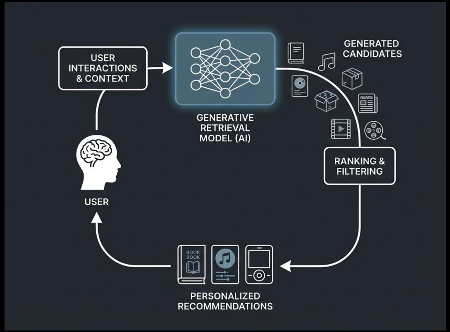
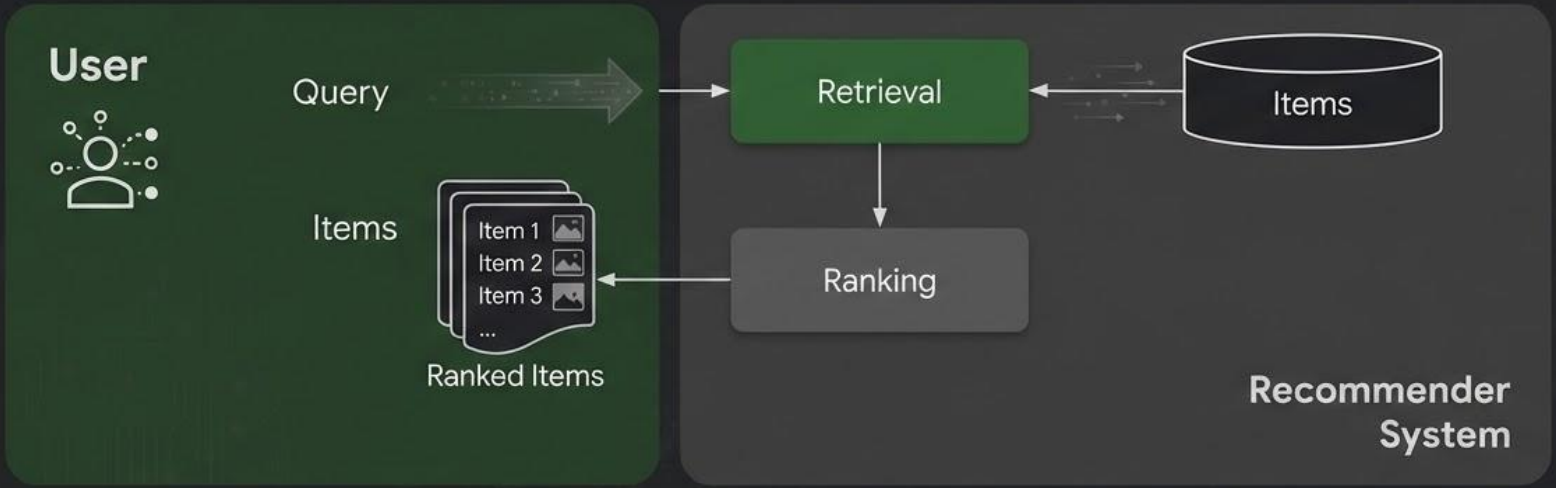
Personalized Recommendations in the era of GenAI



Nikhil Mehta | nikhilmehta@
Staff Research Scientist, Google DeepMind
Presenting work done in collaboration with GDM/YouTube.

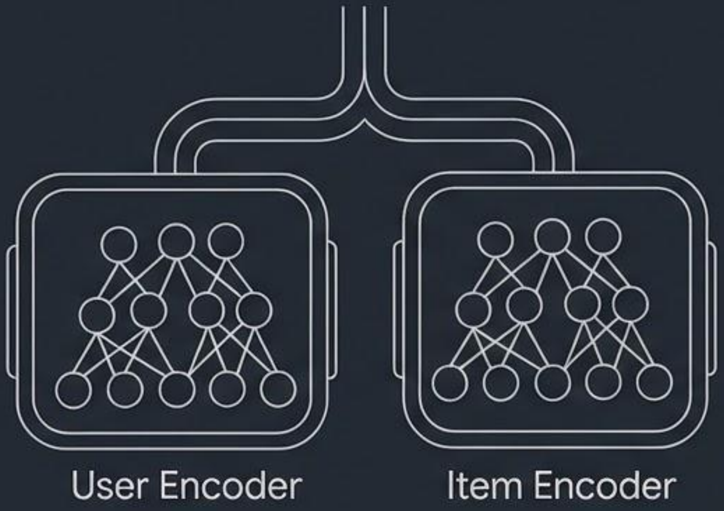
2026-01-23

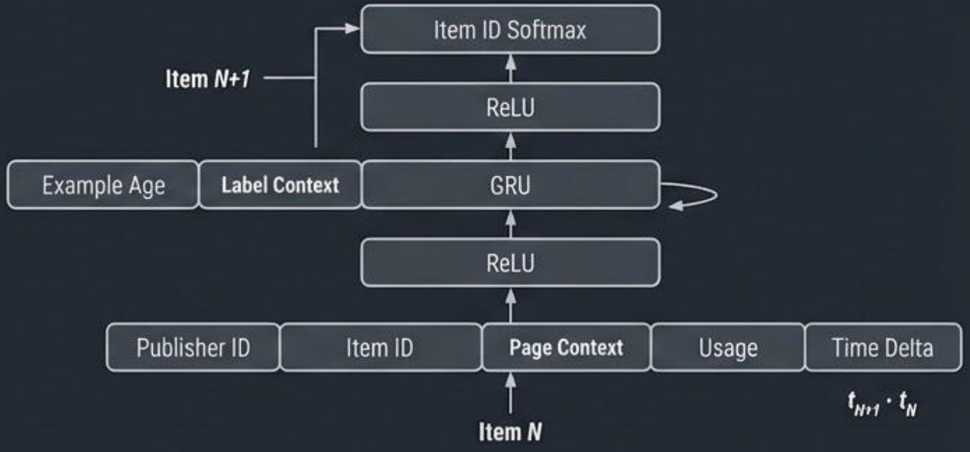
Typical Recommendation Pipeline



Query features: User + Contextual features.

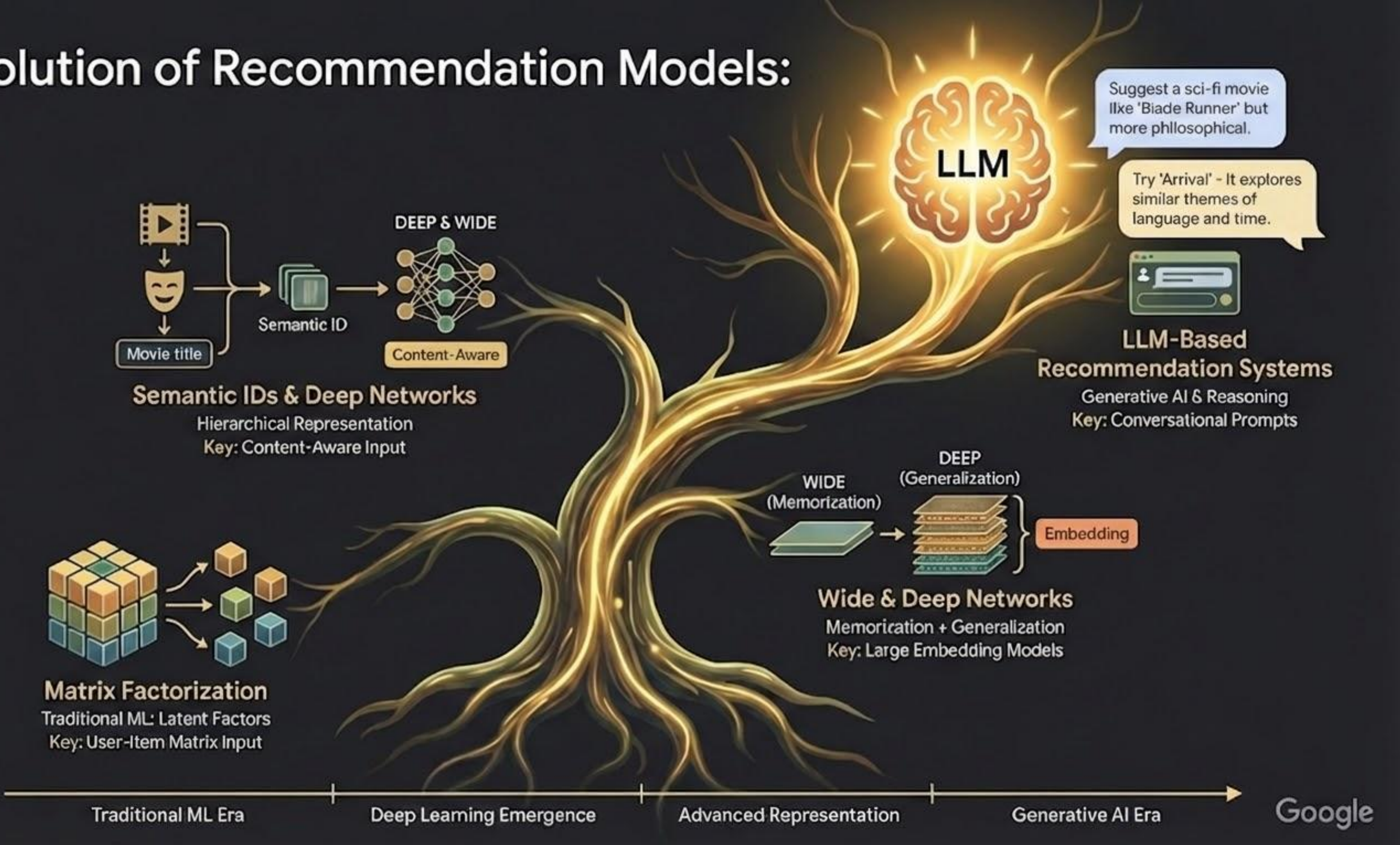
Traditional Retrieval Models...





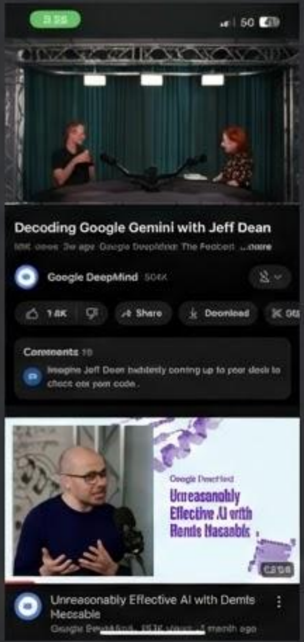
Google

Evolution of Recommendation Models:



Recommendations at YouTube

Home



WatchNext

Shorts

Search

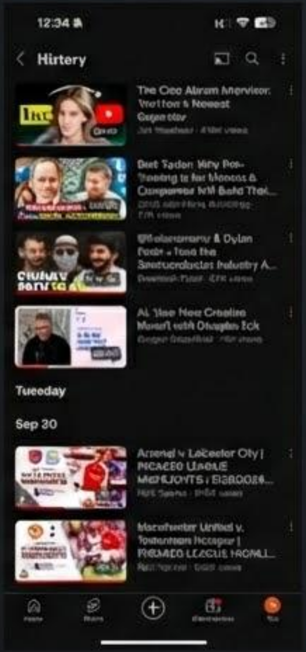
Others

Google

Personalized Recommendation

f (

,



) ≈

User Context Recs

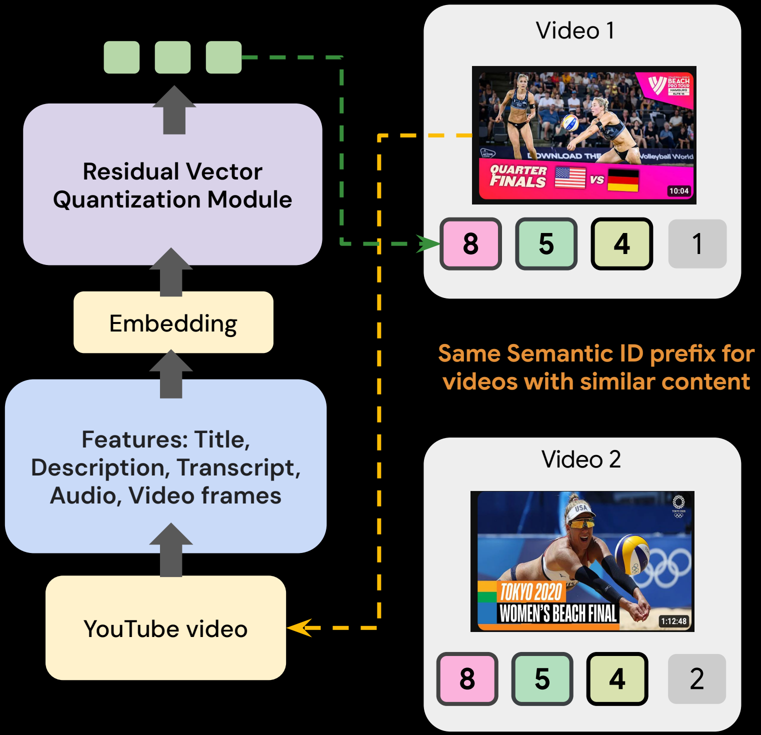
Google

Item/Corpus Representation Landscape

Item Representation	Memorization	Generalization	Feature storage Cost
Head item embeddings + OOV vocab	✅ Excellent performance for popular items	❌ Poor generalization to tail items	Low
Atomic IDs w/ Randomized hashing (random collisions)	✅ Excellent performance for popular items	❌ Shared embeddings help, but no guarantees due to random collisions	Low
Content Embeddings	❌ Poor memorization*	✅ Generalization from content	High (for user history features)
Semantic IDs	✅ Excellent performance for popular items	✅ Generalization from semantically meaningful collisions	Low

*Memorization with content embeddings could be achieved at the cost of increasing dense parameters.

Represent Corpus: ~~Atomic IDs~~ Hierarchical Semantic IDs

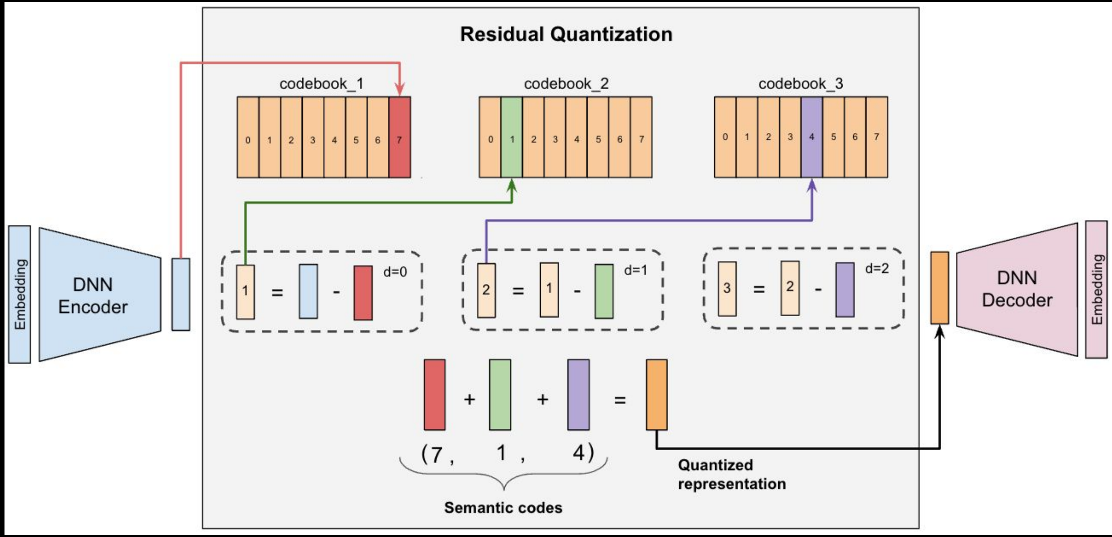


Better Generalization with Semantic IDs. (RecSys 2024)

Generating Item Semantic IDs with RQ-VAE

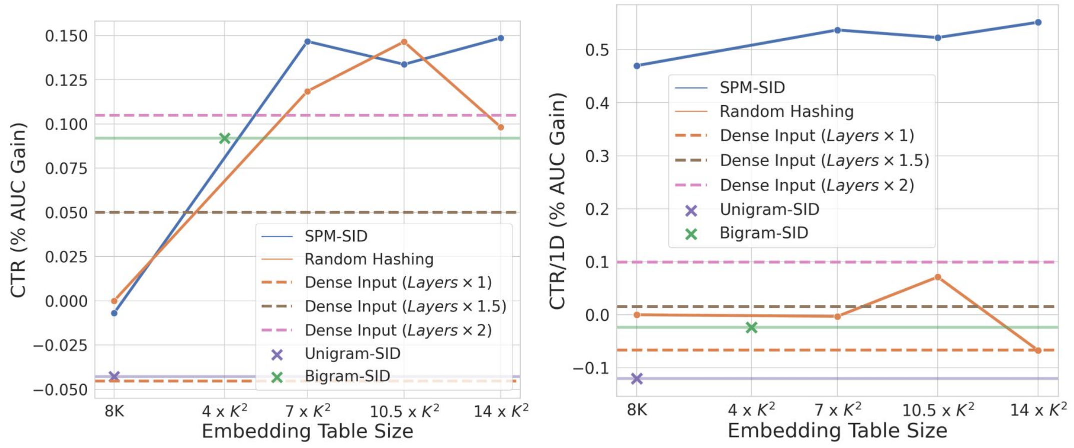
- Train a Residual Quantization VAE (RQ-VAE) with video content embeddings.

- The resultant Semantic IDs for item content embeddings.



Better Generalization with Semantic IDs. (RecSys 2024)
RecSys with Generative Retrieval (NeurIPS 23)

Semantic IDs in YouTube LEMs



(a) Overall CTR AUC (b) Cold-start CTR/1D AUC

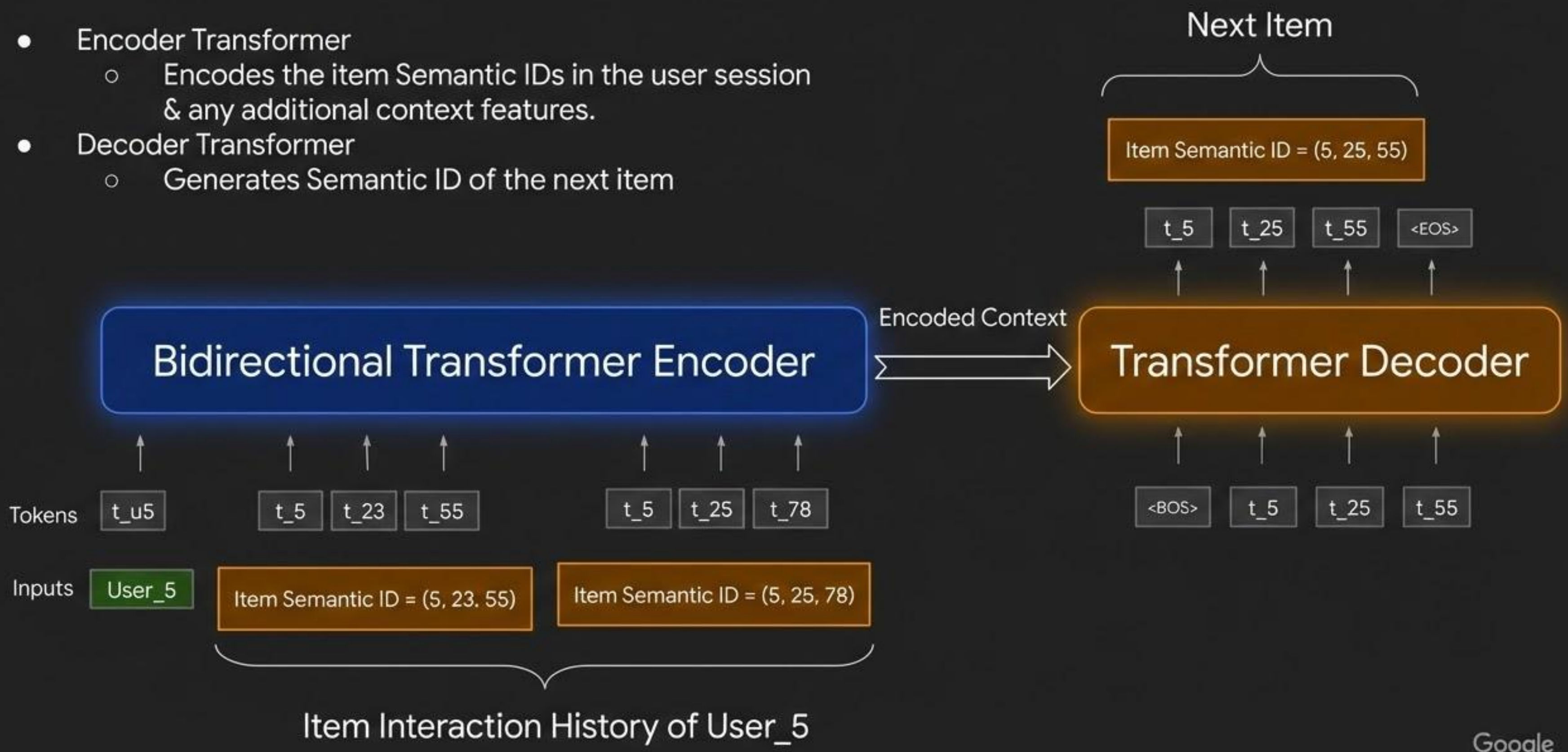
Figure 2: Percentage improvement in CTR AUC metric when user history is not used as a input feature. Improvement is relative to Random Hashing baseline with 8K embedding table size.

Significant impact in YouTube Production Large Embedding Models (LEMs) for improving generalization

Better Generalization with Semantic IDs. (RecSys 2024)

RecSys with Generative Retrieval (TIGER NeurIPS 23)

- Encoder Transformer
  - Encodes the item Semantic IDs in the user session & any additional context features.
- Decoder Transformer
  - Generates Semantic ID of the next item



Google

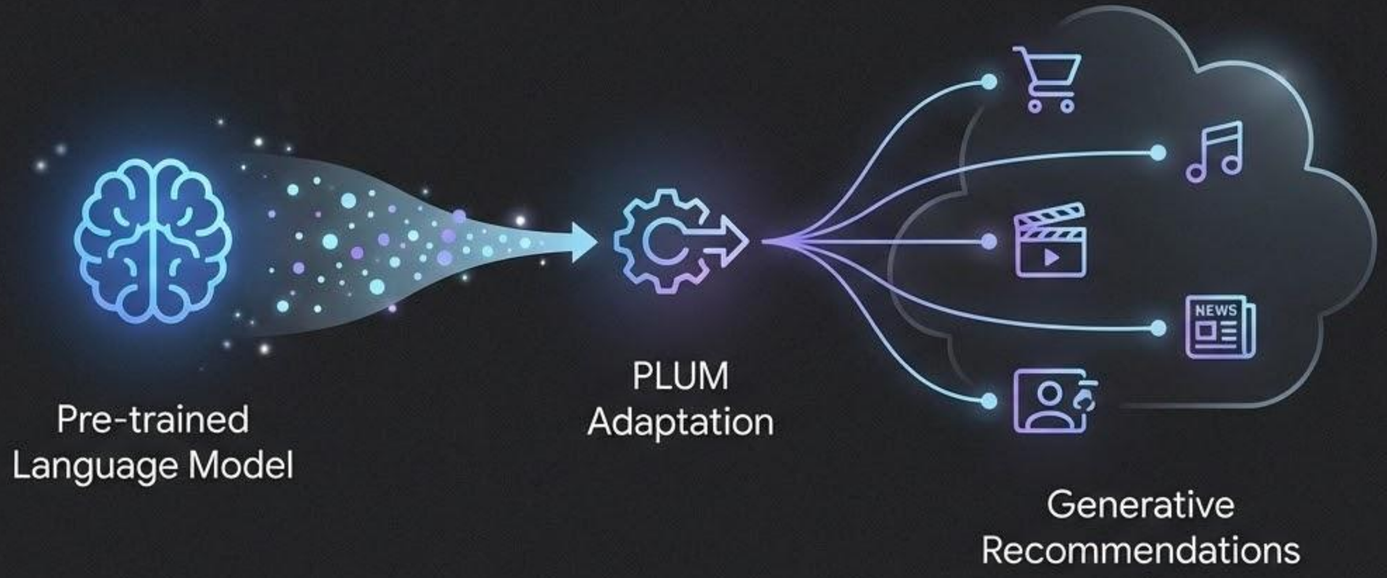
Results on public benchmark: Amazon Dataset

Performance on the sequential recommendation task on public recommendation benchmarks.

	Methods	Sports and Outdoors				Beauty				Toys and Games			
		Recall @5	NDCG @5	Recall @10	NDCG @10	Recall @5	NDCG @5	Recall @10	NDCG @10	Recall @5	NDCG @5	Recall @10	NDCG @10
Prior Methods	Caser	0.0116	0.0072	0.0194	0.0097	0.0205	0.0131	0.0347	0.0176	0.0166	0.0107	0.0270	0.0141
	HGN	0.0189	0.0120	0.0313	0.0159	0.0325	0.0206	0.0512	0.0266	0.0321	0.0221	0.0497	0.0277
	GRU4Rec	0.0129	0.0086	0.0204	0.0110	0.0164	0.0099	0.0283	0.0137	0.0097	0.0059	0.0176	0.0084
	BERT4Rec	0.0115	0.0075	0.0191	0.0099	0.0203	0.0124	0.0347	0.0170	0.0116	0.0071	0.0203	0.0099
	FDSA	0.0182	0.0122	0.0288	0.0156	0.0267	0.0163	0.0407	0.0208	0.0228	0.0140	0.0381	0.0189
	SASRec	0.0233	0.0154	0.0350	0.0192	0.0387	0.0249	0.0605	0.0318	0.0463	0.0306	0.0675	0.0374
	S^3-Rec	0.0251	0.0161	0.0385	0.0204	0.0387	0.0244	0.0647	0.0327	0.0443	0.0294	0.0700	0.0376
Generative Retrieval [Ours]		**0.0264**	**0.0181**	**0.0400**	**0.0225**	**0.0454**	**0.0321**	**0.0648**	**0.0384**	**0.0521**	**0.0371**	**0.0712**	**0.0432**
		+5.22%	+12.55%	+3.90%	+10.29%	+17.31%	+29.04%	+0.15%	+17.43%	+12.53%	+21.24%	+1.71%	+14.97%

Google

PLUM: Adapting Pre-trained Language Models for Industrial-scale Generative Recommendations

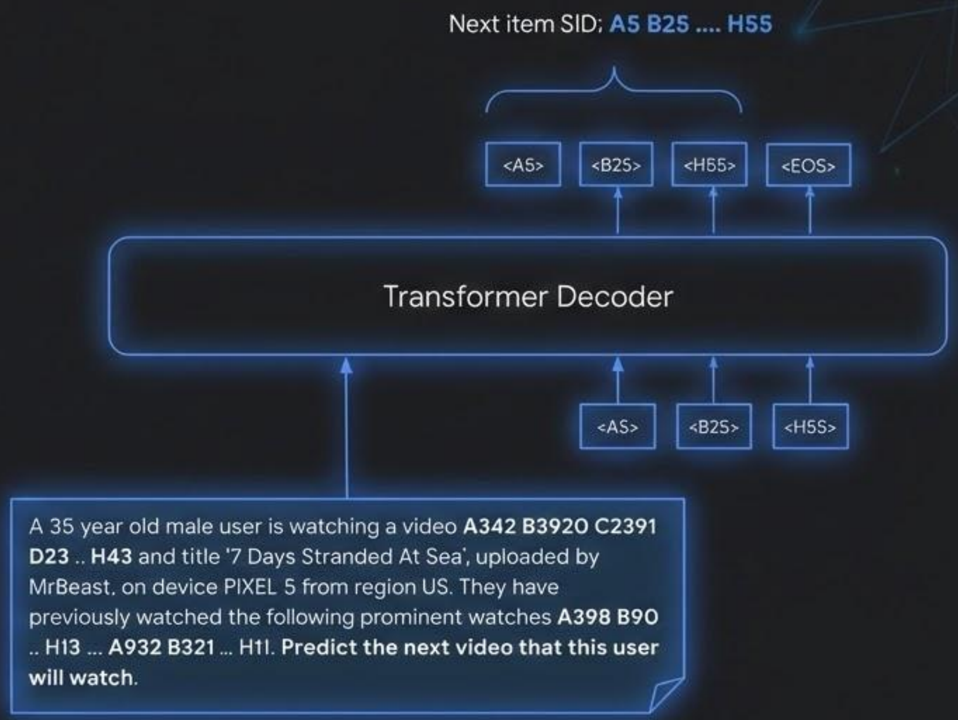


Google

New Paradigm for Industrial Recs: GenRetrieval

LLM Decoder as an implicit index with Semantic IDs:

- Modeling with hierarchical output space represented by SIDs without negative sampling.
- Enhanced user–item interactions.
- Controllable diversity via flexible decoding strategies.



High-level Recipe: LLM x RecSys

1. Tokenize content

Capture the essence of your content into an atomic token

Rich representation → embedding → quantization

Outcome: a new “language” for your domain

2. Adapt LLM: english <> domain language

Adapt foundation model to reason across english & new tokens

Outcome: a “bilingual” LLM across natural language & tokens

3. Prompt with user info

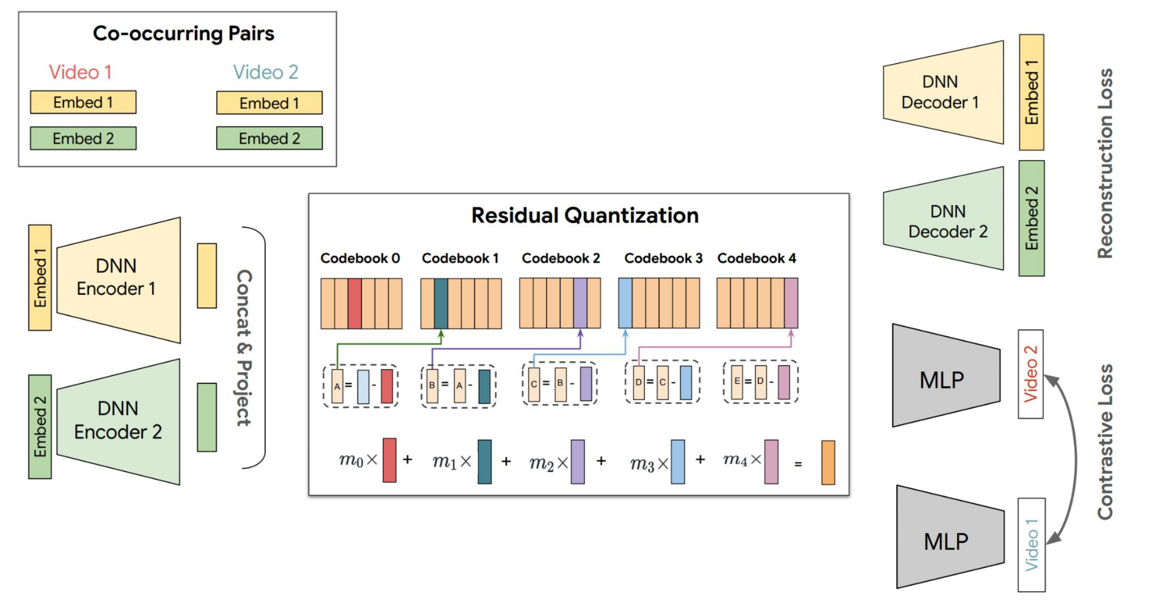
Construct prompts with user information, activity, actions

Train surface/task-specific models

Outcome: Generative recommendations with LLM

Tokenization: Semantic ID v2

- Fuse multimodal embeddings in the input.
- Incorporating engagement signals in SID training.
- Multi-resolution codebooks to reduce the search space during GenRetrieval decoding.
- Progressive masking for improving hierarchy



SID Model	SID Uniqueness	VID Recall@10
SIDv1 (Baseline)	94.0%	12.3%
SIDv2 (Ours)	96.7%	14.4%
Ablate Multi-Resolution	94.8%	13.2%
Ablate Multi-Embedding	96.9%	12.8%
Ablate Co-occurence	91.8%	12.6%

Table 4: Ablation experiments on SIDv2 changes

Adaptation: Continued Pre-training from Gemini

- Align semantic ID (SID) tokens and text tokens through domain-specific data.
- Inject recommendation knowledge into model weight, e.g. co-watch signals, user behaviors, user preferences, etc

Gemini — YouTube

Continued Pre-Training

Training mixtures:

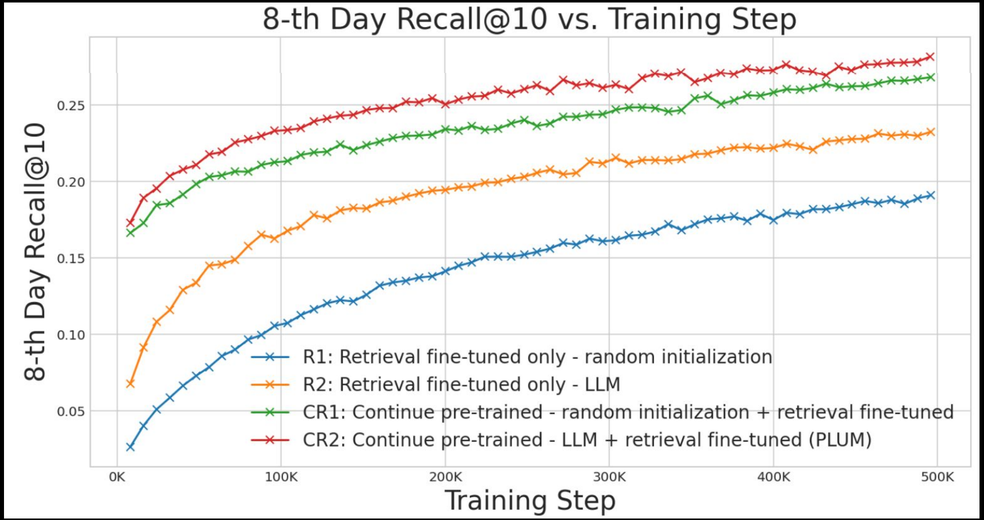
- Synthetic data: text + SID
- User behaviors: SID sequences of video watches

Table 1: Example schemas used in continued pre-training.

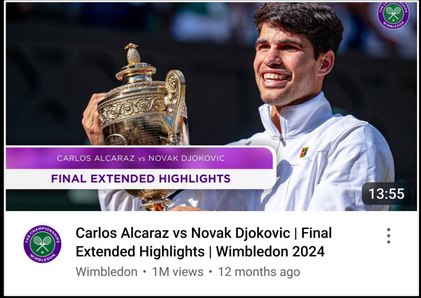
Example user behavior training data
wh = <sid_1> <channel_name> <watch_ratio> <watch_time> <hours_since_final_watch> <sid_2> <channel_name> ... || <sid_n>
SID + video title
Video <sid> has title (en): <video_title>
SID + video topics
The topics in video <sid> are: <topics>

Continued Pre-training (CPT) leads to better recall

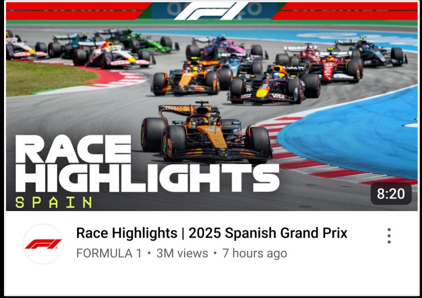
Model	Pre-trained LLM	CPT	Recall@10 (8th-day)
R1	No	No	0.19
R2	Yes	No	0.23
CR1	No	Yes	0.27
CR2	Yes	Yes	0.28



Continued Pre-training (CPT) enables reasoning across Semantic IDs

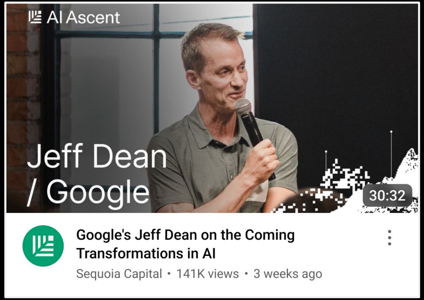


[A185 ... H201]



[T707 ... W300]

[Y212 ... K978]



[J110 ... R561]

Prompt:

Video **`A185`** ... **`H201`** is interesting to Tennis fans since it is about Wimbledon.
Video **`T707`** ... **`W300`** is interesting to F1 fans since it is about Spanish Grand Prix.
Video **`Y212`** ... **`K978`** is interesting to Math fans since it is about Pi.
Video **`J110`** ... **`R561`** is interesting to

Output:

Technology fans since it is about AI.

Generative Retrieval: Prompt LRM with demographics, seed video, watch history

User

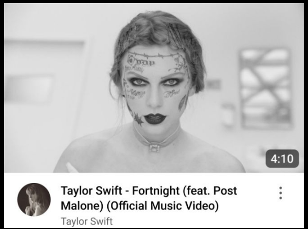
24 yr old, Female
US, Android

Context video

PLUM prompt

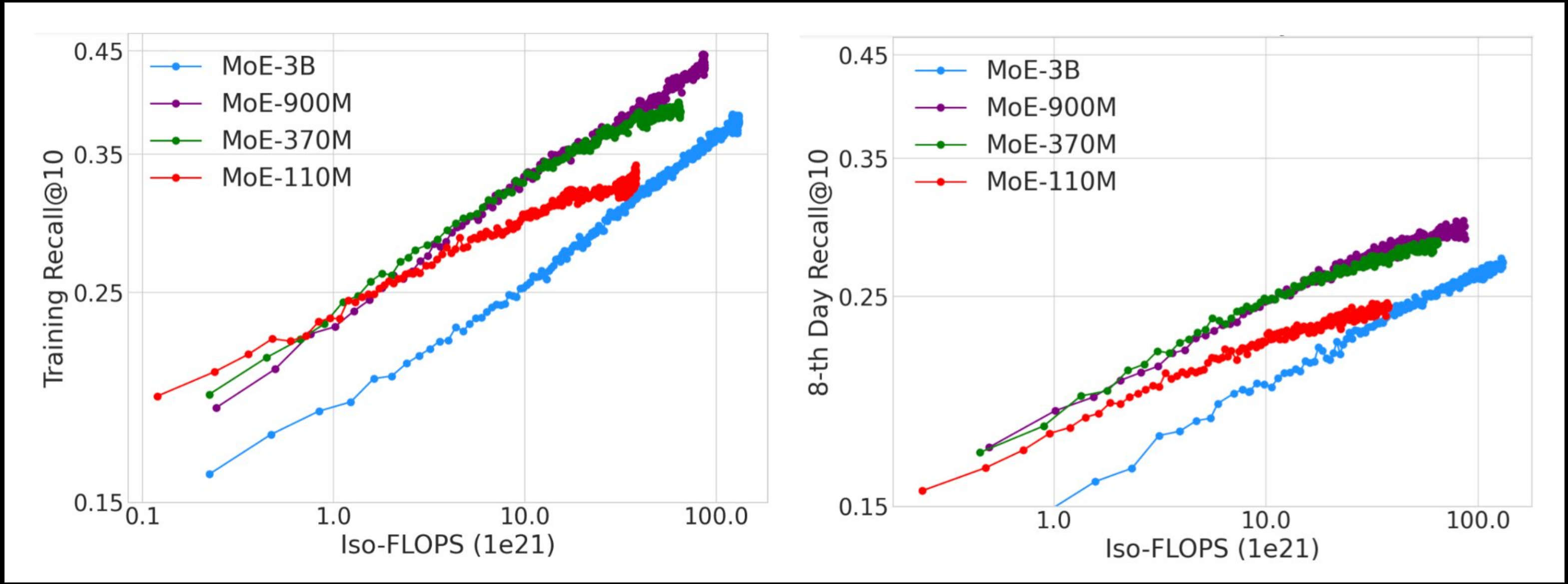
```
User: region US | 24 years female | device
ANDROID | origin watch next|

Context video: channel Olympics |
title WHAT A COMEBACK! | Men's 400m |
#Paris2024 highlights | SemanticID_1

Watch history:
SID_1 Taylor Swift 100% 260.00s
SID_2 Kris Hui 40% 260.00s
SID_3 NBC Sports 100% 320.00s
```

Watch history



Google

Scaling study with Gemini 1.5 small models



Strong Power-Law relationship b/w compute and Recall@K

PLUM Improves Recs Quality

Table 2: Comparison of recommendation quality: Each number is a ratio, dividing the metric for PLUM by that of LEM.

Metric	LFV	Shorts
Effective Vocab Size	2.60x	13.24x
CTR	1.42x	1.33x
WT/View	0.72x	1.13x
WF/View	1.32x	1.03x

PLUM is powerful, but is expensive to serve

Strengths
Learns quickly Training data efficient: less data needed to reach prod performance **Handles toughest recs** Complex recs tasks when we know least about users
Limitations
Expensive Serving costs can be too large

Can we serve PLUM offline?

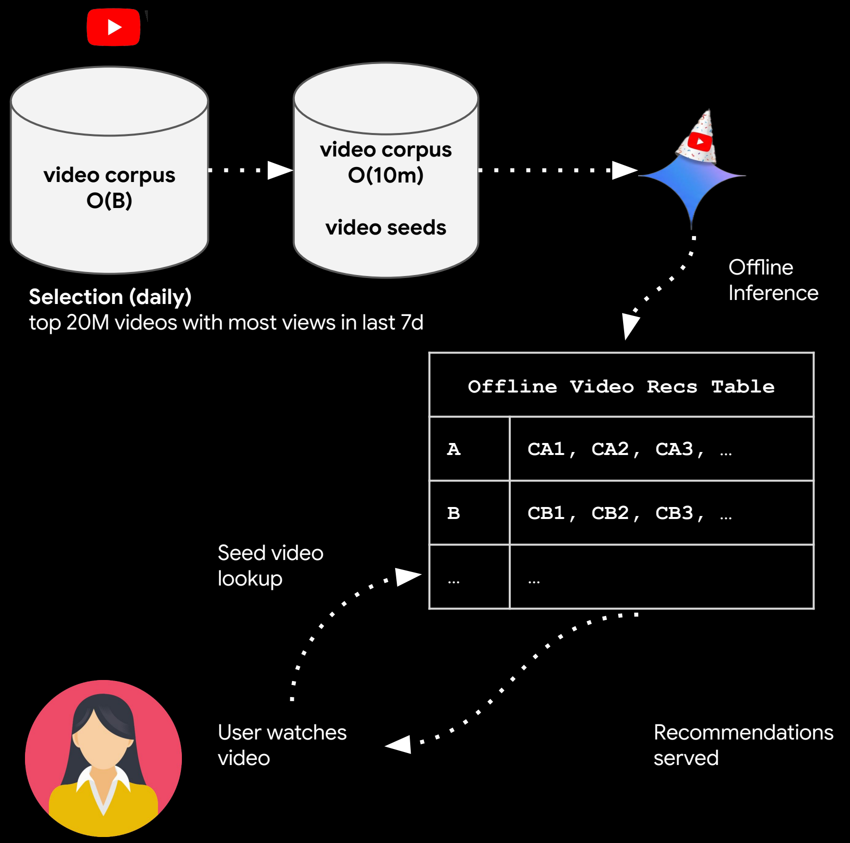
PLUM Offline for Efficient Serving

Goal: build offline video → recommendations table

seed video A	[candidate A1, candidate A2, … , candidate A80]
seed video B	[candidate B1, candidate B2, … , candidate B80]
seed video Z	[candidate Z1, candidate Z2, … , candidate Z80]

Unpersonalized prompt

```
language {seed_lang} | duration
{video_length} | age {video_age} | title
{title} | channel {uploader} | {seed_sid}
```



Google

Thank you!

Thoughts? Reach me at nikhilmehta.dce@gmail.com