Power of LLMs for Recommender Systems

Daniel Homola, AI Research Engineer at BMW

1. The Why and Evolution of Recommendations

2. Case Studies in Action: Learning from YouTube, Spotify, Netflix

3. LLMs for RecSys: Opportunities & Gaps

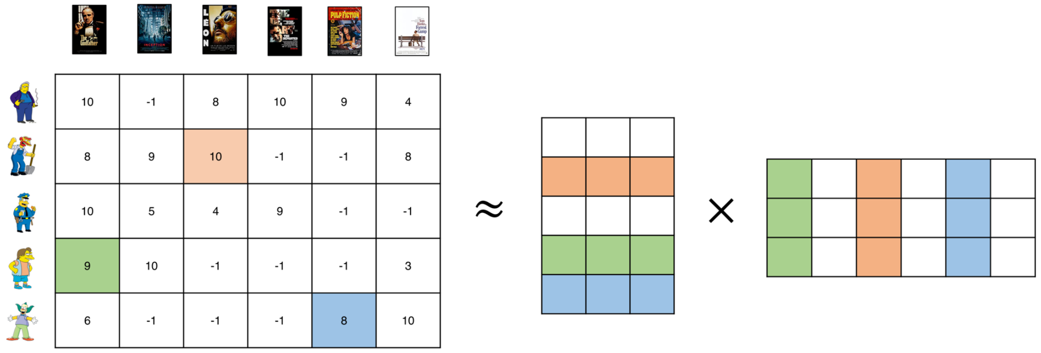
Majority of digital engagement is driven by recommendations

YouTube watch time
Spotify streams
Netflix content consumption
...

>70% of consumers expect companies to deliver personalized recommendations

RECOMMENDER SYSTEMS
RECOMMENDER SYSTEMS EVERYWHERE
LIGHTYEAR
SPACE RANGER

From **Shallow Model** (e.g. matrix factorization [1]) ...



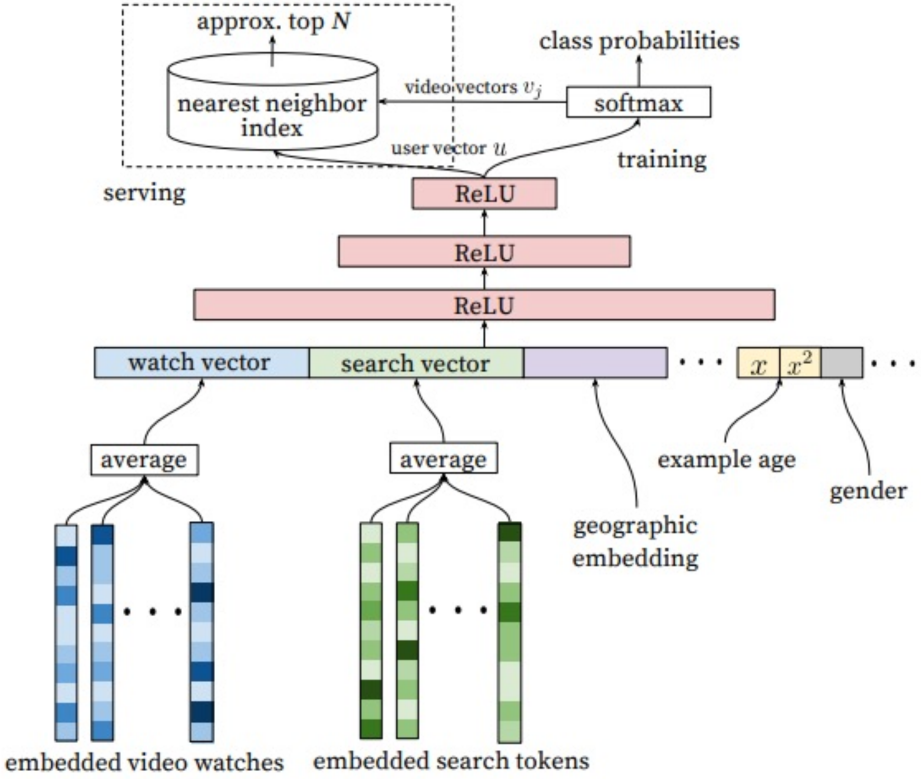
10	-1	8	10	9	4
8	9	10	-1	-1	8
10	5	4	9	-1	-1
9	10	-1	-1	-1	3
6	-1	-1	-1	8	10

User-item Interaction Matrix
(R)

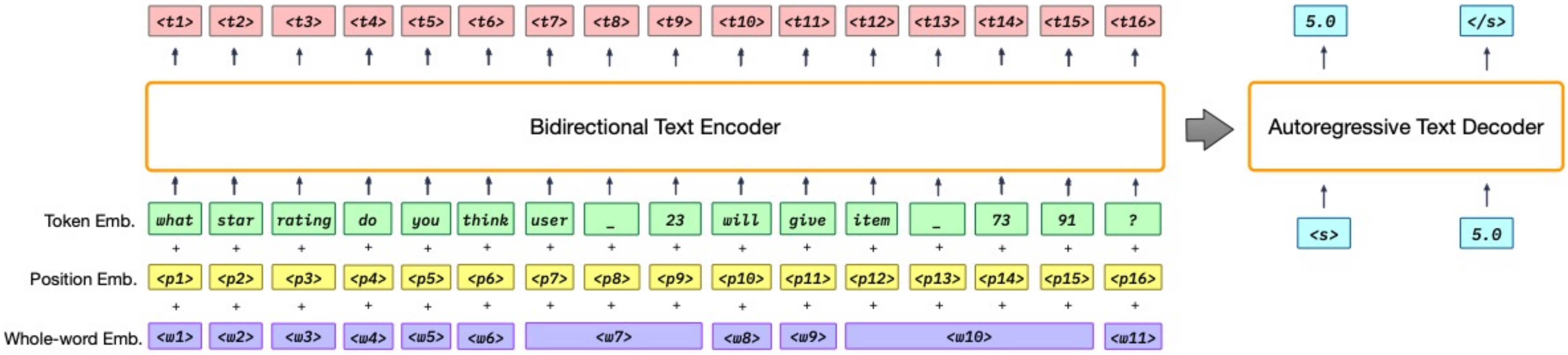
User Matrix
(Q)

Item Matrix
(P)

From Shallow Model, to **Deep Model** (e.g. deep neural network [2])...



From Shallow Model, to Deep Model, and to **Large Model** (e.g. P5 [3])



<t1> <t2> <t3> <t4> <t5> <t6> <t7> <t8> <t9> <t10> <t11> <t12> <t13> <t14> <t15> <t16>
5.0
</s>
Bidirectional Text Encoder
Autoregressive Text Decoder
Token Emb.
what star rating do you think user _ 23 will give item _ 73 91 ?
<s>
5.0
Position Emb.
<p1> <p2> <p3> <p4> <p5> <p6> <p7> <p8> <p9> <p10> <p11> <p12> <p13> <p14> <p15> <p16>
Whole-word Emb.
<w1> <w2> <w3> <w4> <w5> <w6> <w7> <w8> <w9> <w10> <w11>

Some challenges in recommender systems...

Personalization

Scalability Cold Start

Generalization Bias

Fairness Data Sparsity

Explainability

Representation Quality

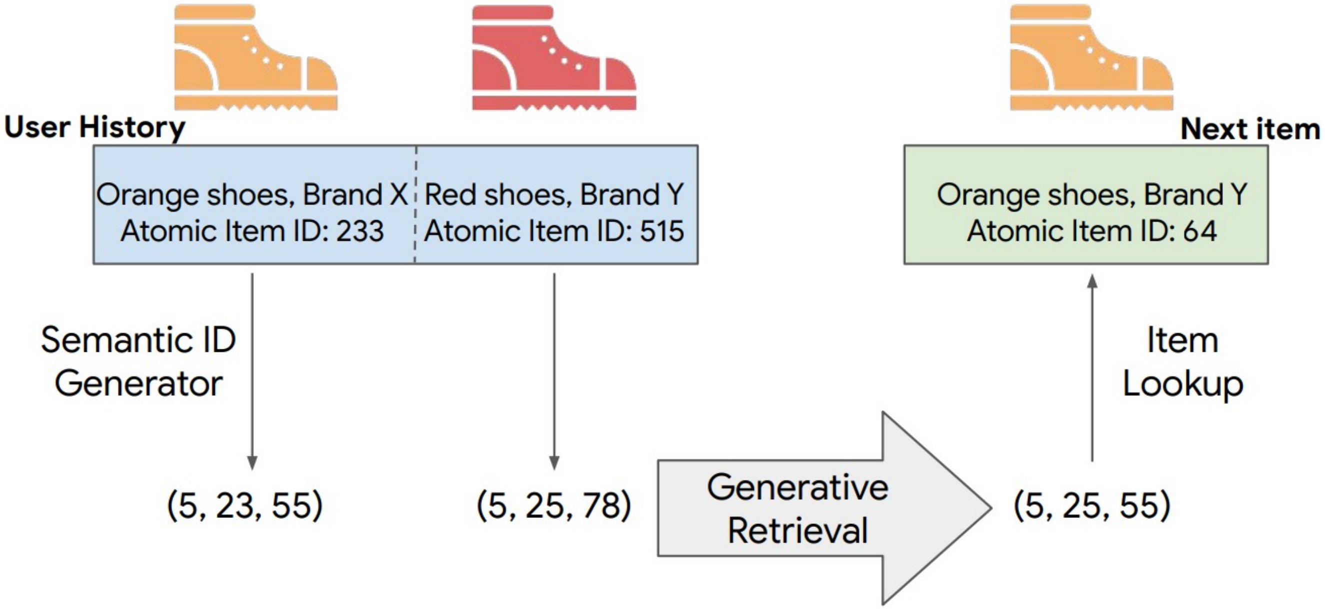
Retrievability

Challenges and LLM-inspired innovations
at YouTube, Spotify, and Netflix

How can **YouTube** better understand and recommend new content by replacing random item IDs with content-based representations—without slowing down the system?

YouTube: Generative Retrieval and Scalable Ranking with Semantic IDs [4, 5]

- Move from random item IDs to semantic IDs
- These compact item representations are flexible for **both generative retrieval** [4] and **efficient ranking** (YouTube production models [5])



Outcome of YouTube's effort

🔁 Semantic IDs improve generalization through compact, hierarchical tokens that enhance retrieval and ranking

🧠 LLM-style tokenization enables subword embeddings that effectively adapt Semantic IDs in ranking models

🚀 Productionized in YouTube's ranking models over a corpus of billions of videos with controlled memory cost and low inference latency

How can **Spotify** improve content retrievability and encourage exploration in search?

Spotify: LLM-assisted Data Generation for Query Recommendations [6, 7]

- Shift from instant search to hybrid query recommendation
- Using LLMs to generate broad intent queries, enrich training data, and shape user search behavior

Outcome of Spotify's effort

🔍 Increased exploratory behavior: More exploratory intent queries

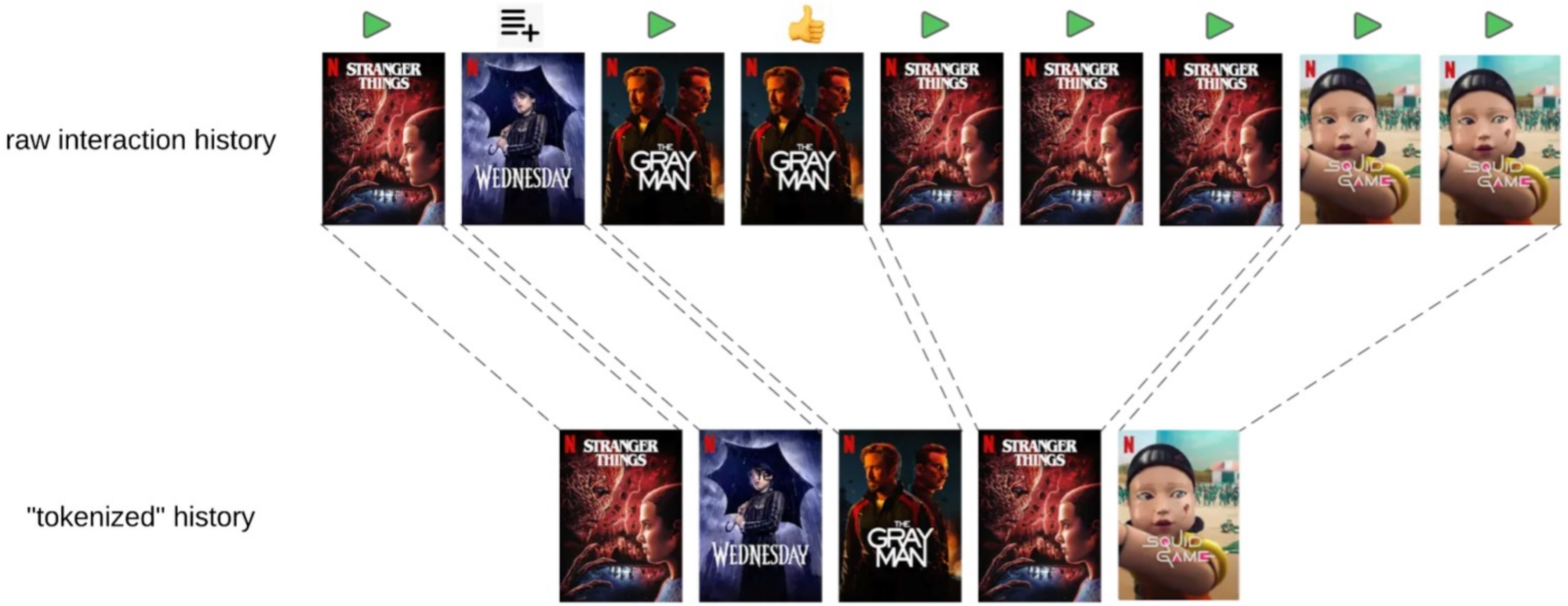
🕵️‍♂️ Improved content retrievability for long-tail items

🎵📖 Broad application across domains: music, podcasts, and audiobooks

How can **Netflix** transition from many specialized models to a foundation model that captures long-term member preferences and scales across use cases?

Netflix: Foundation Model for Personalized Recommendation [8, 9]

- Next-token prediction objective over user interaction sequences similar to GPT but with critical modifications to the objective
- Centralized preference learning capturing long-term user behavior
- Sliding window training & sparse attention mechanisms



Outcome of Netflix’s effort

🧩 Unified data-centric system for various downstream applications:

Direct use as a predictive model
Utilization of user and entity embeddings

🧠 Improved long-term personalization and representation quality

📈 Scalable training on large histories

LLMs can help in RecSys.

LLM concepts enrich recommender systems.

LLMs need help in RecSys.

Not all LLM concepts work out-of-the-box.

Takeaways: LLMs can help

Hybrid systems: LLMs don't replace all recommendation models but increasingly enhance them in hybrid setups

Richer training data: LLMs help in training more robust models by enriching training data with synthetic or inferred data

Foundational recommender models: Inspired by LLMs, we can train foundational recommenders using large-scale unlabeled interaction data and next-token prediction objective

Takeaways: LLMs need help

Scalability: Large models often struggle to meet latency requirements of industrial recommender systems, requiring hybrid solutions or optimizations

Lack of item expertise: Mechanisms such as stable, semantically rich token vocabularies or incremental training strategies are required to improve generalization and enable fast adaptation to new items

Behavior alignment: Models need to be aware of user collaborative signals

Other trends in RecSys

Scaling laws

Multitask learning: Unified architecture of search and recommendations

Conversational, multimodal recommender systems

Resources

[1] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," Computer, vol. 42, no. 8, pp. 30–37, 2009. [link]

[2] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for YouTube recommendations," in Proc. 10th ACM Conf. Recommender Systems (RecSys), 2016. [link]

[3] S. Geng, S. Liu, Z. Fu, Y. Ge, and Y. Zhang, "Recommendation as Language Processing (RLP): A Unified Pretrain, Personalized Prompt & Predict Paradigm (P5)," in Proc. 16th ACM Conf. Recommender Systems (RecSys), 2022. [link]

[4] S. Rajput, N. Mehta, A. Singh, R. Hulikal Keshavan, T. Vu, L. Heldt, and M. Sathiamoorthy, "Recommender systems with generative retrieval," in Advances in Neural Information Processing Systems (NeurIPS), vol. 36, 2023. [link]

[5] A. Singh, T. Vu, N. Mehta, R. Keshavan, M. Sathiamoorthy, Y. Zheng, and X. Yi, "Better generalization with semantic IDs: A case study in ranking for recommendations," in Proc. 18th ACM Conf. Recommender Systems (RecSys), 2024. [link]

[6] G. Penha, E. Palumbo, M. Aziz, A. Wang, and H. Bouchard, "Improving content retrievability in search with controllable query generation," in Proc. ACM Web Conf. (WWW), 2023. [link]

[7] H. Lindstrom, H. J. Corona Pampin, E. Palumbo, and A. Liu, "Encouraging exploration in Spotify search through query recommendations," in Proc. 18th ACM Conf. Recommender Systems (RecSys), 2024. [link]

[8] K.-J. Hsiao, Y. Feng, and S. Lamkhede, "Foundation model for personalized recommendation," Netflix Tech Blog, Mar. 21, 2025. [link].

[9] S. Joshi, Y. Feng, K. J. Hsiao, Z. Zhang, and S. Lamkhede, "Sliding window training—Utilizing historical recommender systems data for foundation models," in Proc. 18th ACM Conf. Recommender Systems (RecSys), 2024. [link]