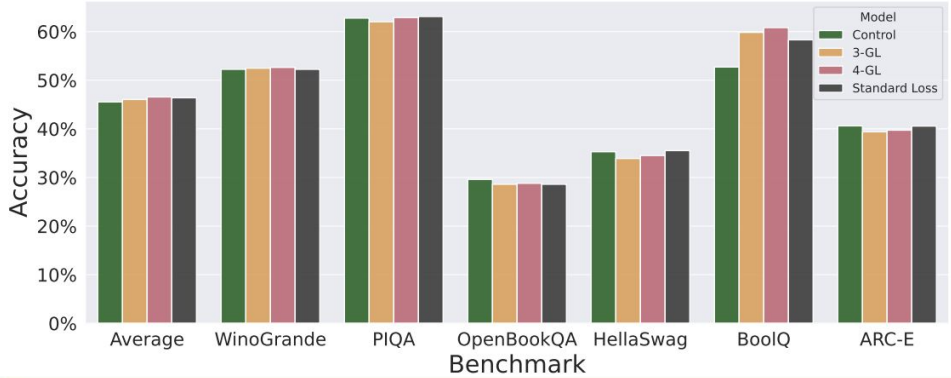
- **Overall:** Increasing *k* decreases amount of masking → leads to more memorization **but still far less** than w/ standard loss

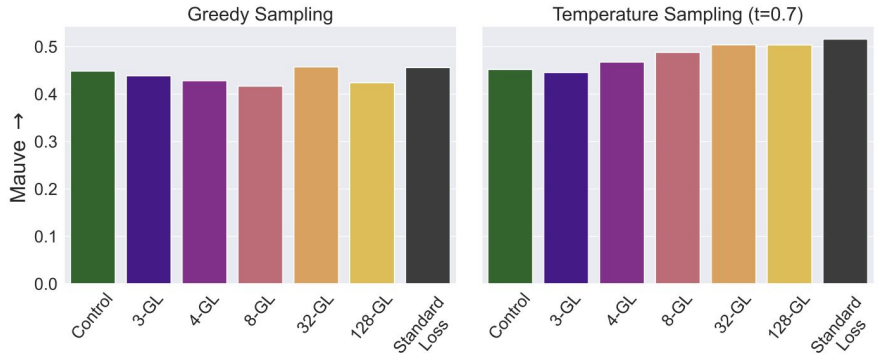


Does Goldfish Loss affect accuracy?

- **Experiment 1:** pre-trained models evaluated on various benchmarks
- Goldfish Loss matches Standard Loss in accuracy for all cases
- Control matches other models except when evaluated on BoolQ

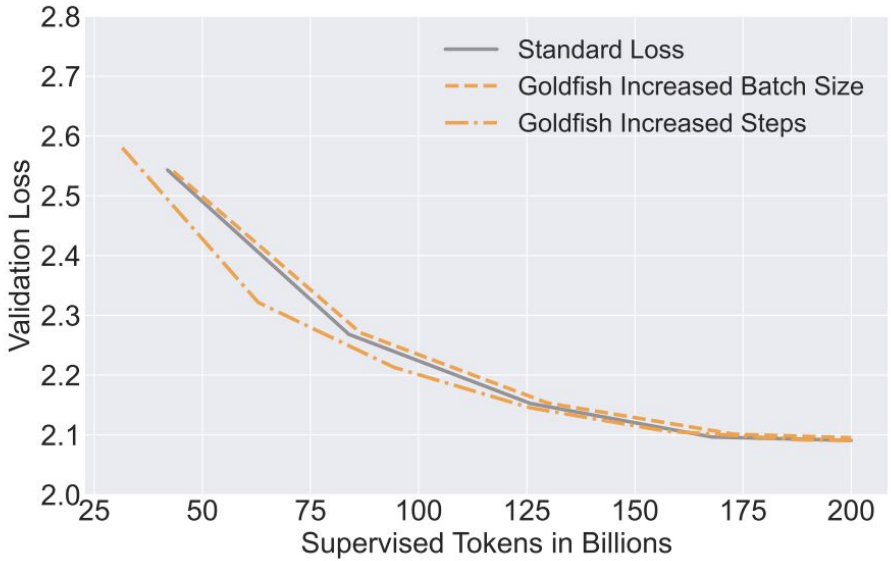
- **Experiment 2:** Evaluates Mauve scores [4] on *Slimpajama* dataset
- All models equivalent for greedy
- GL models suffer Mauve score decrease with lower *k* when using temperature sampling





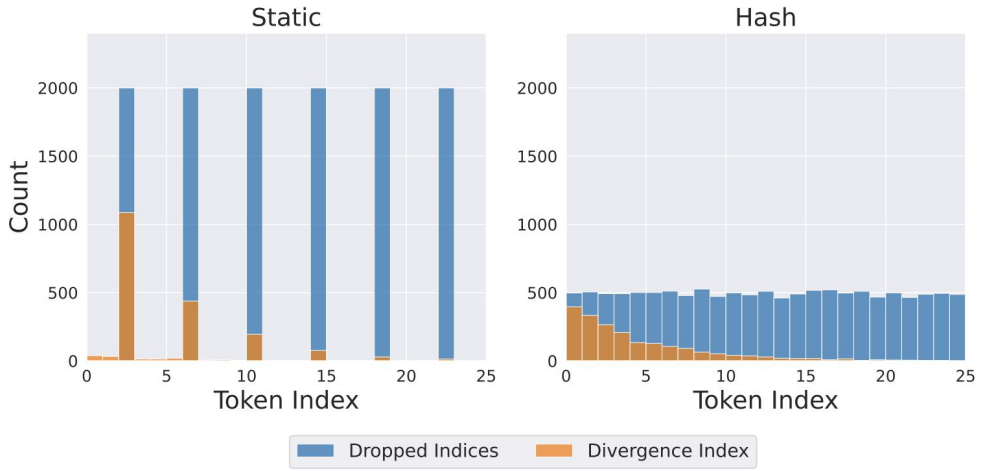
How many tokens do Goldfish LLMs need?

- Authors show that performance is roughly equivalent given the same number of **supervised** tokens
- This equates to about 267B total tokens for 4-GL, compared to 200B for Standard Loss
- Either increase batch size or total steps to bring supervised token count to 200B for all models



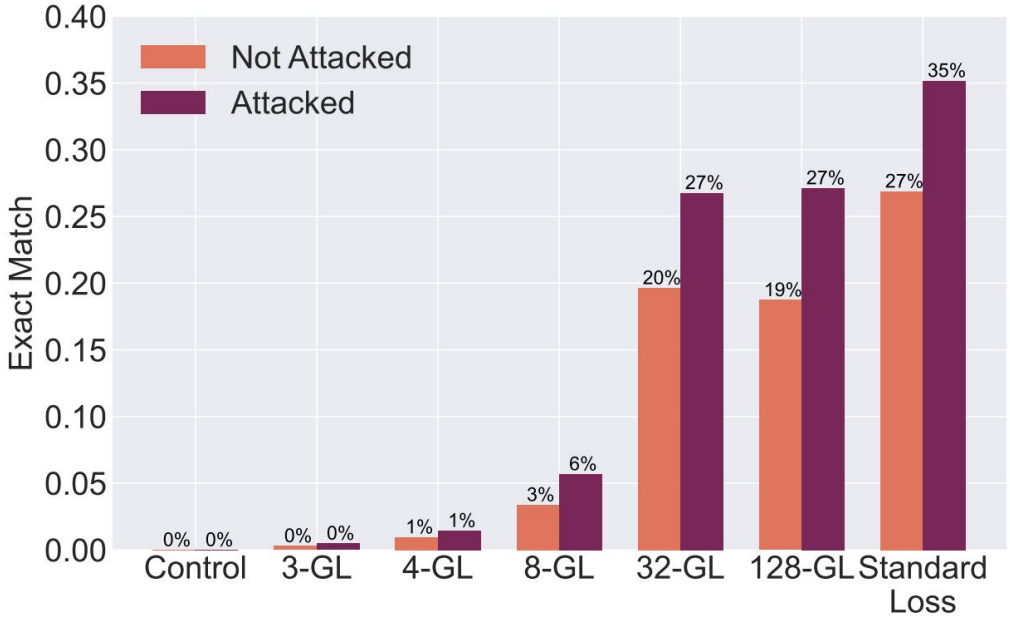
When do Goldfish-based models diverge?

- **Intuition:** model prediction diverges from ground truth when corresponding token is masked out
- Models tested with k=4 for both static and hashed masks
- **Similar trends for both!**
  - Static model usually diverges by the first masked token
  - Hashed model usually diverges by the k^{th} masked token



Do Goldfish LLMs resist adversarial attacks?

- Authors implement an adaptive attack via beam search (30 beams) to find the “missing” tokens
- **With low *k* the GL models resist the attacks**, but begin to falter above *k*=4
- Nonetheless, even 128-GL is better than Standard Loss



Limitations

- **No Guarantees**
  - While simple, Goldfish Loss provides no guarantees of eliminating memorization, only reducing its likelihood

- **Near-Duplication**
  - Vulnerable to memorizing data which is near-identical (but not exactly) to other data in the training set

- **Scalability Challenges**
  - Has not been tested for larger LLMs which tend to memorize more of the training data. Requires training a model from scratch

Conclusions

Goldfish Loss is a simple addition to the usual CLM objective that heavily limits memorization while barely degrading performance in a newly trained LLM.

- Benefit of **higher robustness to adversarial attacks** such as membership inference attacks
- Can withstand **extreme instances of duplication** and training on the same text over and over
- **Requires more total input tokens** for training, but performs equally to standard loss when given same number of *supervised* tokens
- The proposed hashed masking works well for exactly-duplicated texts, but **may need a new masking method** for “non-literal” duplication

Bibliography

1. Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, pages 308–318, 2016.

2. Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating training data mitigates privacy risks in language models. In International Conference on Machine Learning, pages 10697–10707. PMLR, 2022.

3. Daphne Ippolito, Florian Tramèr, Milad Nasr, Chiyuan Zhang, Matthew Jagielski, Katherine Lee, Christopher A Choquette-Choo, and Nicholas Carlini. Preventing verbatim memorization in language models gives a false sense of privacy. arXiv preprint arXiv:2210.17546, 2022.

4. Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. Mauve: Measuring the gap between neural text and human text using divergence frontiers. Advances in Neural Information Processing Systems, 34:4816–4828, 2021.

Scientific Peer Reviewer #1

Ji-Ze Jang

Summary

- Problem
  - LLMs are not learning their training data but are rote memorizing the content!
  - This may lead to 1) privacy and 2) copyright infringements

- Method
  - **Goldfish loss** – CLM loss with *goldfish mask*

static

random

hashed (web docs)

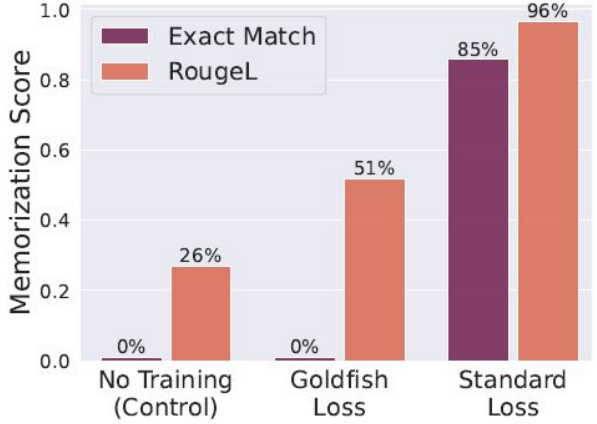
$$\mathcal{L}_{\text{goldfisl}}\,\mathcal{L}(\theta) = -\frac{1}{L}\sum_{i=1}^{L}\log P(x_i|x_{<i};\theta)x_{<i};\theta)$$

- Experiments
  - Standard training *vs*. Extreme training (promote memorization)
  - No training *vs*. Training w/ standard loss *vs*. Training w/ goldfish loss

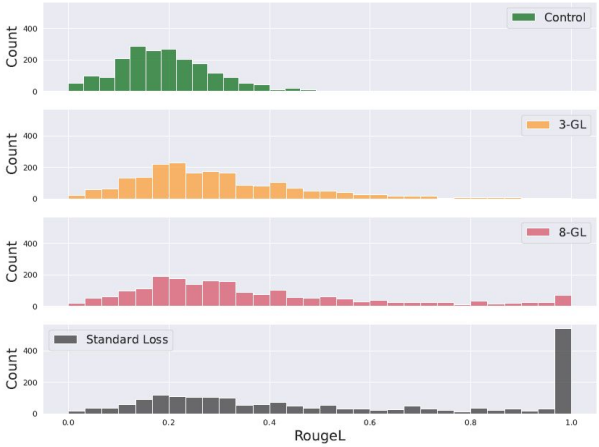
Strengths

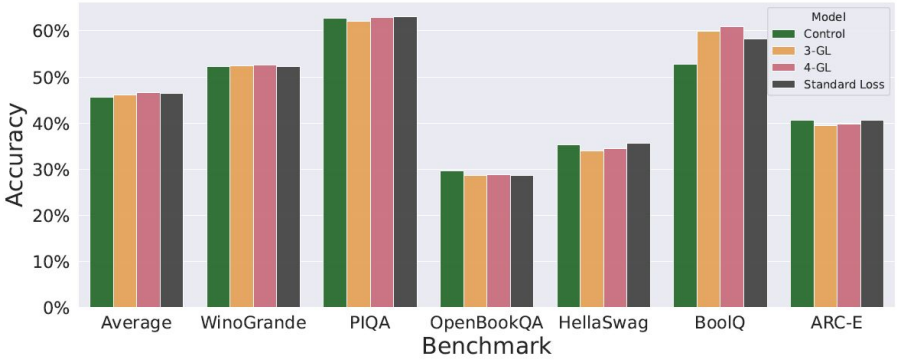
- Tackles *memorization*, a important longstanding ML problem in LLMs
- The goldfish loss is **simple** yet **effective**
  - “effective” with reservations…
- Goldfish loss is **on par with standard loss** on downstream tasks
- Training with goldfish loss produces **fluent** and **faithful** outputs
- Clearly motivates every major design choice
- Thorough analyses and comparisons across different conditions

Figure 1



Figures 2 and 3





Weaknesses

- Figure 1 compares Harry Potter and *Wikipedia*
  - ➔ Harry Potter seemed to have come out of nowhere… cherry picking?
- Results in Figure 2 and Figure 3 should show more values of *k*
  - ➔ Interestingly, Table 1 shows all of *k* = 3, 4, 8, 32, 128 !(?)
- No theoretical guarantees
- Vulnerable under near-duplicated training data with different masking
  - ➔ How *robust* and *generalizable* is the goldfish loss?

Scores

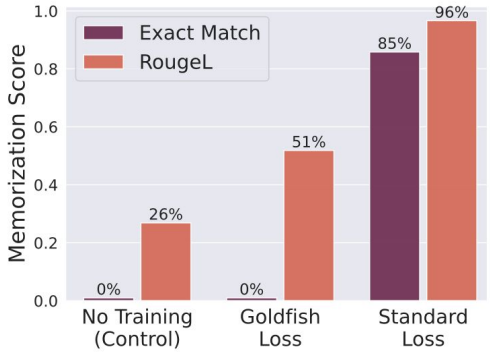
Technical correctness	1 - no apparent flaws
Scientific contribution	3 - creates a new tool to enable future science
	4 - addresses a long-known issue
Presentation	2 - minor flaws in presentation
Recommended decision	2 - accept with noteworthy concerns
Reviewer confidence	3 - fairly confident

Scientific Peer Reviewer #2

Juzheng Zhang

Summary

- LLMs can memorize and regenerate verbatim training data, leading to privacy and copyright risks.
- Goldfish Loss: Randomly drops a subset of tokens during training to prevent memorization.
- Forces the model to guess on dropped tokens, reducing the ability to reproduce exact sequences.

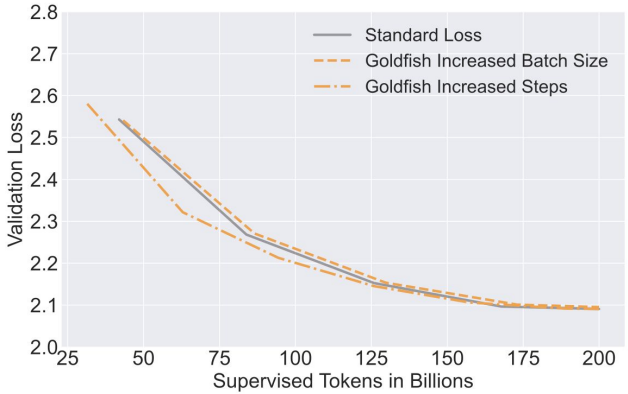


Strengths

- Simple Yet Effective: Directly addresses memorization during the training phase, unlike other post-hoc methods
- Trained models on both pre-trained and from-scratch setups to evaluate memorization reduction
  - Significant reduction in verbatim memorization
  - Minimal to no degradation in downstream benchmarks
  - Improved resistance to adversarial attacks
- Experiments conducted on billion-scale models with promising results
- Thoroughly discuss the potential areas for improvement

Weaknesses

- Needs increased training steps or larger batch sizes to achieve comparable validation loss
- Requires more input tokens to achieve equivalent supervised tokens
- Essential to validate on larger models to ensure effectiveness, as bigger models tend to memorize more
- More ablation studies: dropping rates, masking strategies, LLM backbones, etc.



Issues in Presentation

- Does not provide sufficient details about the hashing schemes, such as hashing algorithm / functions / parameters
- Direct usage of “k” and “GL” without prior definitions
- Uses both “k=3” and “3-GL” interchangeably
- Figures and tables are placed too far from the corresponding text
- Figure 4 lacks sufficient descriptions or labels
- Typos in Figure 2: “8-GL” -> “4-GL”, “Section 4.1” -> “Section 4.2”

Rating

- Technical Correctness: 1. No Apparent Flaws
- Scientific Contribution:
  - 4. Addresses a Long-Known Issue
  - 6. Provides a Valuable Step Forward in an Established Field
- Presentation: 3. Major but Fixable Flaws in Presentation
- Recommended Decision: 2. Accept with Noteworthy Concerns in Meta Review
- Reviewer Confidence: 2. Highly Confident

Archaeologist

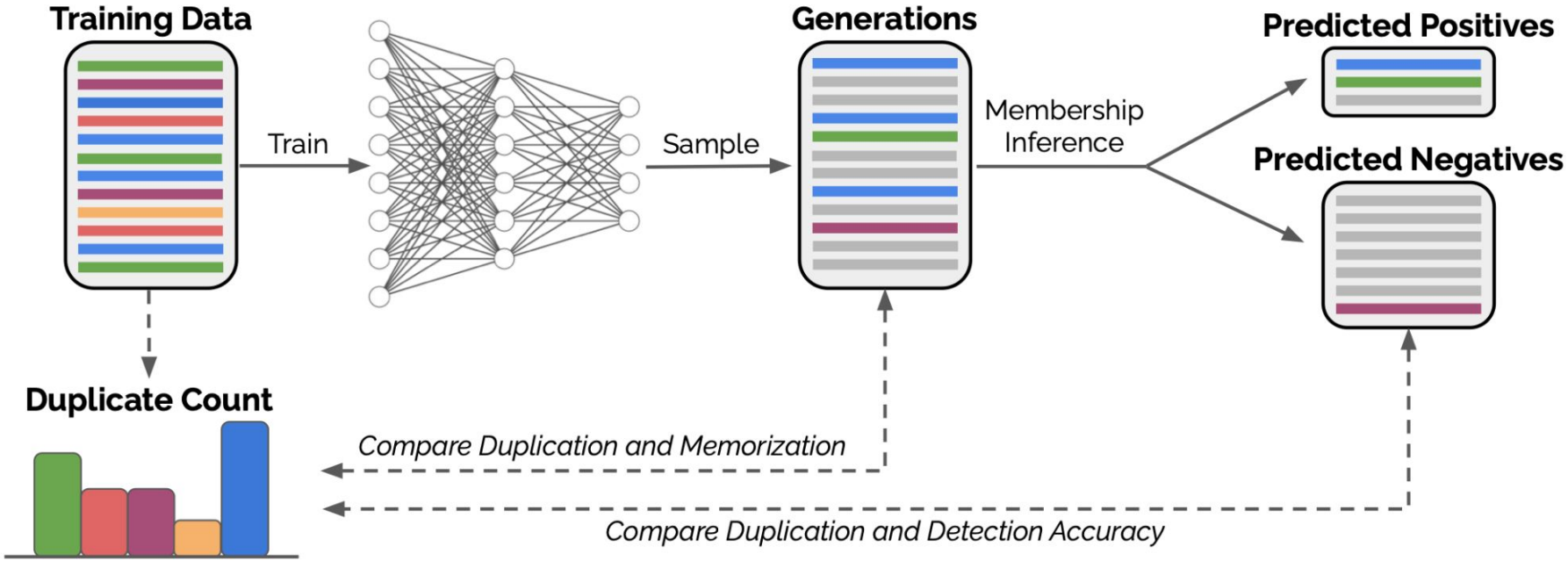
Xinchen Yang

Previous Work

Deduplicating Training Data Mitigates Privacy Risks in Language Models

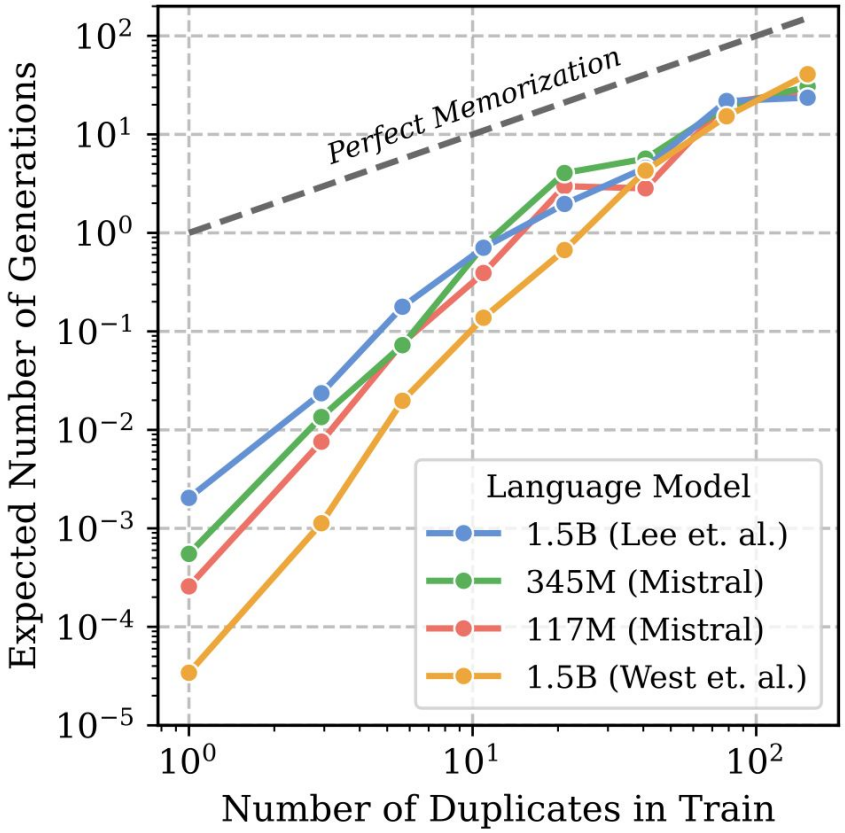
Nikhil Kandpal [1] **Eric Wallace** [2] **Colin Raffel** [1]

Analysis Pipeline



Results

- Size(Train) = Size(Samples)
- All LMs tested show a **superlinear increase** in the expected number of generations (slopes > 1 on a log-log plot).
- **Training samples** that are **not duplicated** are very **rarely generated**.
- Samples that are duplicated multiple times appear dramatically more frequently.



Connection with Current Paper

- The previous paper (Kandpal et al., 2022) reveals that **deduplicating training data can mitigate memorization**.
- However, this is complicated by the scale of web data and the prevalence of near-duplicated versions of many texts.
- The current paper introduces a new loss function, called the **goldfish loss**, where a **random subset of tokens are excluded from the loss computation**. It is conceptually different from the deduplication approach and saves worries about the complexity of the training data itself.

Subsequent Work

Strong Copyright Protection for Language Models via Adaptive Model Fusion

Javier Abad [*1] **Konstantin Donhauser** [*1] **Francesco Pinto** [2] **Fanny Yang** [1]

Connection with Current Paper

- The subsequent paper introduces **Copyright-Protecting Fusion** (CP-Fuse), which adaptively combines LMs to minimize the regurgitation of protected materials.
- The subsequent paper cites the current paper in its “Related works” part, indicating that **heuristic alternatives such as the goldfish loss** have proven effective in copyright protection.
- The subsequent paper suggests future research **evaluate CP-Fuse as a wrapper for** mitigation methods such as the **goldfish loss**, etc..

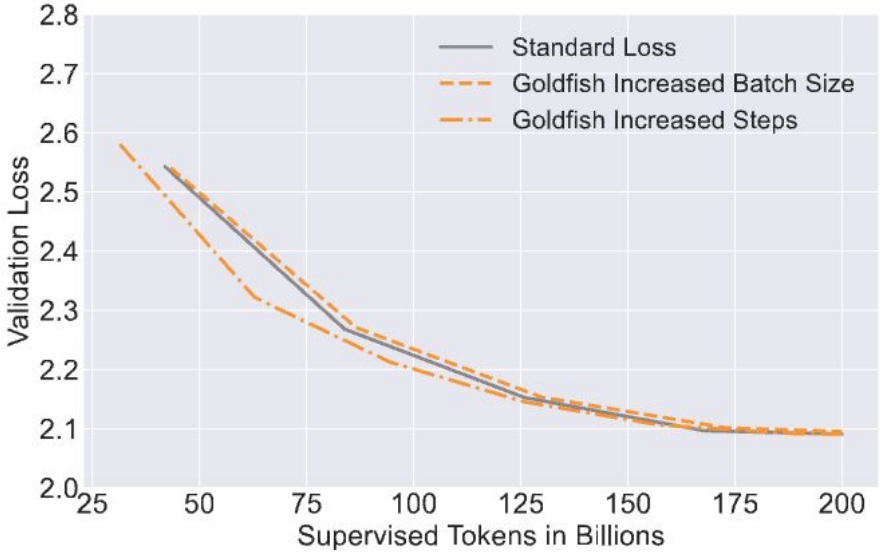
Academic Researcher

Sean McLeish

Academic Researcher

- Currently, goldfish requires more compute to achieve the same loss

- What if we are allowed to train fully on some data (e.g. not copyrighted)



Academic Researcher

1. Pre-hash all possibly copyrighted data
   a. More fine grained than deduplication
2. Apply goldfish loss (to all data) only using these pre-computed hashes

- Is less compute required?
- Does the goldfish anti-memorisation still hold?
- Could add hashes of copyrighted data which you don't know if it is in the training set

Detective Amisha Bhaskar on the Case

Investigation Report: The Goldfish Case – Abhimanyu Hans

The Subject

- A graduate student at the University of Maryland.
- Involved in high-profile AI research under the watchful eye of Prof. Tom Goldstein.
- Interest in securing AI systems and eliminating memorization risks.
- Finished his undergrad at University of Delhi with major in Mathematics.

The Crime Scene: Memorization in LLMs

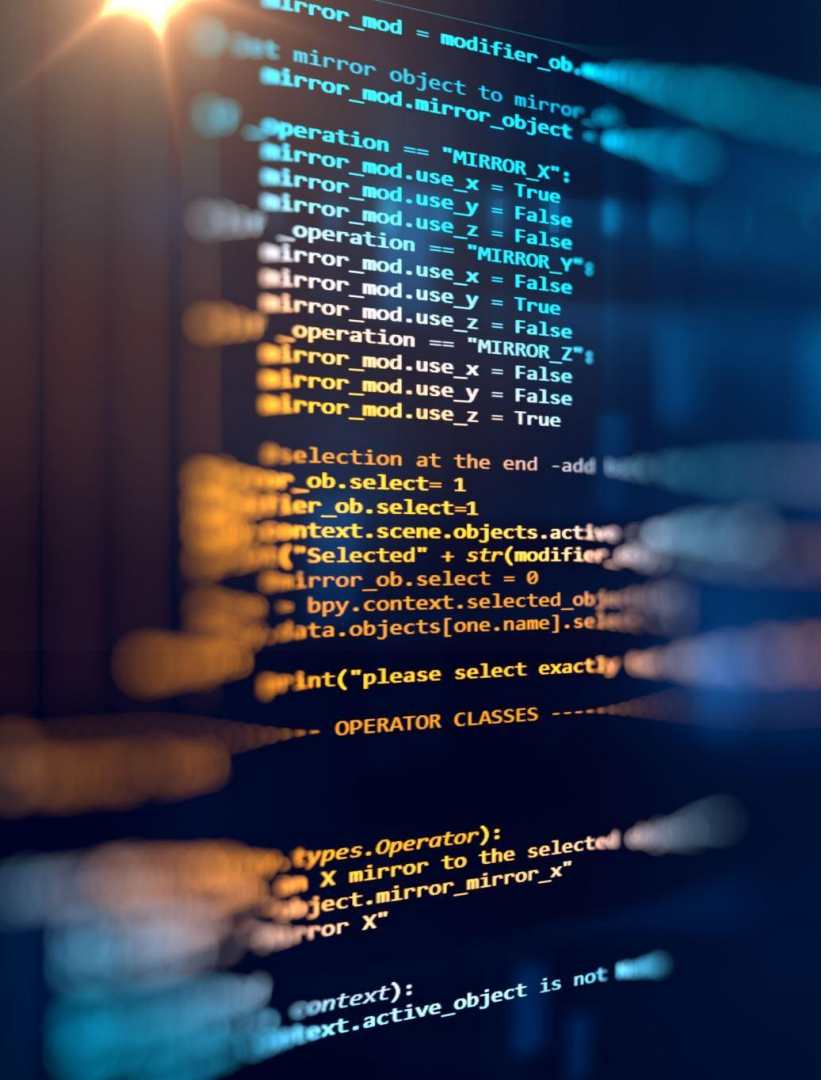
- Language models are unintentionally storing and reproducing sensitive information.
- This memorization poses risks: privacy violations, copyright infringement.
- The suspect’s mission: stop models from regurgitating confidential data.

Suspect's Profile: What Led Him to This?

- Years of working in security-sensitive environments (PayPal, HDFC Bank).
- Recognized the flaws in model training where sensitive data could be leaked.
- In academia, he joins forces with experts (Prof. Tom Goldstein, his supervisor) in AI security to tackle this threat.

Active Twitter Presence

- **Active Since August 2021**
- His Twitter activity helps him connect with both academic researchers and industry professionals, fostering collaboration and knowledge sharing.



Invention of the Year Award - UMD

- Abhimanyu's breakthrough work on zero-shot detection of machine-generated text earned him and his team the prestigious **Invention of the Year Award** at UMD.
- The project was developed in collaboration with Prof. Tom Goldstein and other researchers.
- **Impact**: This invention tackles a crucial problem—detecting whether text is human- or machine-generated, with an accuracy rate nearing 90%.

Private Investigator

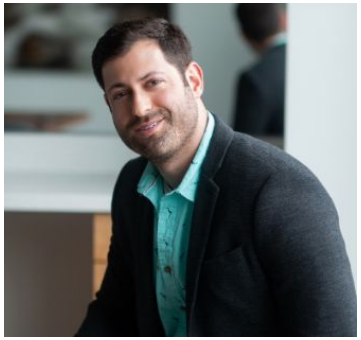
Dinithi Wickramaratne

First Author - Abhimanyu Hans

- Also taking this course and in the panel today!!
- MS in Computer Science - University of Maryland
- BS Mathematics - University of Delhi
- Research Interests - ensuring the security, efficiency and robustness of generative models. memorization and privacy related problems around language models.

Team Tom Goldstein

- Abhimanyu Hans
- Yuxin Wen
- John Kirchenbauer
- Hamid Kazemi
- Gowthami Somepalli
- Jonas Geiping



- Professor of Computer Science at University of Maryland
- Research areas: machine learning and optimization (applications in computer vision signal processing)

Team Abhinav Bhatele

- Prajwal Singhania
- Siddharth Singh

- Associate Professor of Computer Science at University of Maryland
- Research interests: systems and networks, with a focus on parallel computing and large-scale data analytics

Industry Practitioner

Taewon Kang

Advantages

- **Privacy and Copyright Protection:** The algorithm prevents the model from memorizing parts of its training data, which helps to avoid the reproduction of sensitive information (e.g., personal data or copyrighted text). This reduces the risks associated with privacy and copyright violations when language models generate text.

- **Minimal Performance Degradation:** Goldfish Loss is designed to have little to no impact on the model's performance. This means it allows for limiting memorization while still maintaining the overall effectiveness of the model on downstream tasks, making it a more efficient solution compared to other privacy-preserving methods.

- **Improved Storage and Processing Efficiency:** Since the model does not need to memorize unnecessary portions of data, it becomes less complex and more efficient in terms of memory usage. This can lead to reduced processing time and greater computational efficiency, as the model can focus on more relevant parts of the data.

- **Enhanced Security During Training and Inference:** By preventing excessive memorization of the training data, the algorithm minimizes the risk of sensitive data being exposed during both the training process and when the model is deployed in real-world applications. This makes it suitable for use in contexts that handle sensitive or private data, such as corporations or public institutions.

Disadvantages

- **Potential Loss of Useful Information:** Since some tokens are randomly excluded from the loss calculation, there is a risk that the model may fail to memorize important information. This can particularly affect the model if the excluded data turns out to be significant for understanding or predicting certain tasks.

- **Limitations with Complex Data:** While Goldfish Loss may work well for standard text data, it could be less effective for complex or specialized data (e.g., legal or technical documents), where certain excluded words might carry more weight. The model may struggle to provide deep or accurate responses if key terms are omitted from memorization.

- **Limited Scope of Application:** This technique is primarily suitable for language models, and it might not be as effective when applied to other types of data, such as unstructured data or images. These other types of data may require different approaches to prevent memorization and leakage of sensitive information.

- **Difficulty in Measuring Effectiveness:** It can be challenging to quantitatively evaluate the impact of Goldfish Loss. Since it randomly drops some tokens, it is hard to determine exactly which information is lost and which is retained, making it difficult to assess whether the technique is always beneficial in specific use cases.

Should We Adopt MM in Generative LLMs

When to adopt?

- **Privacy-sensitive applications:** If your model is handling sensitive or personal data (e.g., medical records, legal documents, private conversations), adopting Goldfish Loss is highly beneficial to minimize the risk of the model memorizing and reproducing that data.
- **Copyright concerns:** When training on datasets containing copyrighted text, using Goldfish Loss can help reduce the chance of the model reproducing long, verbatim excerpts from those sources, mitigating legal risks.
- **Models used for public interaction:** If your model will interact with the public (e.g., chatbots, virtual assistants), applying Goldfish Loss can help protect user privacy and prevent the leakage of private information.
- **Compliance with data protection regulations:** If you need to comply with privacy regulations, Goldfish Loss can be a valuable tool to ensure your model doesn't retain sensitive information from its training data.

When not to adopt?

- **Models requiring exact recall of training data:** If the model needs to precisely recall and reproduce certain parts of the training data (e.g., models used for legal document retrieval, scientific papers), Goldfish Loss could hinder performance.
- **Training on specialized or structured data:** When working with complex or highly structured data (e.g., technical manuals, medical literature), random exclusion of tokens might cause the model to miss key information, reducing its accuracy.
- **When memorization is crucial:** In cases where the model's ability to memorize data is essential for performance (e.g., language translation or summarization of specific texts), Goldfish Loss may not be appropriate as it limits memorization.
- **Real-time, critical systems:** If you're deploying a model in real-time systems where every token counts for accurate decision-making (e.g., automated customer support), avoiding any randomness in token exclusion may be important for consistency.

Should We Adopt MM in Generative LLMs

In summary, **adopt** Goldfish Loss when privacy, copyright, or regulatory compliance is **critical**,

but **avoid** it when exact memorization or the full context of data is **essential** to the model's task.

Social Impact Assessor

Paul Zaidins

Author Self-Assessment

- Protects users from accidental copyright infringement
  - Only with respect to end use
- Partially mitigates the effectiveness of membership inference attacks
  - Not guaranteed and the authors make a point to advise against believing this provides absolute protection
- I largely agree with these positive impacts and have nothing further to say about positive impacts

The Negative Impact of Most AI Improvements

- A LM that performs better is more likely to be adopted
- Adoption of LMs has in general lead to a decrease in employment as companies prefer them over humans
  - Either cheaper to use or able to produce more (an AI can work 24/7)
- In general the newly unemployed are likely to suffer hardship

A More Specific Negative Impact

- The issue of copyright infringement upstream the end use is explicitly ignored
- The legality of training on copyrighted material is still in flux and various greatly between countries
- Goldfish loss may be used a legal shield to deflect from the central question of this legality as lawsuits emerge
- This would be along the lines of, we are not violating copyright as we have taken good faith efforts to ensure copyright righted material cannot be exact reproduced
- This would also apply to end use cases where the copyrighted material is not exactly reproduced but sufficiently similar to the original material

Social Impact Assessor

Abhimanyu Hans

Goldfish Impact

Positives

- Can safeguard both model and data owners from their data being regurgitated at inference time
- Model naturally acts as paraphraser for dataset with goldfish switched on during training
- This is works with data licenses that allows usage under certain conditions such as restrictions on reproduction of underlying data

A Common Misunderstanding

Two Problems

1. Using unlicensed / pirated data to train generative models
2. Producing training data during runtime (potentially breaching licence agreements in place, for eg. code licenses)

...are different.

Negatives

- IF the current legal setup only accepts verbatim memorizat
  proof-of-training
- THEN companies might use this on illegally acquired dataset and plead non-guilty based on technicality that it they didn't regenerate the dataset.

← LEGAL BOTTLENECK

OpenAI vs NYT still pending in court