Revenue-Preserving Unlearning for Recommendation Systems

Ahson Saiyed[1] **Nishitha Khasnavis**[1] **Sam Levy**[2] **Chirag Agarwal**[1]

[1]School of Data Science, UVA
[2]Darden School of Business, UVA

2026 FAIR Symposium

Imagine. . . You're a Customer

You're shopping on Amazon:

- You bought that *embarrassing* gift for your weird uncle

Imagine... You're a Customer

You're shopping on Amazon:

- You bought that *embarrassing* gift for your weird uncle
- Now your homepage is... concerning

Imagine. . . You're a Customer

You're shopping on Amazon:

- You bought that *embarrassing* gift for your weird uncle
- Now your homepage is... concerning
- You see a page: **"Improve Your Recommendations"**

Imagine. . . You're a Customer

You're shopping on Amazon:

- You bought that *embarrassing* gift for your weird uncle
- Now your homepage is... concerning
- You see a page: **"Improve Your Recommendations"**

What would you do?

- Click "Delete this from my history"?

Imagine. . . You're a Customer

You're shopping on Amazon:

- You bought that *embarrassing* gift for your weird uncle
- Now your homepage is... concerning
- You see a page: **"Improve Your Recommendations"**

What would you do?

- Click "Delete this from my history"?
- Hope Amazon *actually* forgets?

Imagine... You're a Customer

You're shopping on Amazon:

- You bought that *embarrassing* gift for your weird uncle
- Now your homepage is... concerning
- You see a page: **"Improve Your Recommendations"**

What would you do?

- Click "Delete this from my history"?
- Hope Amazon *actually* forgets?
- Wonder: *"Will this really work?"*

Imagine... You're a Customer

You're shopping on Amazon:

- You bought that *embarrassing* gift for your weird uncle
- Now your homepage is... concerning
- You see a page: **"Improve Your Recommendations"**

What would you do?

- Click "Delete this from my history"?
- Hope Amazon *actually* forgets?
- Wonder: *"Will this really work?"*

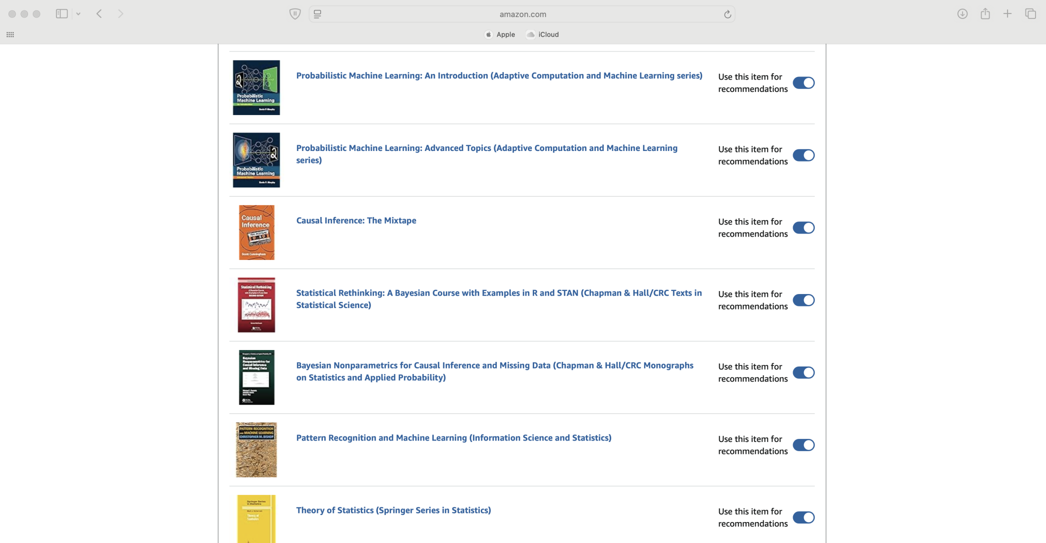
The Trust Problem

Can you verify Amazon actually unlearned your preferences?

$\Downarrow$

Transparency is essential for consumer trust

Amazon: “Improve Your Recommendations”



Consumers Are Paying Attention

What the surveys say:

- **81%** of consumers suspect companies train AI on their data *without consent*

Consumers Are Paying Attention

What the surveys say:

- **81%** of consumers suspect companies train AI on their data *without consent*
- **84%** would stop using companies that are opaque about AI data usage

Consumers Are Paying Attention

What the surveys say:

- **81%** of consumers suspect companies train AI on their data *without consent*
- **84%** would stop using companies that are opaque about AI data usage
- **76%** would switch to a competitor offering more transparency—even if it costs more!

Consumers Are Paying Attention

What the surveys say:

- **81%** of consumers suspect companies train AI on their data *without consent*
- **84%** would stop using companies that are opaque about AI data usage
- **76%** would switch to a competitor offering more transparency—even if it costs more!

European consumers agree:

- **59%** uncomfortable with personal data training AI

Consumers Are Paying Attention

What the surveys say:

- **81%** of consumers suspect companies train AI on their data *without consent*
- **84%** would stop using companies that are opaque about AI data usage
- **76%** would switch to a competitor offering more transparency—even if it costs more!

European consumers agree:

- **59%** uncomfortable with personal data training AI
- **62%** feel they've "become the product"

Consumers Are Paying Attention

What the surveys say:

- **81%** of consumers suspect companies train AI on their data *without consent*
- **84%** would stop using companies that are opaque about AI data usage
- **76%** would switch to a competitor offering more transparency—even if it costs more!

European consumers agree:

- **59%** uncomfortable with personal data training AI
- **62%** feel they've "become the product"

The Bottom Line

Consumers want to hear:

"Yes, we can delete you from our model"

...and then **actually do it**.

Sources: Relyance AI Consumer Survey 2025; Usercentrics State of Digital Trust 2025

Now, imagine... You're a Business Owner

You run a recommendation system:

- Millions of users, billions of interactions

What Would You Do?

Option A: Retrain from scratch
(Expensive, slow)

Option B: "Surgically" remove the poisoned data
(Fast, but does it work?)

$\Downarrow$

Machine Unlearning

Now, imagine... You're a Business Owner

You run a recommendation system:

- Millions of users, billions of interactions
- Revenue depends on recommendation quality

What Would You Do?

Option A: Retrain from scratch
(Expensive, slow)

Option B: "Surgically" remove the poisoned data
(Fast, but does it work?)

$\Downarrow$

Machine Unlearning

Now, imagine. . . You're a Business Owner

You run a recommendation system:

- Millions of users, billions of interactions
- Revenue depends on recommendation quality
- Then you discover: **data poisoning attack!**

What Would You Do?

Option A: Retrain from scratch
(Expensive, slow)

Option B: "Surgically" remove the poisoned data
(Fast, but does it work?)

$\Downarrow$

Machine Unlearning

Now, imagine... You're a Business Owner

You run a recommendation system:

- Millions of users, billions of interactions
- Revenue depends on recommendation quality
- Then you discover: **data poisoning attack!**

The poisoning problem is real:

- Injecting just **0.5%** fake users can make a target item appear in recommendations for **150$\times$ more users**

What Would You Do?

Option A: Retrain from scratch
(Expensive, slow)

Option B: "Surgically" remove the poisoned data
(Fast, but does it work?)

$\Downarrow$

Machine Unlearning

Now, imagine. . . You're a Business Owner

You run a recommendation system:

- Millions of users, billions of interactions
- Revenue depends on recommendation quality
- Then you discover: **data poisoning attack!**

The poisoning problem is real:

- Injecting just **0.5%** fake users can make a target item appear in recommendations for **150× more users**
- Bad actors can manipulate your system

What Would You Do?

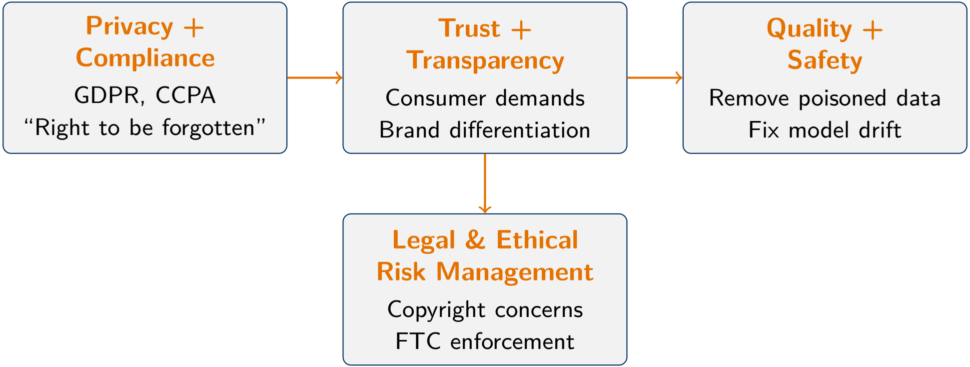
Option A: Retrain from scratch
(Expensive, slow)

Option B: "Surgically" remove the poisoned data
(Fast, but does it work?)

$\Downarrow$

Machine Unlearning

Why Machine Unlearning is Increasingly Important



The Big Picture

All of this is about **Ethical AI**. Machine unlearning is a *promising* and *necessary* field.

The Scale of the Problem

Demand is exploding:

- Amazon: **5.6 million** deletion requests in 2024

The Scale of the Problem

Demand is exploding:

- Amazon: **5.6 million** deletion requests in 2024
- **246%** increase in data privacy requests since 2021

The Scale of the Problem

Demand is exploding:

- Amazon: **5.6 million** deletion requests in 2024
- **246%** increase in data privacy requests since 2021
- E-commerce leads: **1,577 DSRs** per million identities

The Scale of the Problem

Demand is exploding:

- Amazon: **5.6 million** deletion requests in 2024
- **246%** increase in data privacy requests since 2021
- E-commerce leads: **1,577 DSRs** per million identities

Costs are skyrocketing:

- Single deletion request: ∼**$1,524**

The Scale of the Problem

Demand is exploding:

- Amazon: **5.6 million** deletion requests in 2024
- **246%** increase in data privacy requests since 2021
- E-commerce leads: **1,577 DSRs** per million identities

Costs are skyrocketing:

- Single deletion request: ~**$1,524**
- Manual processing: **$881K/year** per million DSRs

The Scale of the Problem

Demand is exploding:

- Amazon: **5.6 million** deletion requests in 2024
- **246%** increase in data privacy requests since 2021
- E-commerce leads: **1,577 DSRs** per million identities

Costs are skyrocketing:

- Single deletion request: ~**$1,524**
- Manual processing: **$881K/year** per million DSRs
- Businesses spending **36% more** year-over-year

The Scale of the Problem

Demand is exploding:

- Amazon: **5.6 million** deletion requests in 2024
- **246%** increase in data privacy requests since 2021
- E-commerce leads: **1,577 DSRs** per million identities

Costs are skyrocketing:

- Single deletion request: ~**$1,524**
- Manual processing: **$881K/year** per million DSRs
- Businesses spending **36% more** year-over-year

Regulators are watching:

- FTC's "algorithmic disgorgement"

The Scale of the Problem

Demand is exploding:

- Amazon: **5.6 million** deletion requests in 2024
- **246%** increase in data privacy requests since 2021
- E-commerce leads: **1,577 DSRs** per million identities

Costs are skyrocketing:

- Single deletion request: ~**$1,524**
- Manual processing: **$881K/year** per million DSRs
- Businesses spending **36% more** year-over-year

Regulators are watching:

- FTC's "algorithmic disgorgement"
- Everalbum (2021), WW/Kurbo (2022), Rite Aid (2023), Amazon (2023)

The Scale of the Problem

Demand is exploding:

- Amazon: **5.6 million** deletion requests in 2024
- **246%** increase in data privacy requests since 2021
- E-commerce leads: **1,577 DSRs** per million identities

Costs are skyrocketing:

- Single deletion request: ~**$1,524**
- Manual processing: **$881K/year** per million DSRs
- Businesses spending **36% more** year-over-year

Regulators are watching:

- FTC's "algorithmic disgorgement"
- Everalbum (2021), WW/Kurbo (2022), Rite Aid (2023), Amazon (2023)

FTC's Message

"These AI tools are novel, but they are **not exempt from existing rules**"

— Chair Lina Khan

OK Great. . . Businesses Just Have to Unlearn My Data!

Easy enough, right?

OK Great... Businesses Just Have to Unlearn My Data!

Easy enough, right?

Nope. It's hard. Here's why.

The Technical Challenge

- Information is **distributed** across all weights
- No clean mapping: data $\rightarrow$ parameters
- Knowledge is "holographic, not indexed"
- The "Spider-Man problem": concepts are *entangled*

Yet Big Tech is Starting

- Google: Machine Unlearning Challenge
- IBM: SPUNGE framework (224 sec vs months)
- Microsoft, Apple: Data deletion policies
- But: **No native unlearning APIs yet**

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓
- But model *memorized* patterns

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓
- But model *memorized* patterns
- Deleted info remains **reconstructible** from embeddings

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓
- But model *memorized* patterns
- Deleted info remains **reconstructible** from embeddings
- Microsoft admits: "deleting data is insufficient by itself"

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓
- But model *memorized* patterns
- Deleted info remains **reconstructible** from embeddings
- Microsoft admits: "deleting data is insufficient by itself"

The Gap

90% of consumers believe they should be able to delete data from AI

But once data is ingested, users may have **no option of ever having it removed**

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓
- But model *memorized* patterns
- Deleted info remains **reconstructible** from embeddings
- Microsoft admits: "deleting data is insufficient by itself"

The Gap

90% of consumers believe they should be able to delete data from AI

But once data is ingested, users may have **no option of ever having it removed**

Current approaches:

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓
- But model *memorized* patterns
- Deleted info remains **reconstructible** from embeddings
- Microsoft admits: "deleting data is insufficient by itself"

The Gap

90% of consumers believe they should be able to delete data from AI

But once data is ingested, users may have **no option of ever having it removed**

Current approaches:

SISA (Exact Unlearning):

- Shard data, train sub-models

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓
- But model *memorized* patterns
- Deleted info remains **reconstructible** from embeddings
- Microsoft admits: "deleting data is insufficient by itself"

The Gap

90% of consumers believe they should be able to delete data from AI

But once data is ingested, users may have **no option of ever having it removed**

Current approaches:

SISA (Exact Unlearning):

- Shard data, train sub-models
- Retrain only affected shard

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓
- But model *memorized* patterns
- Deleted info remains **reconstructible** from embeddings
- Microsoft admits: "deleting data is insufficient by itself"

The Gap

90% of consumers believe they should be able to delete data from AI

But once data is ingested, users may have **no option of ever having it removed**

Current approaches:

SISA (Exact Unlearning):

- Shard data, train sub-models
- Retrain only affected shard
- Problem: Breaks collaborative signals

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓
- But model *memorized* patterns
- Deleted info remains **reconstructible** from embeddings
- Microsoft admits: "deleting data is insufficient by itself"

The Gap

90% of consumers believe they should be able to delete data from AI

But once data is ingested, users may have **no option of ever having it removed**

Current approaches:

SISA (Exact Unlearning):

- Shard data, train sub-models
- Retrain only affected shard
- Problem: Breaks collaborative signals

RecEraser (For RecSys):

- Balanced partitioning (not random)

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓
- But model *memorized* patterns
- Deleted info remains **reconstructible** from embeddings
- Microsoft admits: "deleting data is insufficient by itself"

The Gap

90% of consumers believe they should be able to delete data from AI

But once data is ingested, users may have **no option of ever having it removed**

Current approaches:

SISA (Exact Unlearning):

- Shard data, train sub-models
- Retrain only affected shard
- Problem: Breaks collaborative signals

RecEraser (For RecSys):

- Balanced partitioning (not random)
- Preserves user-item clustering

How is Unlearning Done? (And Why Isn't Deletion Enough?)

Why deletion isn't enough:

- Delete data from database: ✓
- But model *memorized* patterns
- Deleted info remains **reconstructible** from embeddings
- Microsoft admits: "deleting data is insufficient by itself"

The Gap

90% of consumers believe they should be able to delete data from AI

But once data is ingested, users may have **no option of ever having it removed**

Current approaches:

SISA (Exact Unlearning):

- Shard data, train sub-models
- Retrain only affected shard
- Problem: Breaks collaborative signals

RecEraser (For RecSys):

- Balanced partitioning (not random)
- Preserves user-item clustering
- 10–27$\times$ faster than full retrain

Research Questions

RQ1 **What is the true economic cost** of unlearning specific items or patterns?

Research Questions

RQ1 What is the true economic cost of unlearning specific items or patterns?

- Metrics: Revenue Differential
- "Fatal flaw": Current efficient methods may have hidden costs

Research Questions

RQ1 **What is the true economic cost** of unlearning specific items or patterns?

- ▷ Metrics: Revenue Differential
- ▷ "Fatal flaw": Current efficient methods may have hidden costs

RQ2 To what extent do current unlearning practices **leave latent representations** that violate user privacy?

Research Questions

RQ1 **What is the true economic cost** of unlearning specific items or patterns?

- Metrics: Revenue Differential
- "Fatal flaw": Current efficient methods may have hidden costs

RQ2 To what extent do current unlearning practices **leave latent representations** that violate user privacy?

- Metrics: Membership Inference, Concept Leakage
- "Fatal flaw": Baselines leave traces of user information

Research Questions

RQ1 **What is the true economic cost** of unlearning specific items or patterns?

- Metrics: Revenue Differential
- "Fatal flaw": Current efficient methods may have hidden costs

RQ2 To what extent do current unlearning practices **leave latent representations** that violate user privacy?

- Metrics: Membership Inference, Concept Leakage
- "Fatal flaw": Baselines leave traces of user information

RQ3 Does unlearning impact **differ by customer segment**?

Research Questions

RQ1 **What is the true economic cost** of unlearning specific items or patterns?

- ▷ Metrics: Revenue Differential
- ▷ "Fatal flaw": Current efficient methods may have hidden costs

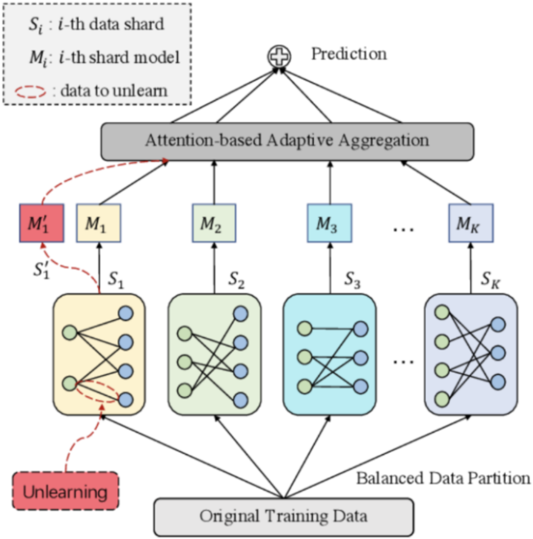
RQ2 To what extent do current unlearning practices **leave latent representations** that violate user privacy?

- ▷ Metrics: Membership Inference, Concept Leakage
- ▷ "Fatal flaw": Baselines leave traces of user information

RQ3 Does unlearning impact **differ by customer segment**?

- ▷ High-value customers vs. casual browsers
- ▷ Equity implications of unlearning

SISA and RecEraser



SISA vs RecEraser: A Quick Comparison

SISA		RecEraser
Random sharding → Breaks collaborative graph		Balanced partitioning → Preserves local signals
Simple majority voting → Accuracy degradation	VS	Attention-based aggregation → Maintains accuracy
Multiple models + checkpoints → Storage overhead		Same shard retraining → 10–27× speedup

Key Limitation

Both SISA and RecEraser assume user influence can be isolated, which does not hold in dense, sequential recommender data.

Case Study: ComScore Behavioral Data

Why ComScore?

- Large-scale, real-world behavioral data
- Full search-to-purchase funnel
- Captures **temporal sequences**, critical for understanding how influence accumulates

Coverage:

- Search queries, clicks, URLs visited
- Products, categories, metadata
- Demographics (age, gender, household)
- Timestamps at per-second granularity

Data Summary

Metric	Value
Total Events	635K
Unique Users	45K
Unique Products	348K
Checkout Sessions	214K
Total Revenue	$709M

Top Domains

amazon.com, walmart.com, dominos.com, etsy.com, target.com

Why ComScore is Uniquely Suited for Unlearning Research

Scale

111M searches
40K items
Multi-platform

Full Pipeline

Search → Click →
Browse → Purchase
Causal chain preserved

Temporal

Per-second granularity
Session reconstruction
Query reformulations

Key Insight for Unlearning

User influence is **not instantaneous**, it accumulates over time.

Removing a user requires removing a **temporally ordered chain of influence**, not a single data point.

What is the True Cost of Unlearning?

The Profit Score Framework:

$$\text{Profit} = \alpha \cdot \text{Utility} - \beta \cdot \text{Compute} - \gamma \cdot \text{Leakage}$$

- **Utility Retention:** NDCG@K, Recall@K
- **Compute Cost:** GPU hours, retraining time
- **Leakage Risk:** Membership inference attack success

Our Hypothesis:

Current "efficient" methods have **hidden costs**—lower compute, but higher revenue impact!

The Misalignment Problem

Existing methods optimize for:

Speed or **Privacy**

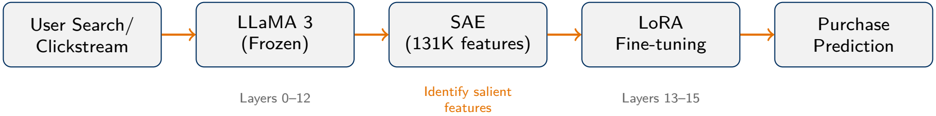
But businesses care about:

Revenue

These objectives are **not aligned**

Our Approach: SAE-Driven Revenue-Aware Unlearning

Key Idea: Use **Sparse Autoencoders** to identify and suppress user-specific features while preserving revenue-generating capabilities.



What Are Sparse Autoencoders?

The Interpretability Problem:

- Neural networks: "black boxes"
- Information distributed across millions of parameters
- Hard to identify *what* the model knows about *whom*

SAEs to the Rescue:

- Learn a **sparse, interpretable** representation
- Each feature corresponds to a "concept"
- Can identify which features activate for specific users

How It Works

1. Encoder: $\mathbf{h} \rightarrow \mathbf{z}$ (sparse)
2. Decoder: $\mathbf{z} \rightarrow \hat{\mathbf{h}}$ (reconstruct)
3. Sparsity: Most $z_i = 0$
4. Interpret: Active z_i = concepts

Our SAE:
131,072 features at layer 12/14

Unlearning with SAEs

Step 1: Contrastive Feature Selection

- Compare activations: **Forget users** vs. **Retain users**

Unlearning with SAEs

Step 1: Contrastive Feature Selection

- Compare activations: **Forget users** vs. **Retain users**
- Identify "salient" features: Active more often for forget users

Unlearning with SAEs

Step 1: Contrastive Feature Selection

- Compare activations: **Forget users** vs. **Retain users**
- Identify "salient" features: Active more often for forget users
- These are the features we need to suppress

Unlearning with SAEs

Step 1: Contrastive Feature Selection

- Compare activations: **Forget users** vs. **Retain users**
- Identify "salient" features: Active more often for forget users
- These are the features we need to suppress

Step 2: Model Optimization

Unlearning with SAEs

Step 1: Contrastive Feature Selection

- Compare activations: **Forget users** vs. **Retain users**
- Identify "salient" features: Active more often for forget users
- These are the features we need to suppress

Step 2: Model Optimization

$$\mathcal{L}_{\text{total}} = \underbrace{\mathcal{L}_{\text{unlearn}}}_{\text{Suppress forget features}} + \underbrace{\mathcal{L}_{\text{retain}}}_{\text{Preserve other users}} + \underbrace{\mathcal{L}_{\text{coherence}}}_{\text{Maintain fluency}}$$

Unlearning with SAEs

Step 1: Contrastive Feature Selection

- Compare activations: **Forget users** vs. **Retain users**
- Identify "salient" features: Active more often for forget users
- These are the features we need to suppress

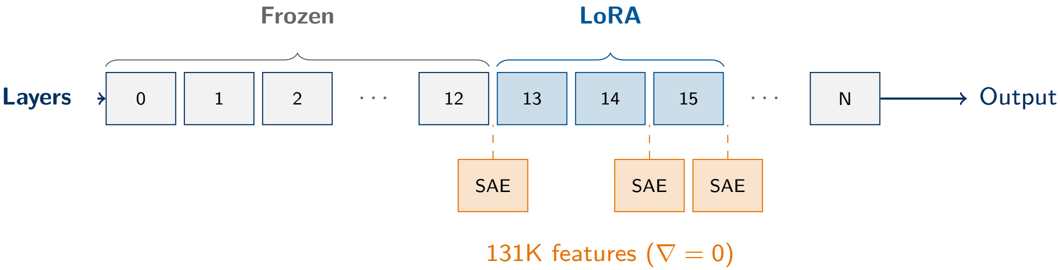
Step 2: Model Optimization

$$\mathcal{L}_{\text{total}} = \underbrace{\mathcal{L}_{\text{unlearn}}}_{\text{Suppress forget features}} + \underbrace{\mathcal{L}_{\text{retain}}}_{\text{Preserve other users}} + \underbrace{\mathcal{L}_{\text{coherence}}}_{\text{Maintain fluency}}$$

Key Advantage

- **Persistent:** Modifies model parameters, not just runtime behavior
- **Interpretable:** Can inspect *which* concepts were unlearned
- **Modular:** Uses LoRA—if unlearning causes issues, adapter can be disabled

Our Framework



Frozen Layers

Preserve general knowledge

LoRA Layers

Learn to suppress user-specific features

SAEs

Identify which features to target

Preliminary Results

Experimental Setup:

- Base Model: Llama-3.2-1B (GLoSS)
- Retain set: 8,249 records (6,143 users)
- Forget set: 1,181 records (1,080 users)

Results After 5 Epochs

Metric	**Epoch 1**	**Epoch 5**	Δ
$\mathcal{L}_{\text{privacy}}$	0.588	0.024	-96%
$\mathcal{L}_{\text{quality}}$	1.06	1.023	-3%
$\mathcal{L}_{\text{retain}}$	0.079	0.081	+2%
Salient Act.	0.536	0.016	-97%

Key Findings

- **Salient features suppressed:** 97% reduction
- **Prediction capability preserved:** Only 3% quality loss
- **Representations stable:** Retain set unaffected

Comparison with Baselines

	RecEraser (2-3h)			**SISA** (4-5h)			**Ours** (3-10m)		
	R@5	N@5	Rev.	R@5	N@5	Rev.	R@5	N@5	Rev.
Baseline	5.71	4.05	–	3.59	2.98	–	5.96	4.92	–
10% unlearn	3.89	2.20	92.96	1.93	1.62	20.12	**3.91**	**3.22**	56.88
20% unlearn	4.03	2.26	36.61	2.40	1.79	33.75	3.23	**2.42**	**71.49**

RecEraser

- Exact unlearning
- Retains 68% utility
- Requires retraining

SISA

- Exact unlearning
- Retains 54% utility
- Random sharding hurts

Ours

- Approx. unlearning
- Retains 66%, best NDCG
- No retraining needed

Summary & Next Steps

What We've Shown:

- Machine unlearning is a **business imperative**
- Current methods trade off between speed, privacy, and revenue
- SAE-based approach offers **interpretable** unlearning
- Preliminary results: 97% feature suppression with 3% quality loss

Next Steps:

- Full evaluation on ComScore dataset
- Revenue impact analysis by customer segment

The Big Picture

Privacy and **Revenue** can coexist.

With the right approach, businesses can:

✓ Respect user rights
✓ Maintain recommendation quality
✓ Preserve revenue

Thank You!

Questions?

Appendix

Appendix: Profit Score Framework Details

Utility Retention Metrics (Revenue Protection):

- NDCG@K, Recall@K, Hit@K
- Measures how well the model still recommends after unlearning

Computational & Operational Cost:

- Retraining time / GPU hours
- Latency per deletion request
- Energy cost for large-scale deletions

Leakage Risk:

- Membership Inference Attack (MIA) success rate
- Post-unlearning: should approach random chance ($\sim$50% AUC)
- Unlearning Accuracy (UA): Drop in accuracy on forget set

Appendix: ComScore Data Details

Data Tables:

- `comscore_search_fact`: Search phrases
- `comscore_url_traffic`: Domains visited
- `comscore_category_map`: Site categories
- `purchase_items`: Transaction records

Top Categories:

- Home & Living: 438K events
- Electronics & Computing: 20K events
- Apparel & Accessories: 17K events
- Books, Music & Video: 11K events

User Behavior:

- Avg actions per user: 14.12
- Most active user: 2,170 actions
- Avg basket size: 2.98 items
- Avg basket value: $3,317

Seasonality:

- Peak: Month 1 (78K events)
- Trough: Month 9 (39K events)
- Holiday uptick: Months 11–12

Appendix: Key Related Work

Paper	Year	Method
SISA (Bourtoule et al.)	2021	Sharded training
RecEraser (Chen et al.)	2022	Balanced partitioning for RecSys
UltraRE	2023	Error decomposition
CURE4Rec	2024	Benchmark for RecSys unlearning
CRISP	2025	SAE-based concept removal
SAE Subspace Projections	2025	SAE-guided parameter updates

SAE Unlearning Literature:

- "Applying Sparse Autoencoders to Unlearn Knowledge in Language Models" (2024)
- "Sparse-Autoencoder-Guided Internal Representation Unlearning for LLMs"
- "SAEs Can Improve Unlearning: Dynamic SAE Guardrails" (2025)

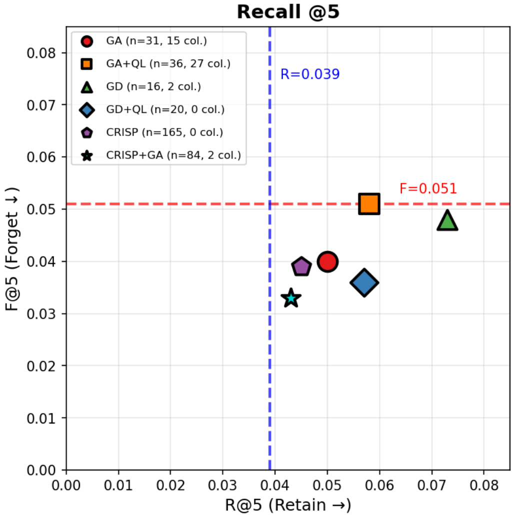
Appendix: Comparison with Approximate Unlearning Methods

Retain & Forget Performance

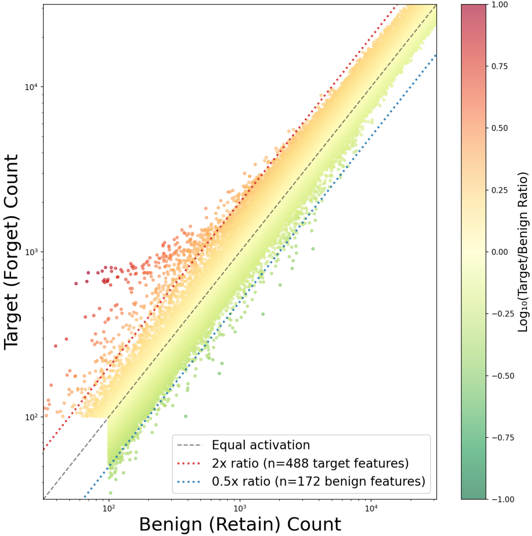
Method	@5 R	@5 F	@10 R	@10 F
GA	.050	.040	.055	.044
GA+QL	.058	.051	.062	.058
GD	.073	.048	.082	.054
GD+QL	.057	.036	.065	.042
CRISP	.045	.039	.053	.042
CRISP+GA	.043	**.033**	.050	**.037**
Baseline	.039	.051	.045	.057

Key Finding

CRISP+GA: Best forget rate (-**36.7%**) while retaining model utility (+12.0%)



Identifying Target-Salient Features via Sparse Autoencoders



SAE Feature Extraction:

1. Forward pass through LLaMA-3.2-1B
2. Capture MLP activations at Layer 10
3. Encode via EleutherAI SAE (131K dims)
4. Count feature activations per set

Feature Selection (CRISP)

Relative Activation Ratio:

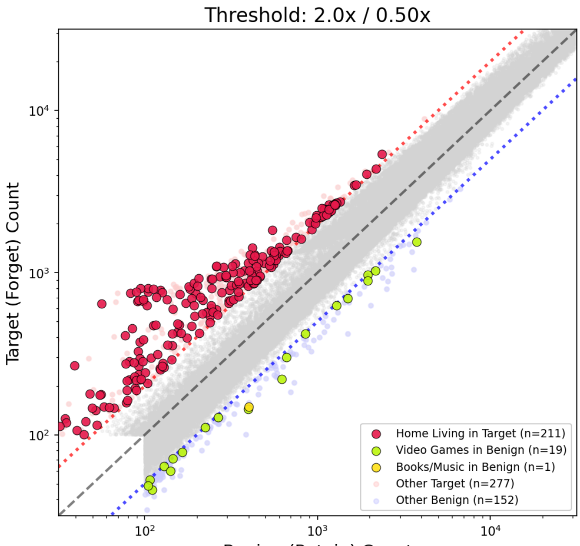
$$\rho(f_i) = \frac{A(f_i, D_{\text{target}})}{A(f_i, D_{\text{retain}}) + \epsilon}$$

Salient Features:

$$F_{\text{salient}} = \{f_i \in F_{\text{freq}} \mid \rho(f_i) \geq \tau\}$$

$\rho(f_i) > 2$ Target-salient (suppress)
$\rho(f_i) < 0.5$ Retain-salient (preserve)

Interpretable Feature Selection via Sparse Autoencoders



Why SAEs for Unlearning?

- Features map to **semantic concepts**
- Salient features cluster by category
- Enables **targeted suppression** of specific knowledge

Interpretability Advantage

Gradient-only:	Which weights changed?
SAE-guided:	Which concepts suppressed?

SAE adds **131K interpretable dimensions** to the optimization landscape for fine-grained control.

Comparison with Baselines-1

Condition	RecEraser		SISA	
	Recall@5	NDCG@5	Recall@5	NDCG@5
Baseline	5.71	4.05	3.59	2.98
10% unlearned	3.89	2.20	1.93	1.62
20% unlearned	4.03	2.26	2.40	1.79
30% unlearned	3.80	2.15	—	—

RecEraser Advantage

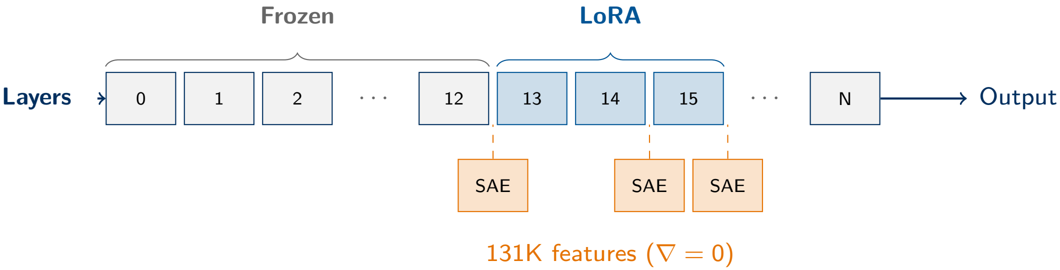
- Higher baseline performance
- Better retention after unlearning
- Balanced partitioning helps

SISA Weakness

- Random sharding hurts RecSys
- Larger performance drop
- Doesn't scale to LLMs well

Next: Compare SAE-based approach on same benchmarks

Our Framework



Frozen Layers

Preserve general knowledge

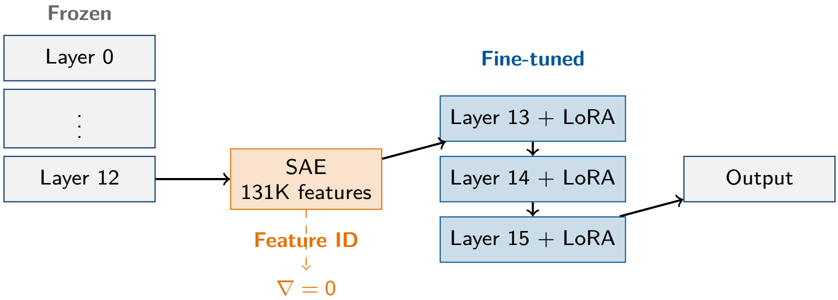
LoRA Layers

Learn to suppress user-specific features

SAEs

Identify which features to target

Our Framework



Frozen Layers

Layers 0–12: Preserve general knowledge

SAE

Identifies user-specific salient features

LoRA

Suppresses salient activations