Representation Learning for Recommender Systems

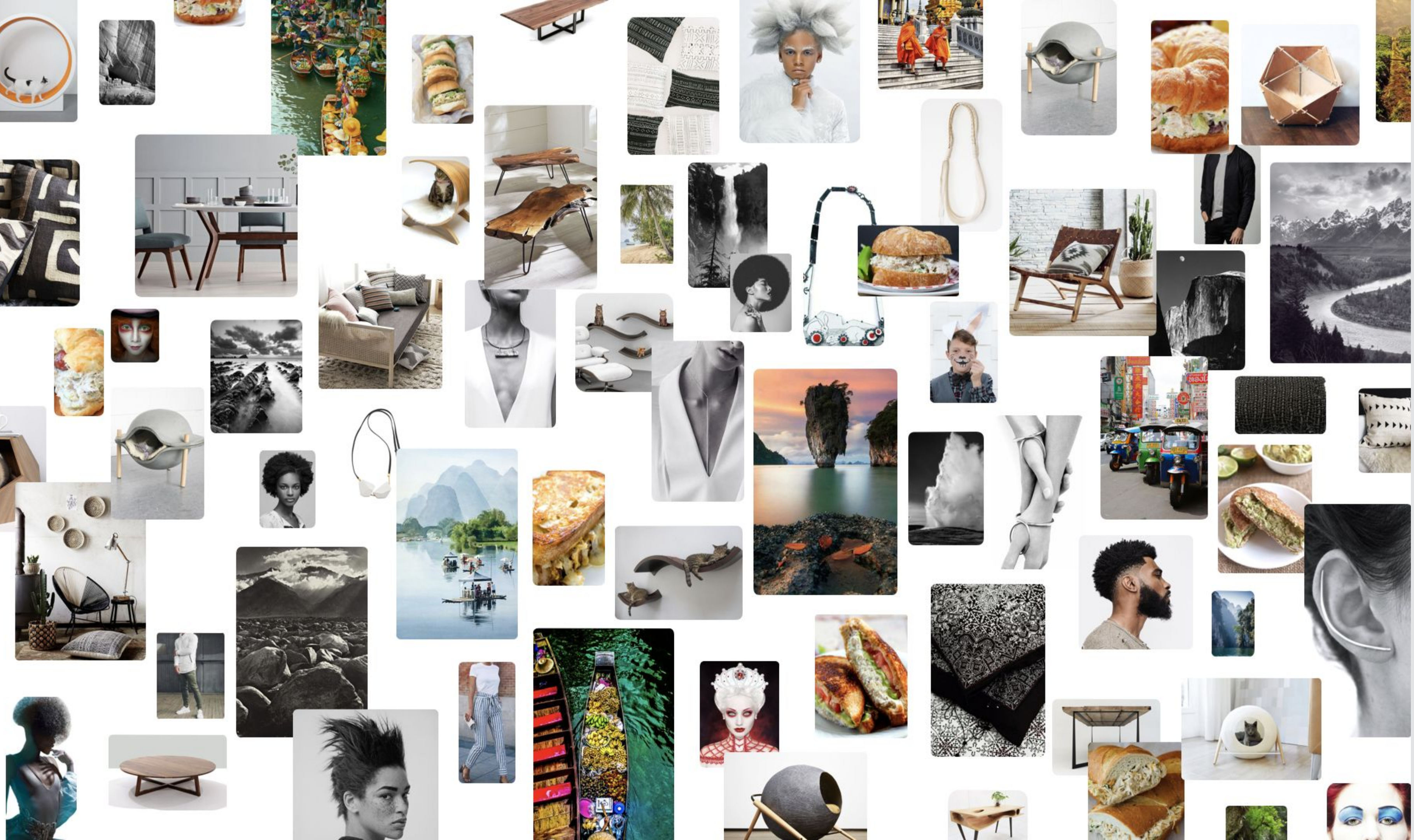
Andrew Zhai, Applied Scientist

Aug 15th 2021

Introduction

Andrew Zhai

- Senior Staff Applied Scientist
- Deep Learning @ Pinterest, TL of Representation Learning
- Work across recommendation funnel to build scalable ML solutions



Pinterest

Bring ***everyone*** the ***inspiration*** to create a life they love

454 M
Global Monthly Active Users[1]

300 B
Pins saved[2]

6 B+
Boards[2]

Pinterest is available in more than
30 languages[3]

91% of Pinners
say Pinterest is a place filled with
positivity[4]

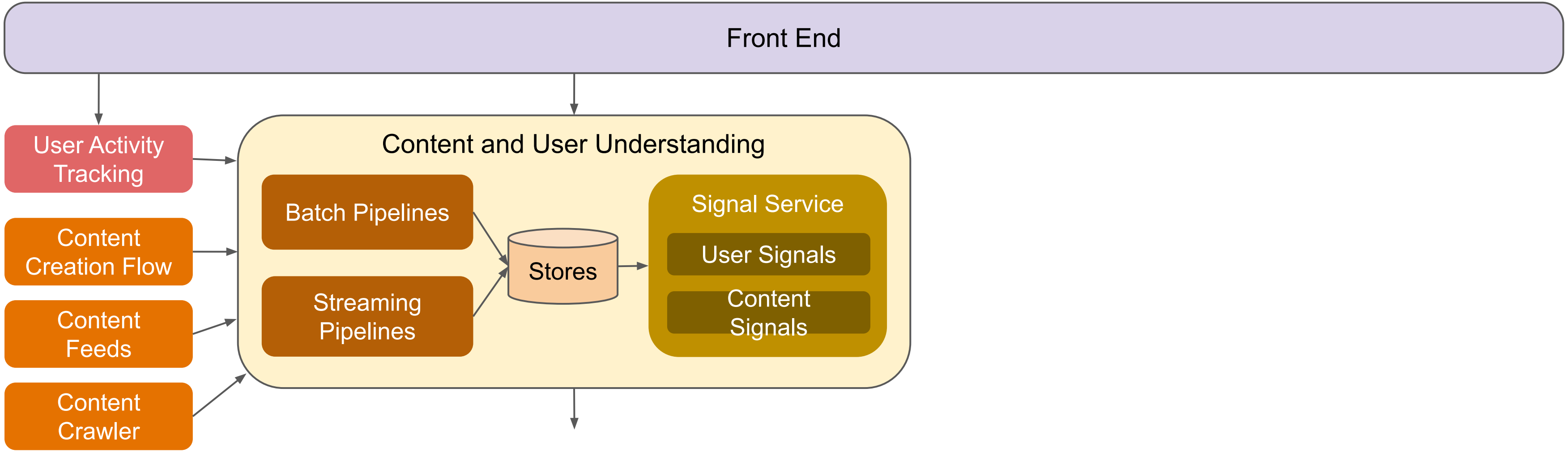
1 Pinterest, Global analysis, June 2021
2 Pinterest, Global analysis, Jan 2021
3 Pinterest internal data, 2020
4 Talkshoppe, US, Emotions, Attitudes & Usage study, Oct 2018

Pinterest Home Feed

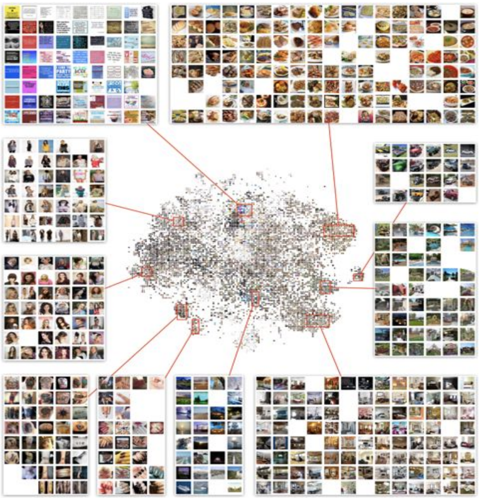
System Architecture for scalability - 00s of thousands of users and few billion Pins (content)

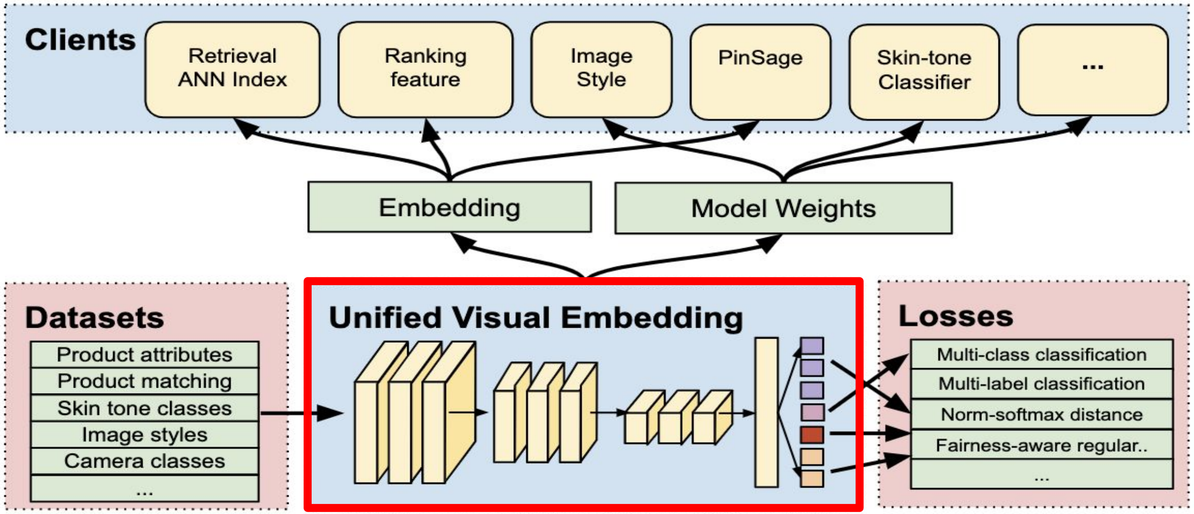
Front End

System Architecture for scalability - 00s of thousands of users and few billion Pins (content)



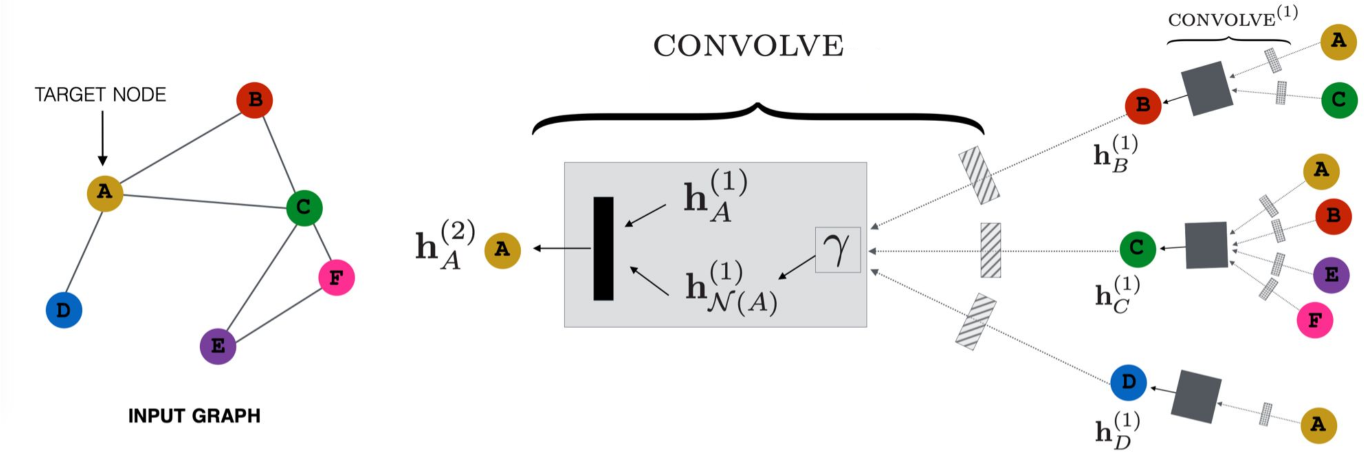
Content Signals: Visual Embeddings





- Input: An image
- Output: An embedding (+ more, later on…)

Content Signals: Graph Embeddings

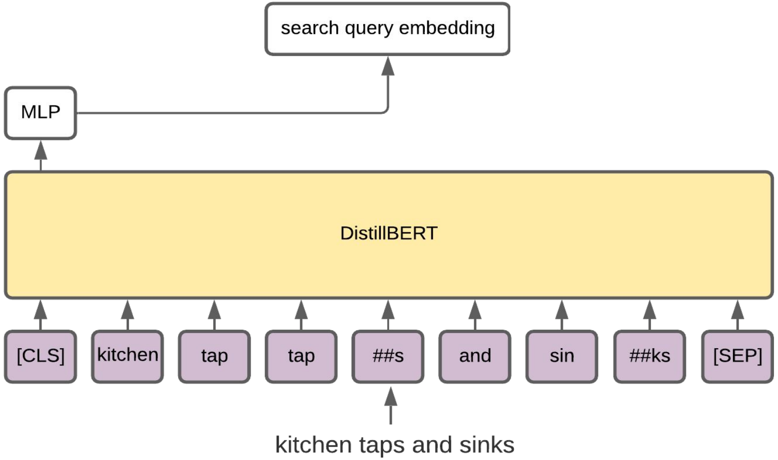


- Combine content and engagement signals in a **inductive** manner to produce more comprehensive representations
- Input: Pin-to-board graph, content features of each node (e.g., text, visual embeddings)
- Output: An embedding per node

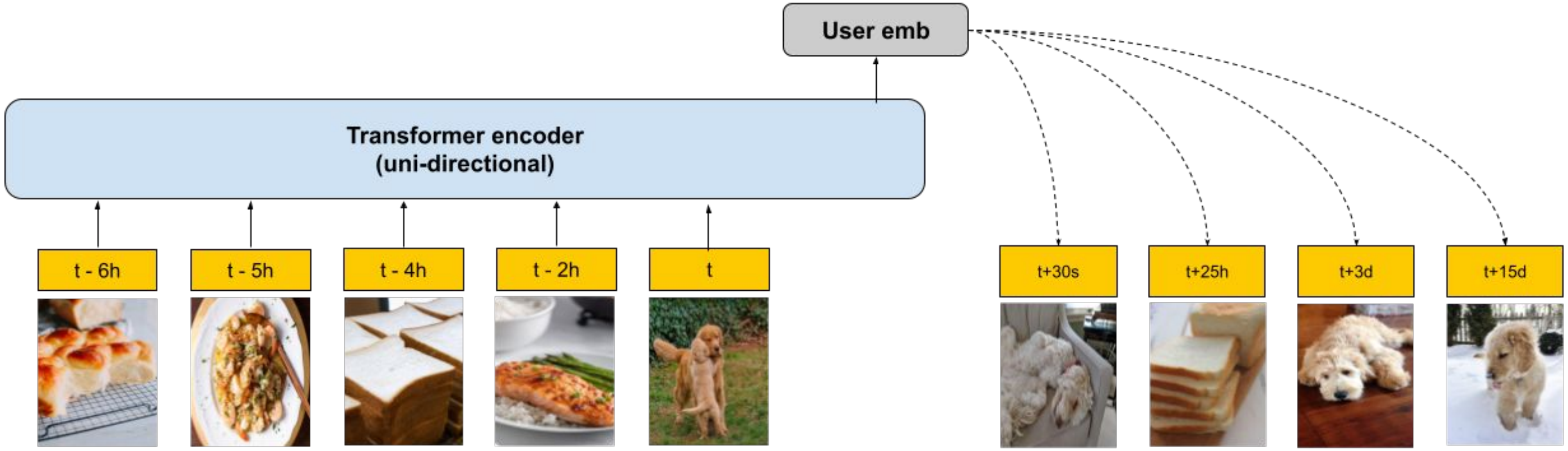
Content Signals: Search Query Embeddings

- Input: Search query (text)
- Output: Embedding optimized for search
- **Tokenized** input for generality

kitchen taps and

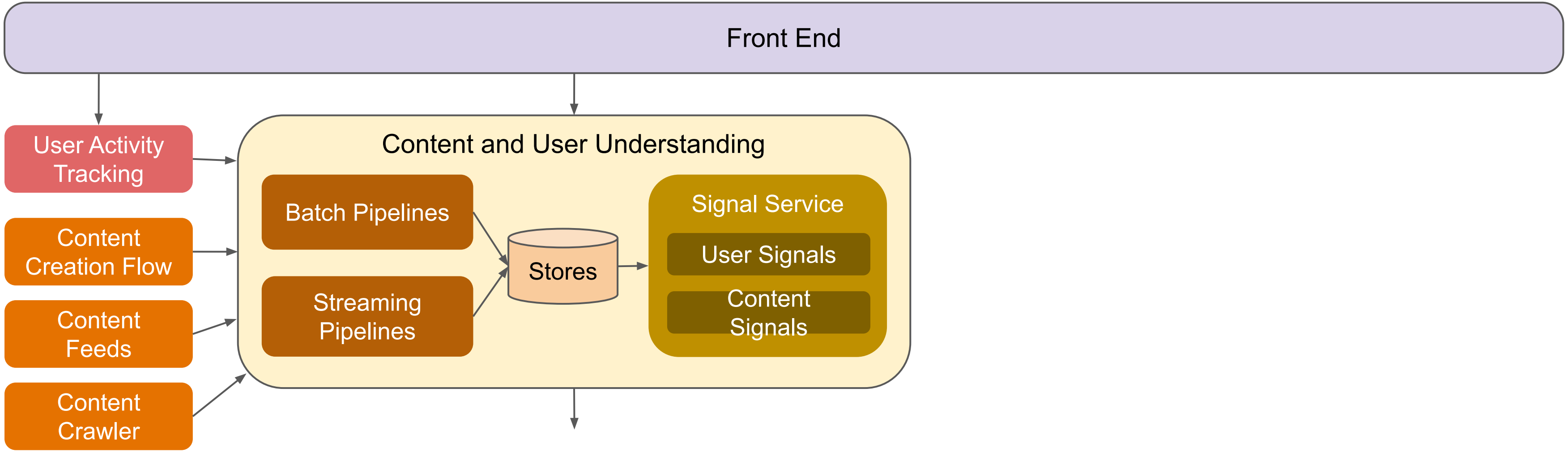


User Signals: User Embeddings

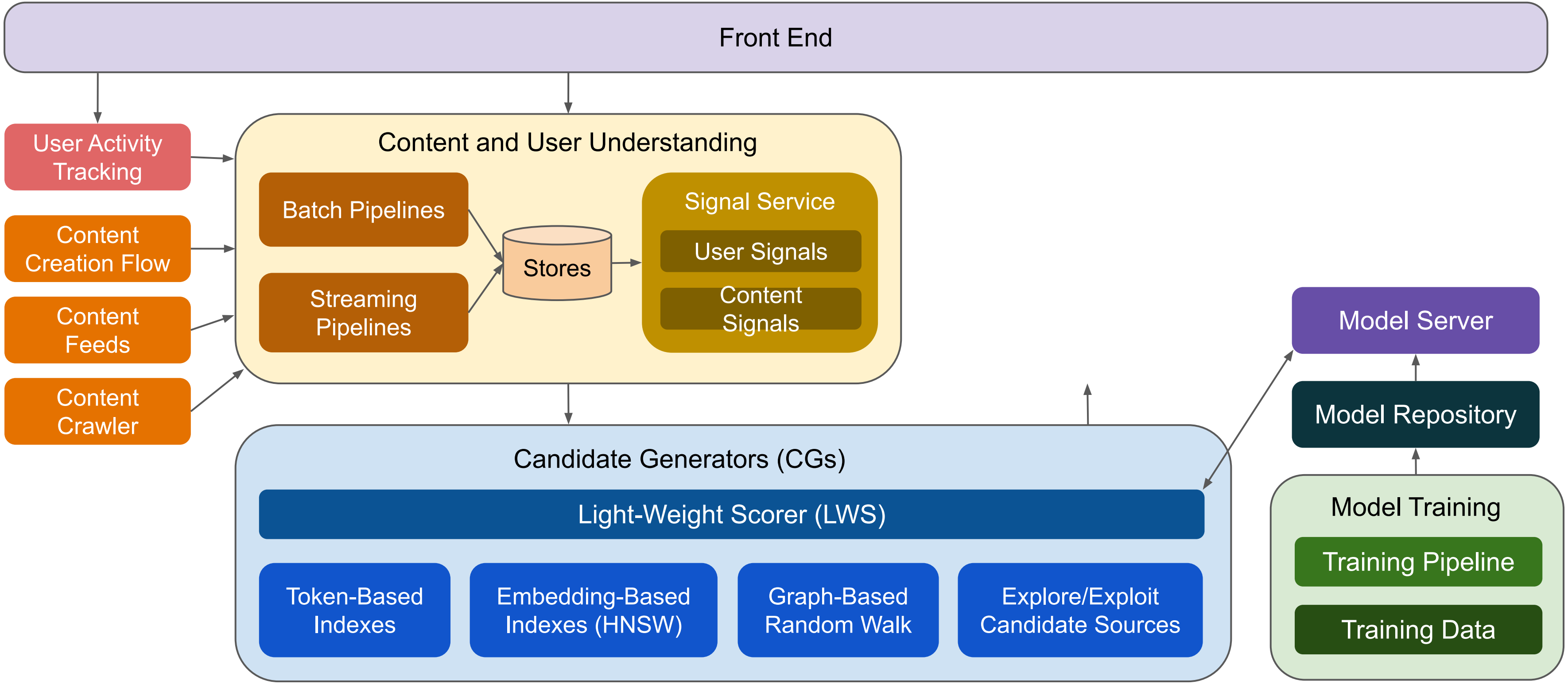


- Input: user activity sequence across all of Pinterest
- Output: one user embedding

System Architecture for scalability - 00s of thousands of users and few billion Pins (content)

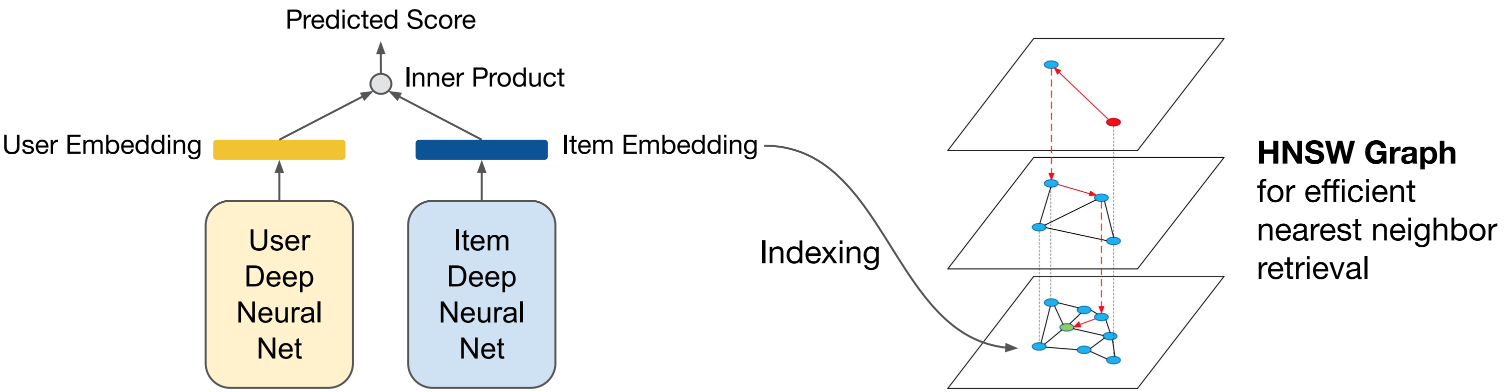


System Architecture for scalability - 00s of thousands of users and few billion Pins (content)



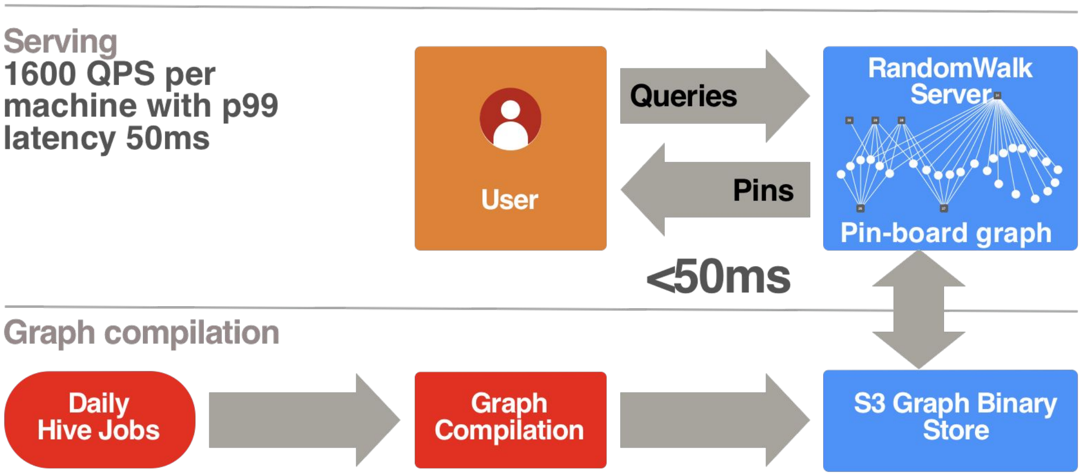
Pinterest

Candidate Generation: Embedding-Based Retrieval



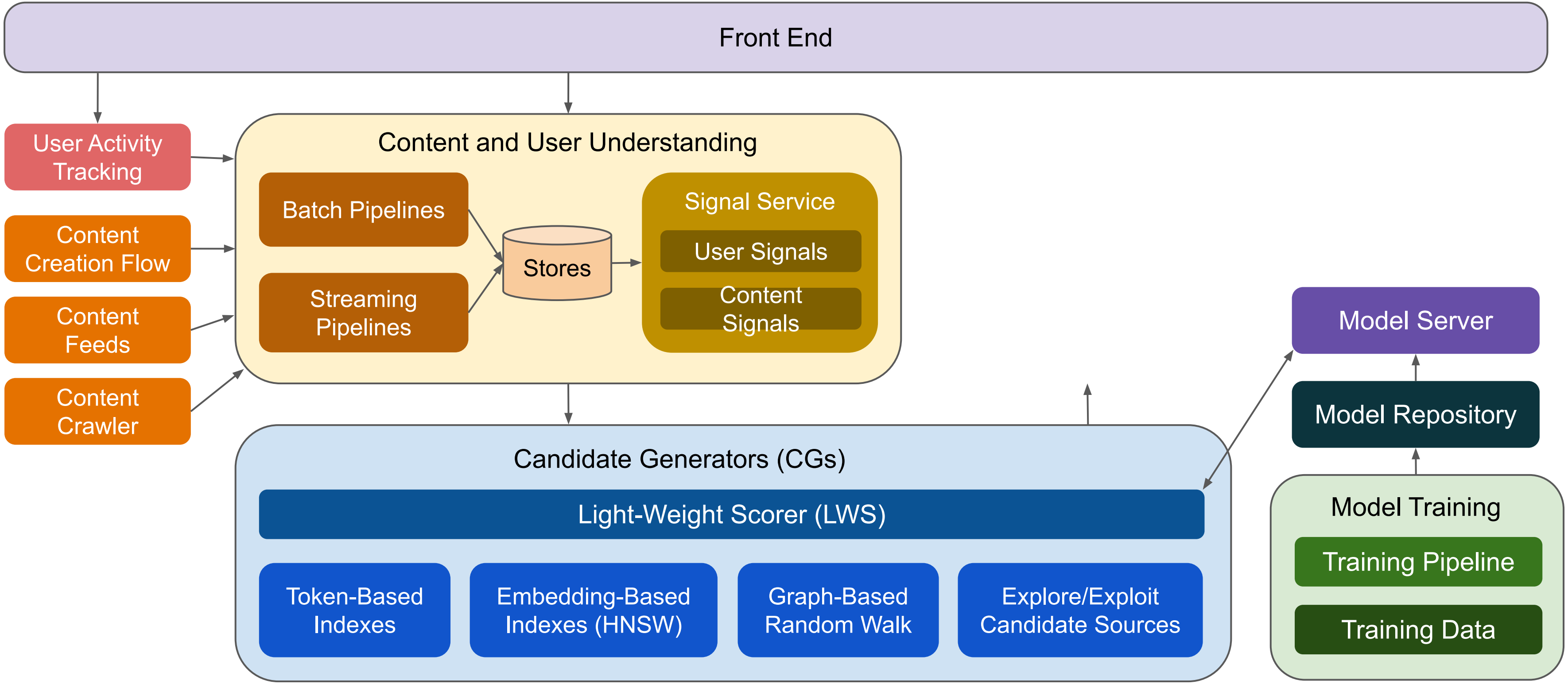
- Train a two-tower deep neural network to predict user engagement
- Precompute the embedding vectors for all items and index them into a Hierarchical Navigable Small World (HNSW) graph
- Given a user embedding vector, retrieve *k* nearest neighbors (items) based on a learned similarity function (the neural network) through the HNSW graph

Candidate Generation: Random Walk on a Graph

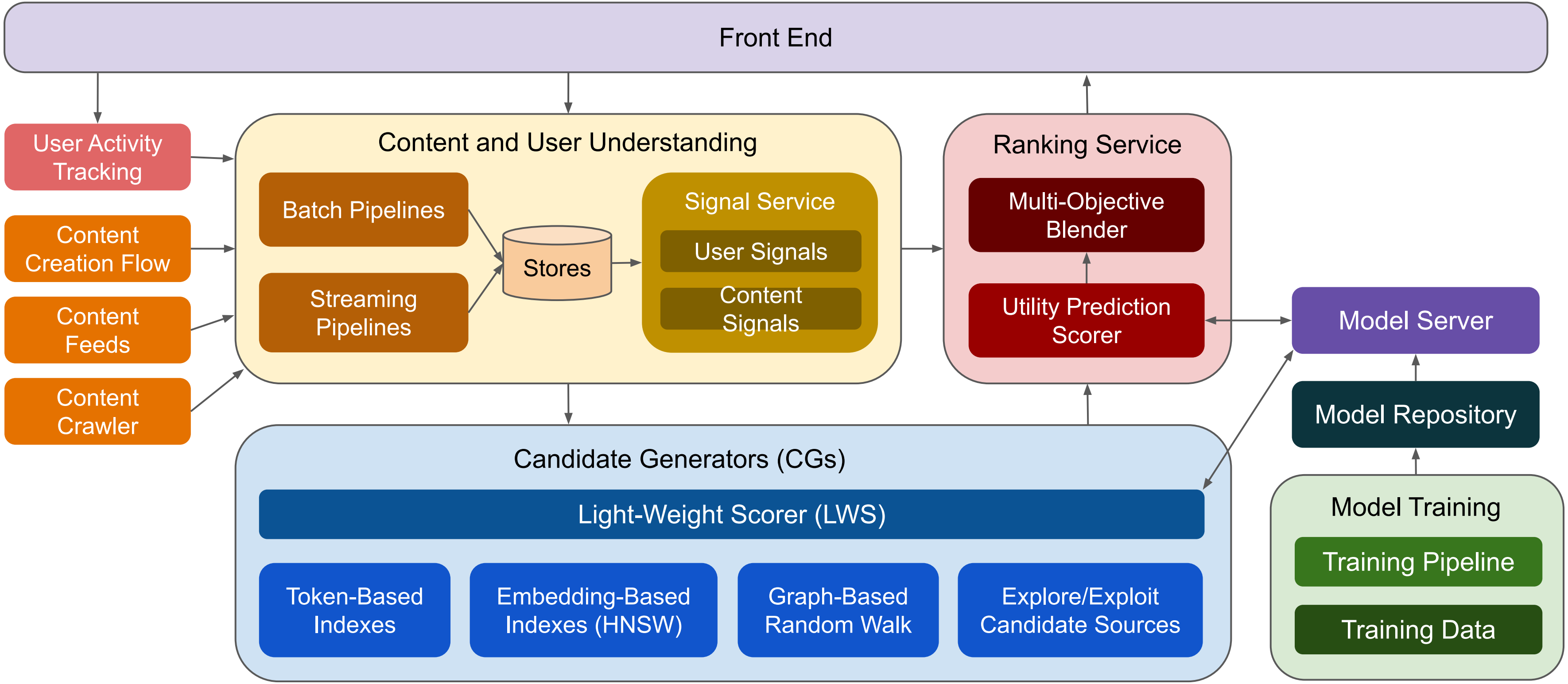


- Start from a set of pins that a user recently interacted with
- Perform random walks from this set of pins
- Return 000s of Pins with the highest visit frequencies (personalized PageRank scores)
- A lot of optimization to make the system highly performant

System Architecture for scalability - 00s of thousands of users and few billion Pins (content)



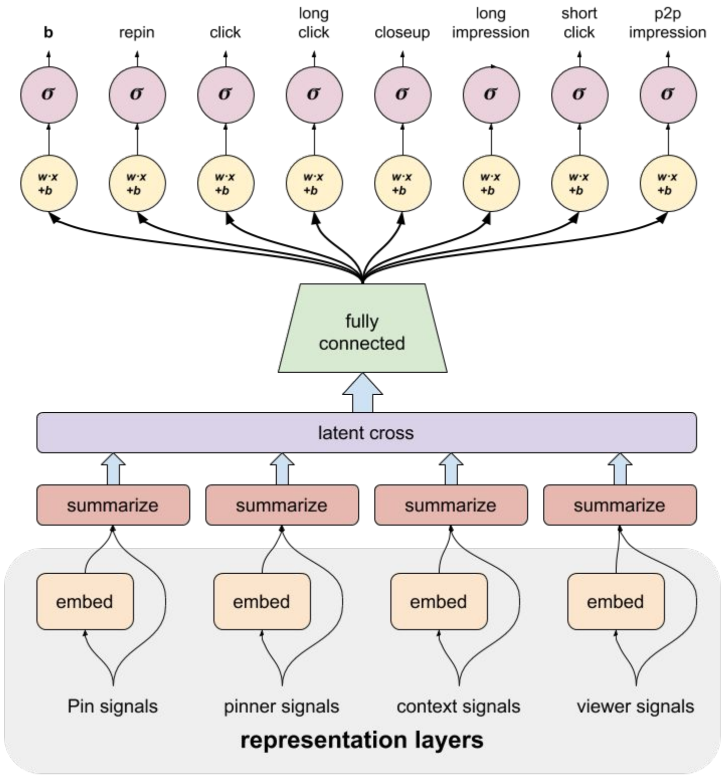
System Architecture for scalability - 00s of thousands of users and few billion Pins (content)



Pinterest

Ranking: User Action Prediction

- Predict a wide variety of user actions for each (user, item) pair through multi-head deep neural network
- User signals include the user's profile, interest vector, and embedding vector of the user's activity sequence (using Transformer)
- Content signals include the item's interest vector, engagement rate estimates, and graph embedding



Ranking: Multi-Objective Optimization

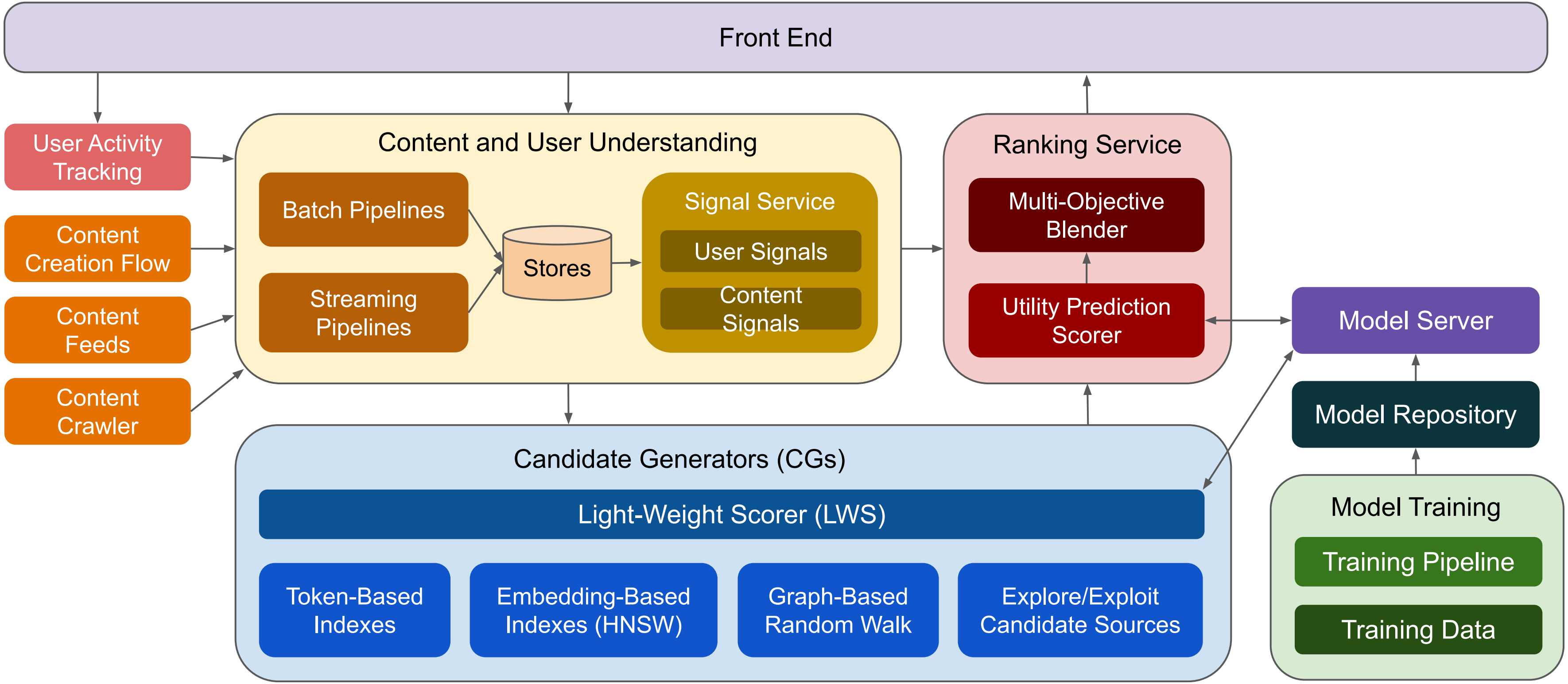
$$\begin{aligned} \max_{\boldsymbol{x}}\ & \text{PinnerUtility}(\boldsymbol{x}) \\ \text{s.t.}\ & \text{CreatorUtility}(\boldsymbol{x}) \geq \theta_1 \\ & \text{MerchantUtility}(\boldsymbol{x}) \geq \theta_2 \\ & \text{AdUtility}(\boldsymbol{x}) \geq \theta_3 \end{aligned}$$

$$\longrightarrow$$

$$\begin{aligned} \max_{\boldsymbol{x}}\ & \text{PinnerUtility}(\boldsymbol{x}) \\ & + w_1\ \text{CreatorUtility}(\boldsymbol{x}) \\ & + w_2\ \text{MerchantUtility}(\boldsymbol{x}) \\ & + w_3\ \text{AdUtility}(\boldsymbol{x}) \end{aligned}$$

- Estimate utility values for different parties on Pinterest based on predicted action probabilities and causal inference
- Use a simple weighted sum of utility terms
- Tune the weights to achieve a desired tradeoff
- Real system - several non-linearities are present

System Architecture for scalability - 00s of thousands of users and few billion Pins (content)



Deep Dive: Representation Learning

Observations

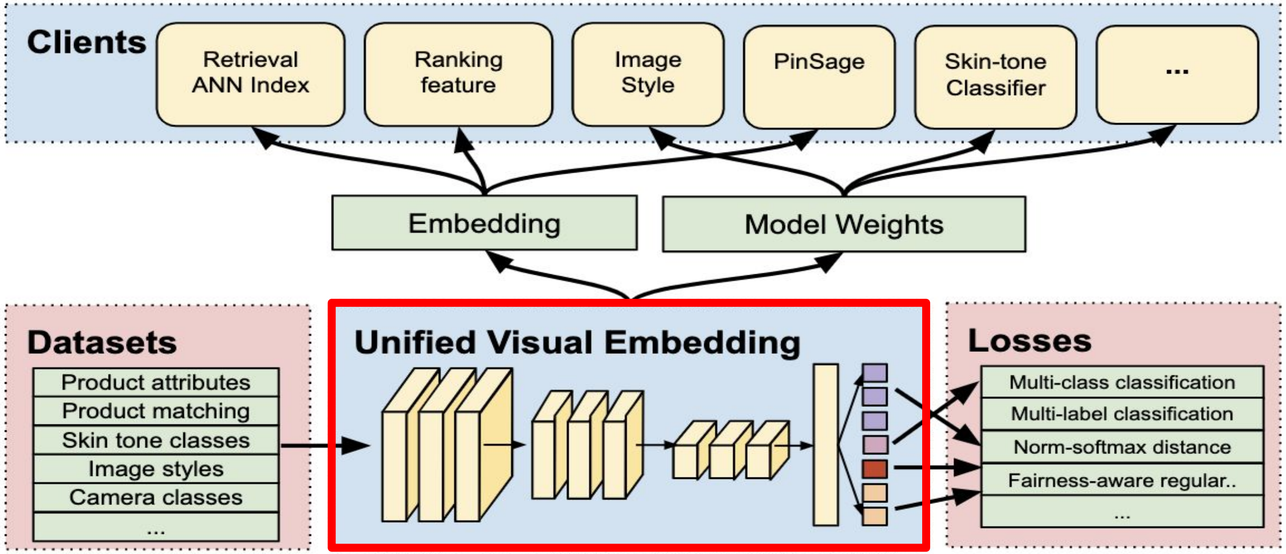
- Request time inference has **tight latency** requirements
  - Ranking scores >10M items per second, p99 < 20ms
  - Need to push complexity offline to signals

- Performance largely depends on how well we understand users, content, search queries, boards:
  - Ranking / retrieval is f(action | (user, context, content))

- Homefeed is 1 recommender system, we still have (search, related pins, board search, ..) x (organic, creator, shopping, advertisement, ..)
  - Need a way to reason about recommender and search systems more uniformly

Visual Embeddings

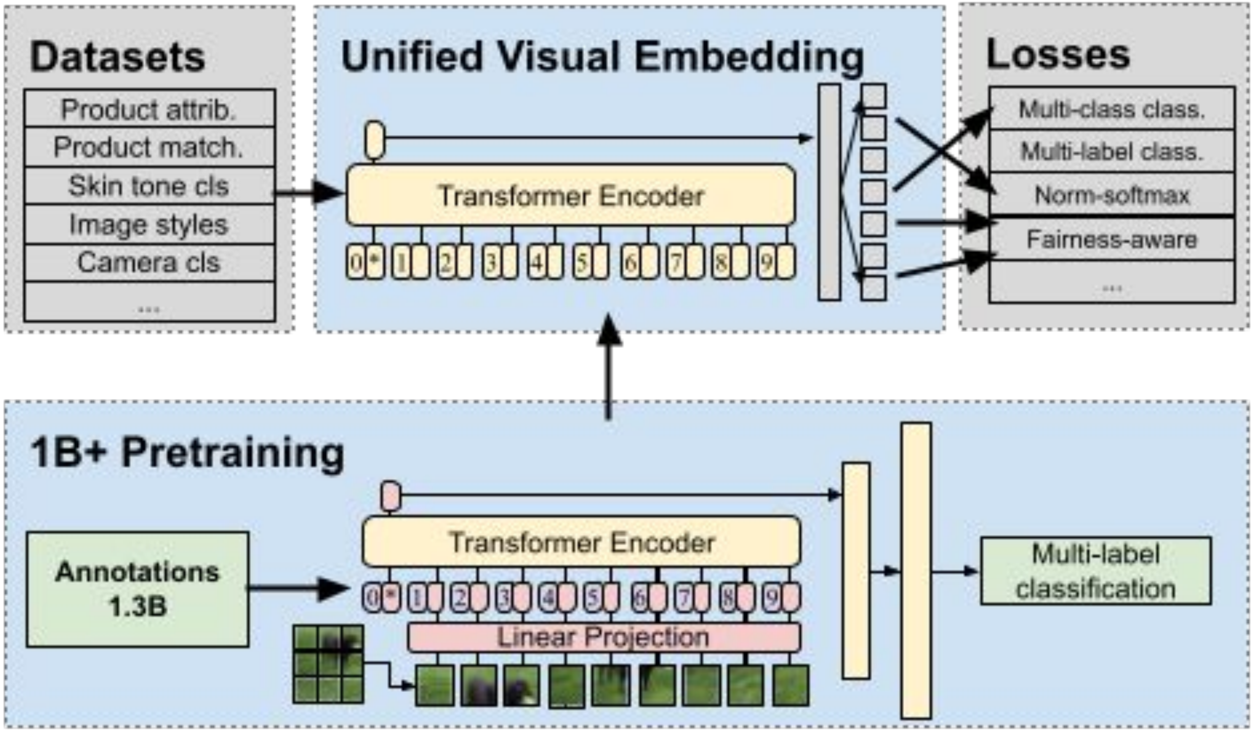
Zhai et al. "Learning a Unified Embedding for Visual Search at Pinterest", KDD'19



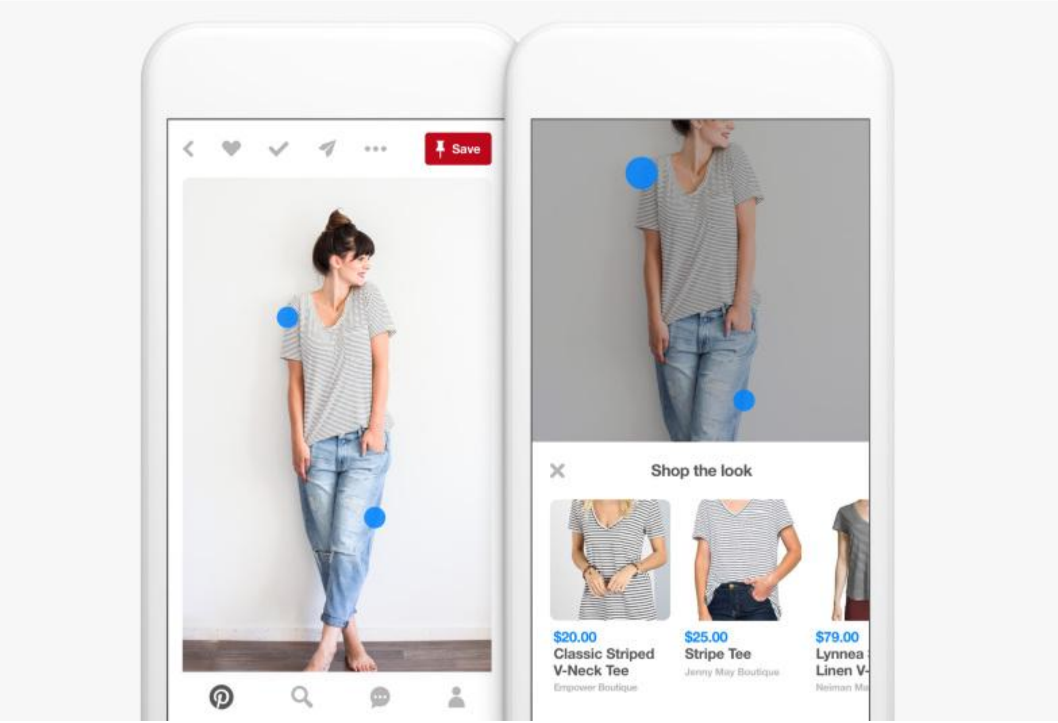
- Input: An image
- Output: embedding, classification, regression, …
- Multi-task training
  - **15+ objectives** across exact product matching, neardup, skin tone classifier

Pretraining

Billion-Scale Pretraining with Vision Transformers for Multi-Task Visual Representations, Beal. et al, WACV 2022



Results

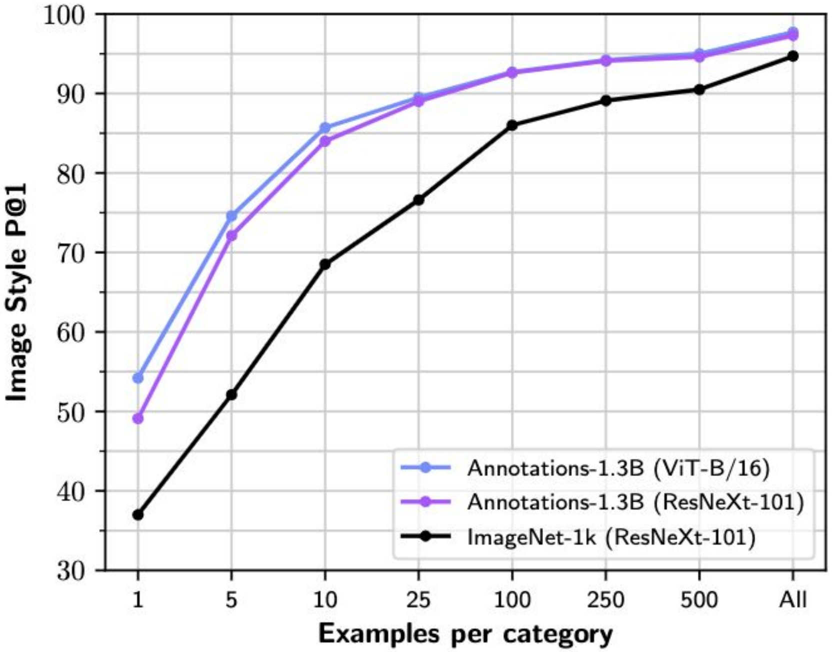


Largest online lifts in Visual Shopping this year

- +38% exact product match@1
- +24-33% long-clickers, and +29-30% long click

Few-Shot Learning

Billion-Scale Pretraining with Vision Transformers for Multi-Task Visual Representations, Beal. et al, WACV 2022

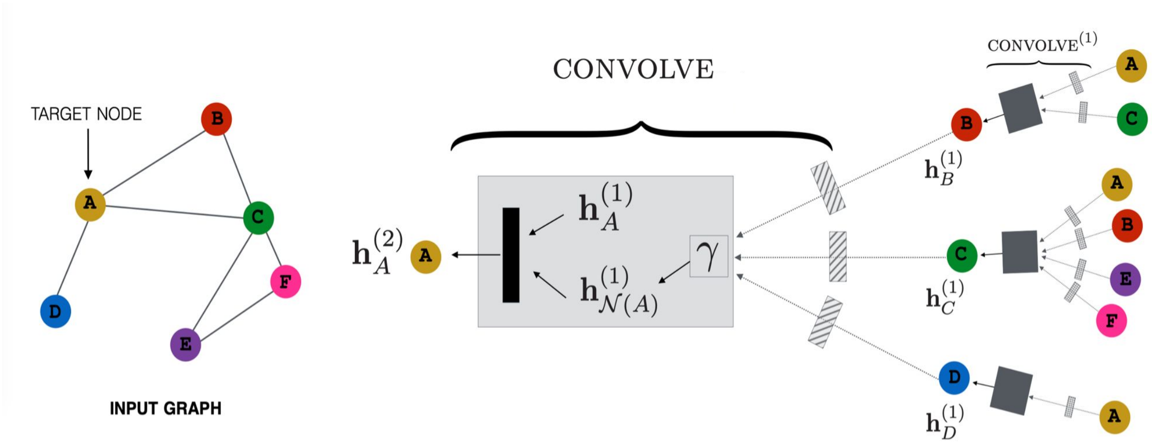


Pretraining serves as an effective multiplier on the labeled dataset size.

Pinterest

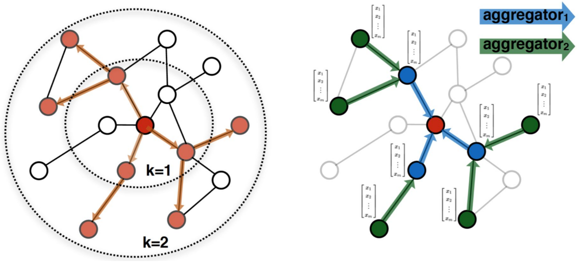
Pin Embeddings (PinSAGE)

Graph Convolutional Neural Networks for Web-Scale Recommender Systems, Ying et al., 2018

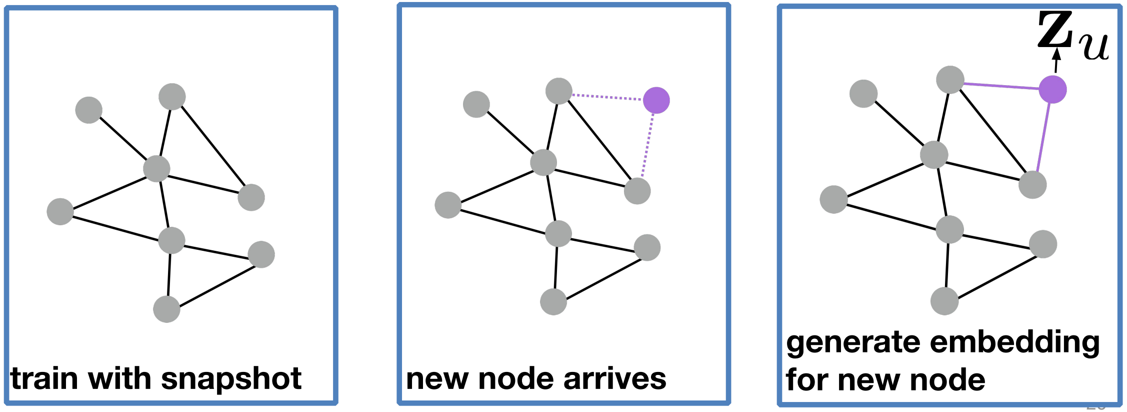


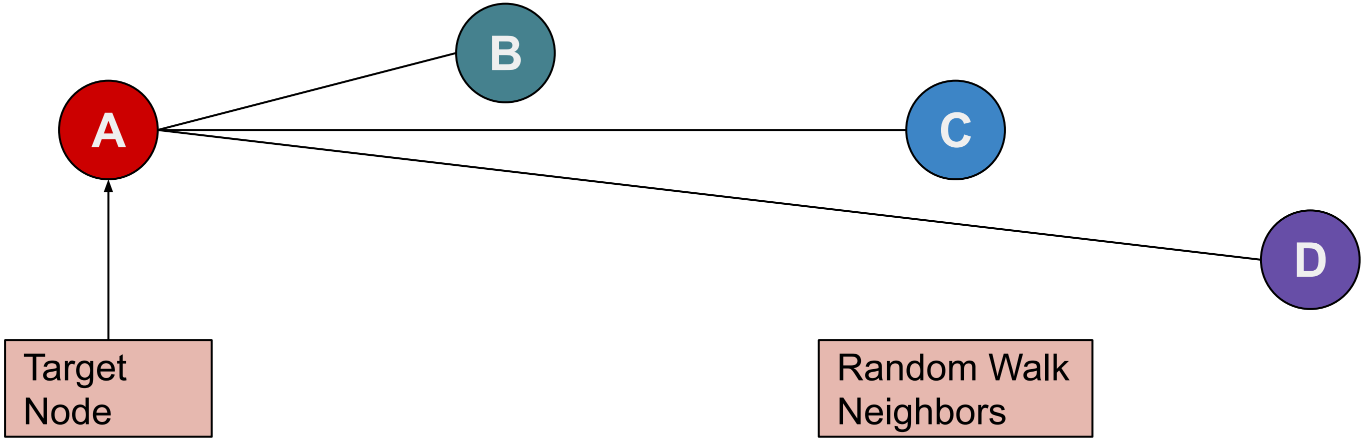
- A **Pin** is predominantly defined by its **content** and **engagement**
  - raw features (visual and text embedding, …) to represent content
  - pin-board graph to represent engagement
- **>100 use-cases** internally across retrieval, ranking, feature engineering, T&S, diversification for shopping, monetization, creators, etc.

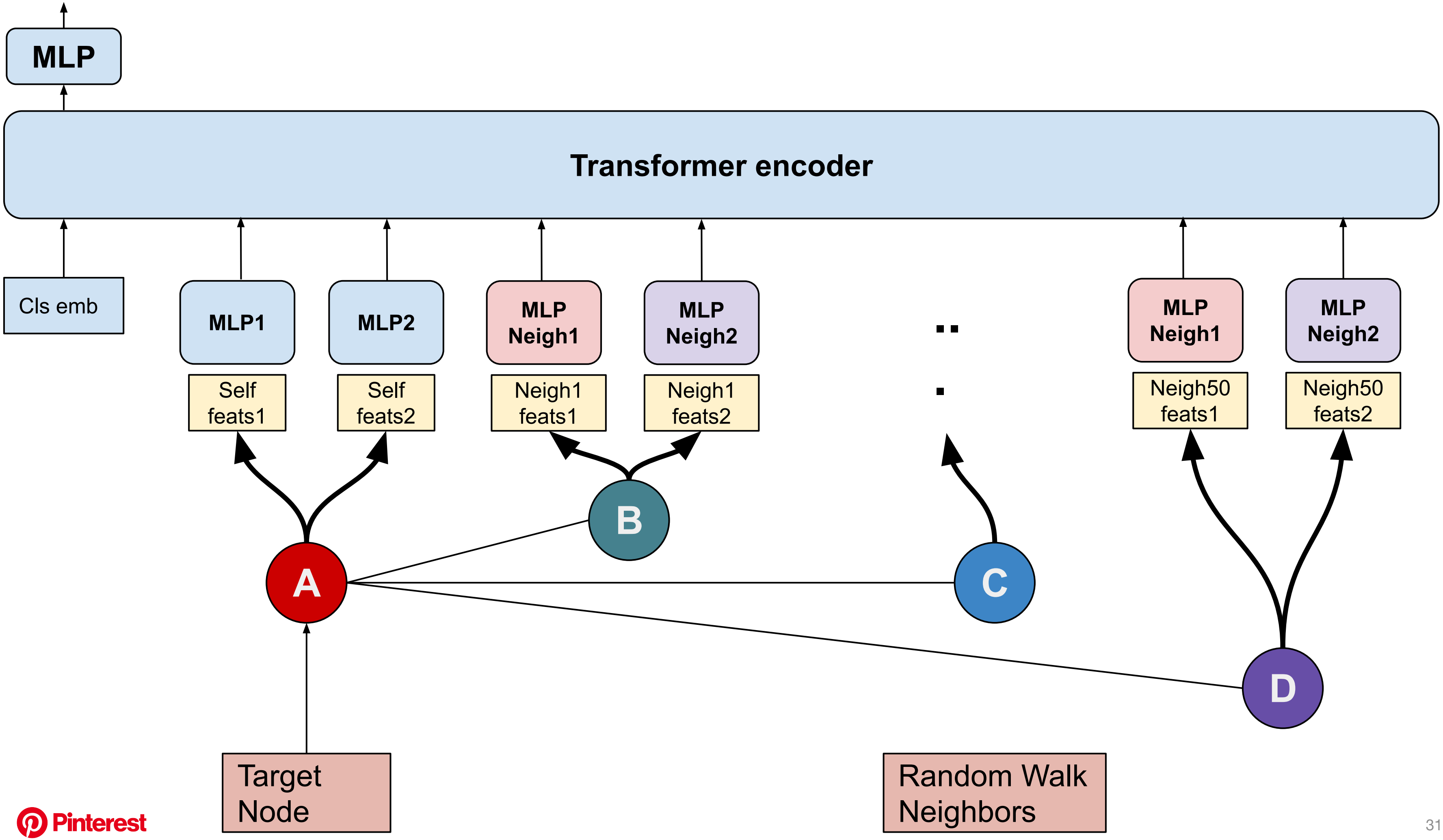
Inductive Learning for Adaptation



- **Dataset:** 3B nodes, 18B edges
- Two Hop Neighborhood Subsampling (V1)
- Random walk-based Neighborhood Subsampling (V2)
  - Approximates personalized PageRank (PPR) score
  - Sampled neighborhood for a node is a list of nodes with the top-K PPR score
- Represent node via context **features** not *unique id* for inductive inference




A
B
C
D
Target Node
Random Walk Neighbors

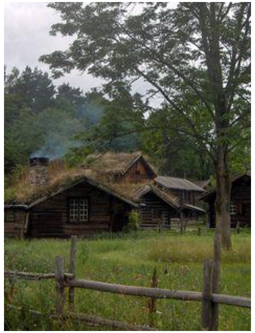

MLP
Transformer encoder
Cls emb
MLP1
MLP2
MLP Neigh1
MLP Neigh2
MLP Neigh1
MLP Neigh2
Self feats1
Self feats2
Neigh1 feats1
Neigh1 feats2
Neigh50 feats1
Neigh50 feats2
A
B
C
D
Target Node
Random Walk Neighbors

Pinterest

Softmax for retrieval loss

Query

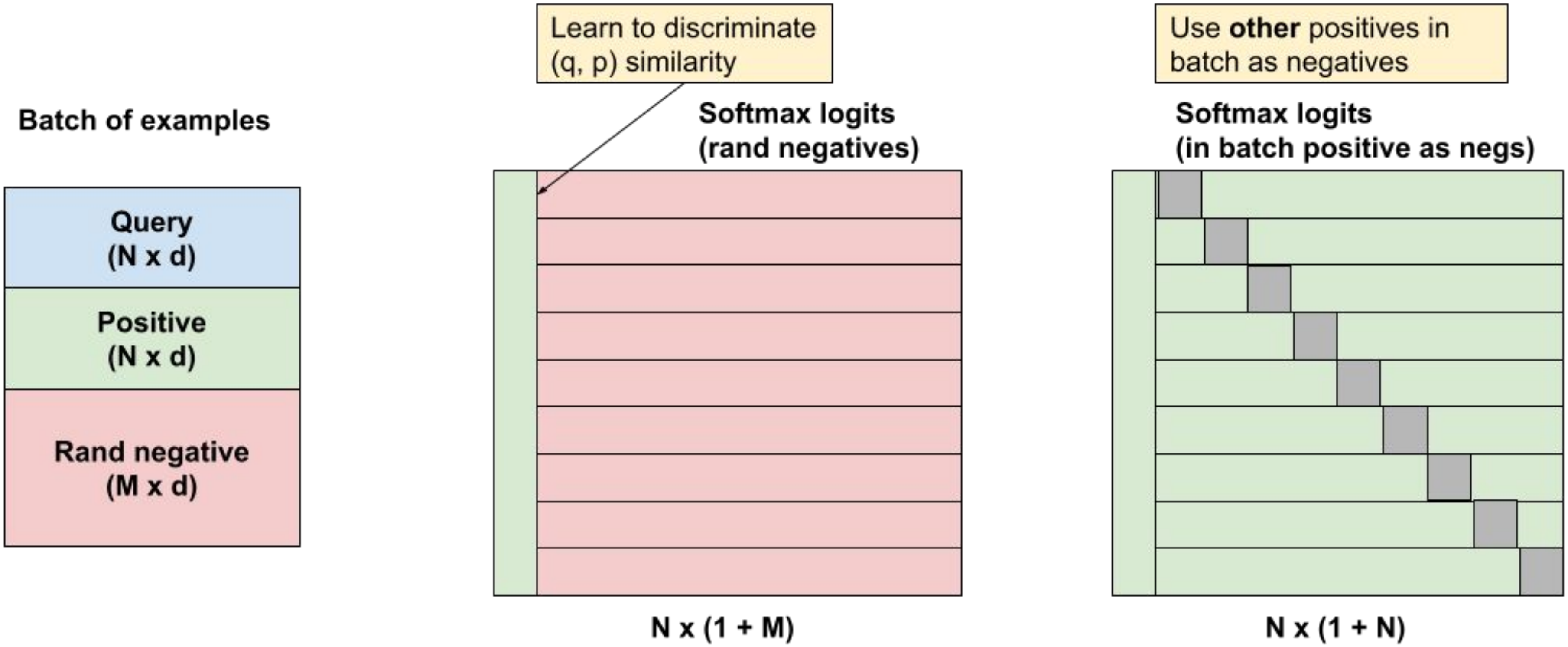
Positive



Negative

- We leverage a retrieval loss to learn meaningful embeddings
- Softmax to predict (q, p) similarity higher than (q, n) for all n $\in$ N
  - Not practical, |N| > 100M in practice
  - Use sampled softmax

Softmax for retrieval loss



Softmax for retrieval loss

- Probability correction is helpful to remove serving bias
- Sampled softmax logits are predicting $\log(P(y \mid x_{i,} C_i))$ (class probabilities over **sampled** classes)
- With some assumptions (e.g. each class is independently sampled), you get:

$$\log P(y \mid x_{i,} C_i) = \log P(y \mid x_i) - \log Q(y \mid x_i) + K(x_i, C_i)$$

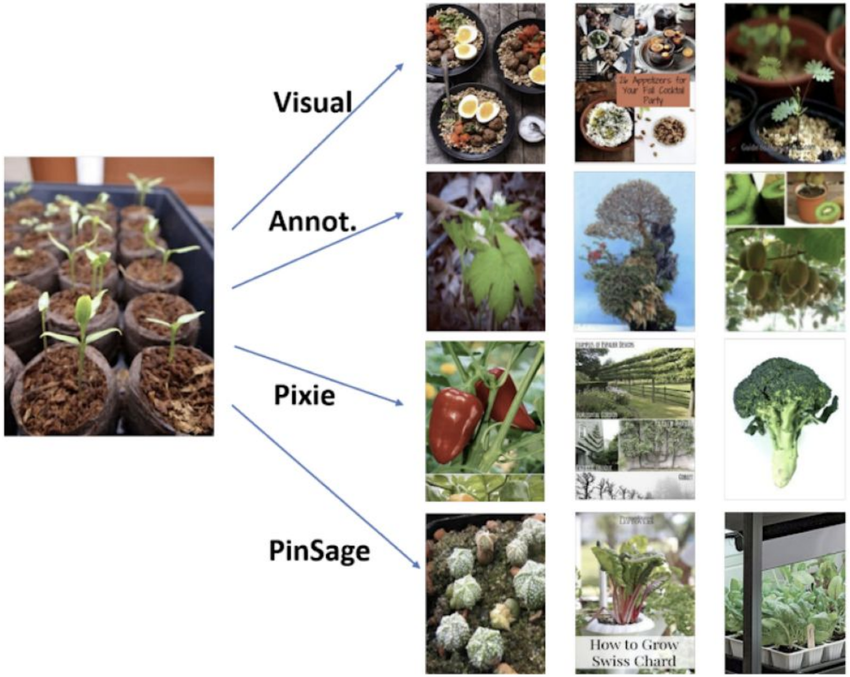
Class sampling probability

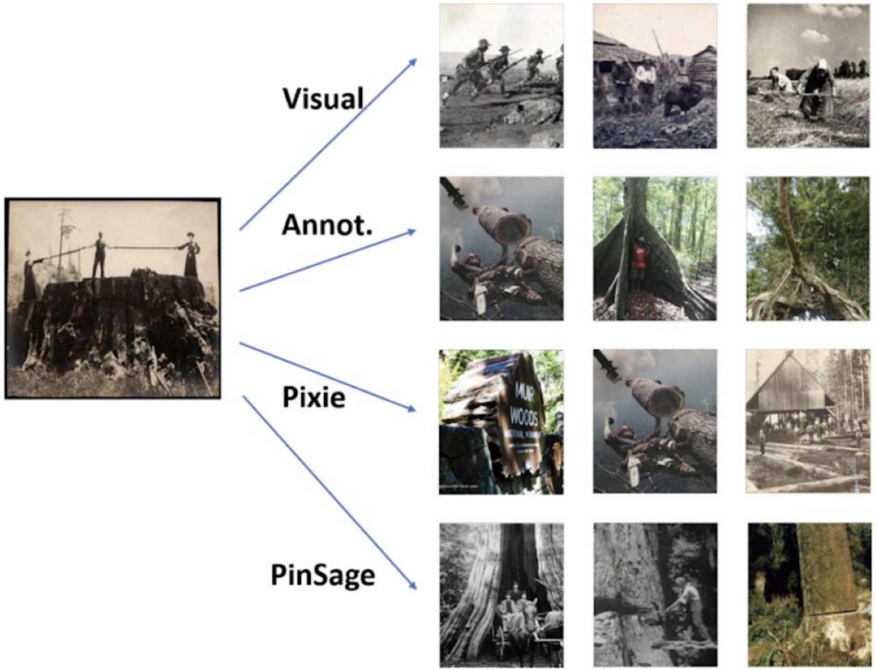
Does not depend on class so normalizes out in softmax

For a more rigorous treatment:
https://www.tensorflow.org/extras/candidate_sampling.pdf
http://arxiv.org/abs/1412.2007

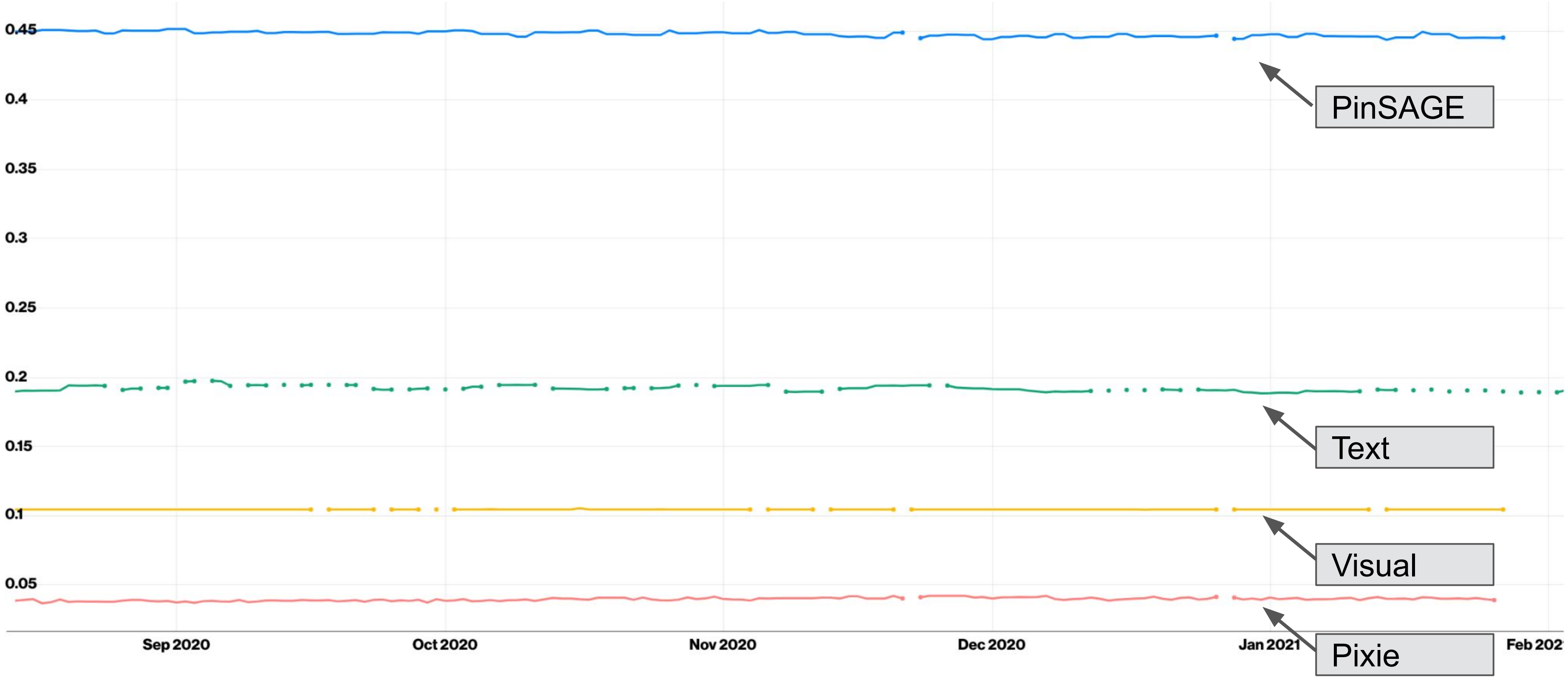
Pin Embeddings (PinSAGE)

Graph Convolutional Neural Networks for Web-Scale Recommender Systems, Ying et al., 2018





Recall@1: PinSAGE is performant, and stable



User Signals: User Embeddings

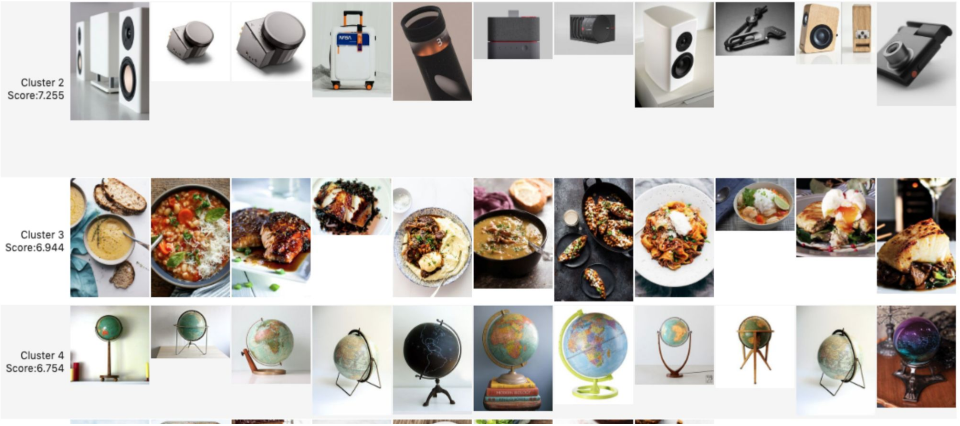
- Our recommender systems already leverage **id** features learned jointly (e.g. in ranking)

- We want a content based signal for users that's more **semantic** and **adapts** online

User Embeddings

PinnerSage: Multi-Modal User Embedding Framework for Recommendations at Pinterest, Pal et al., KDD 2020

PinnerSAGE:
(clustering)



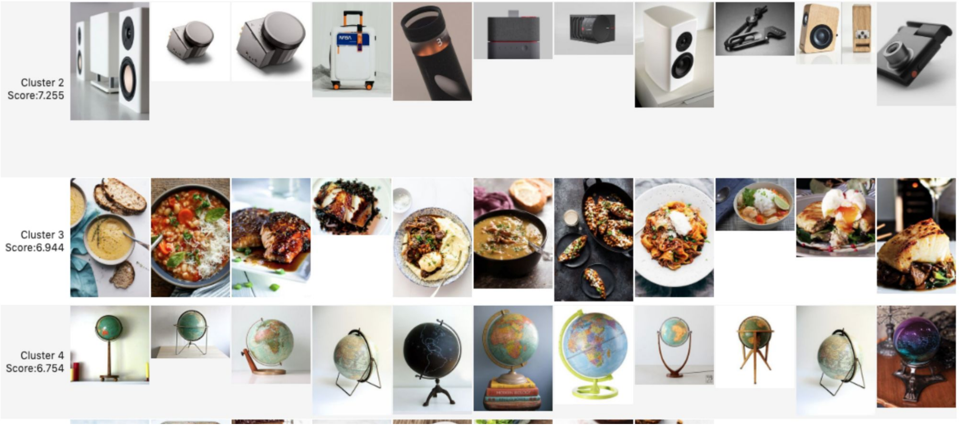
User Emb 1

...

User Emb k

User Embeddings

PinnerSAGE:
(clustering)

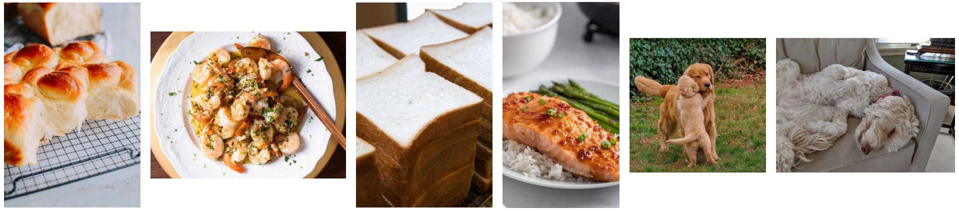


User Emb 1

...

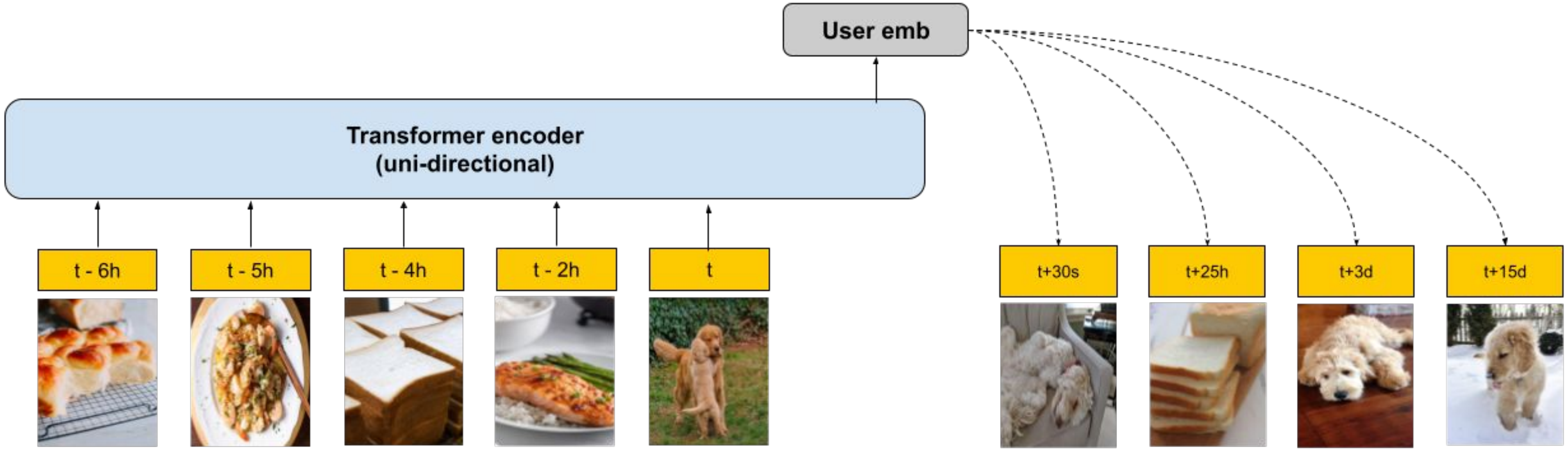
User Emb k

New user embedding (Supervised Transformer)



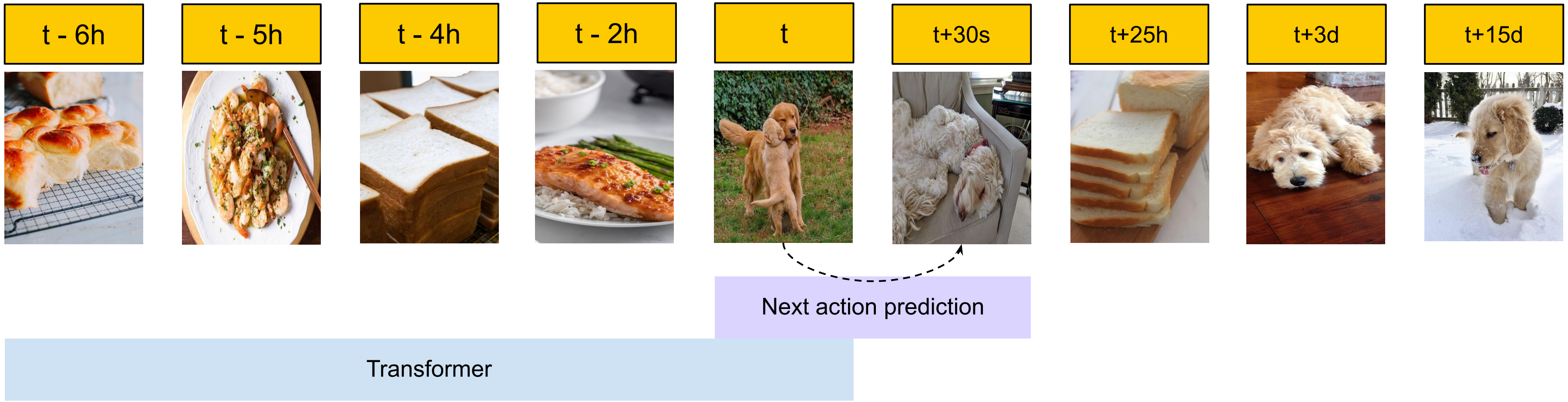
User Emb

User Signals: User Embeddings



- Input: Last K user activity sequence across all of Pinterest
- Output: one user embedding summarizing activity jointly for short and long-term activity prediction.
- O(100M vocab) for "action" on item - **softmax retrieval loss** and **PinSage**

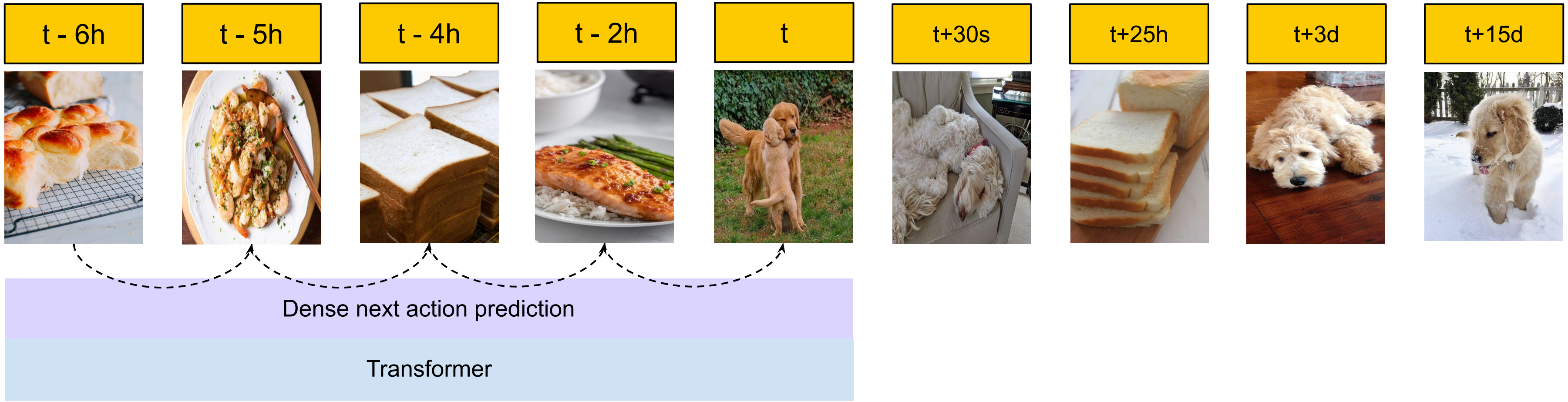
Training Objective: Next Action



Prediction = one example to softmax retrieval

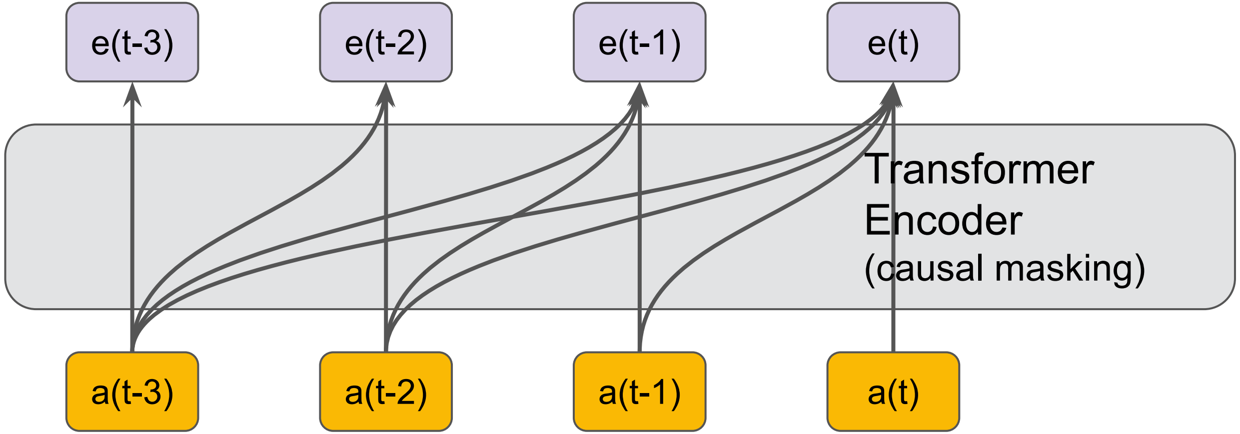
(user, next action positive, random negatives)

Training Objective: “Dense” next Action

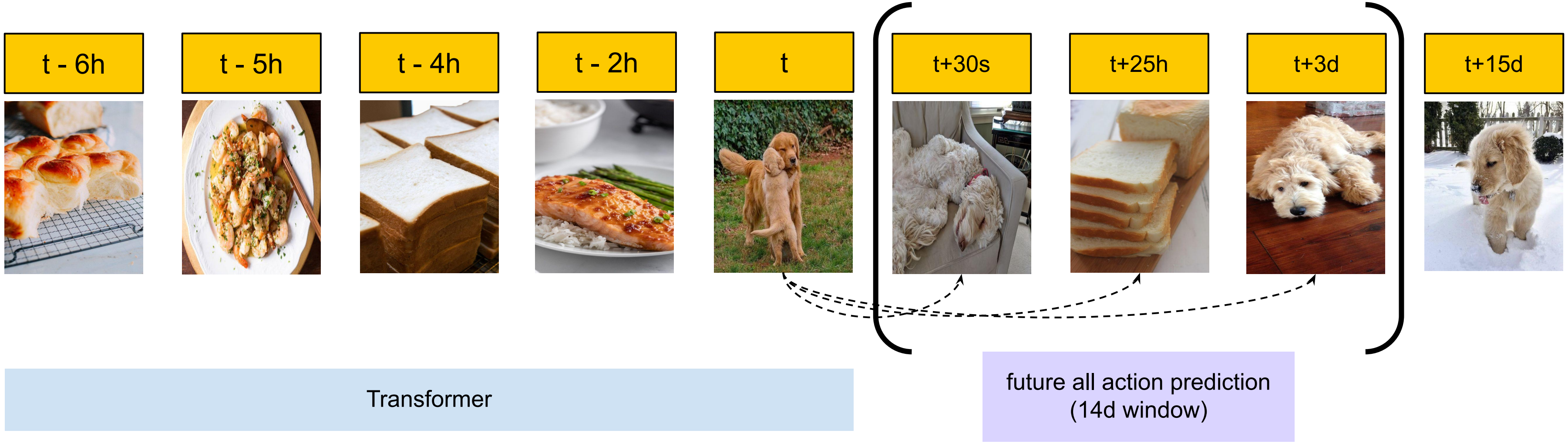


Training Objective: “Dense” Next Action

- e(t-1) predicts action(t)
- Only attend to previous actions



Training Objective: All Action

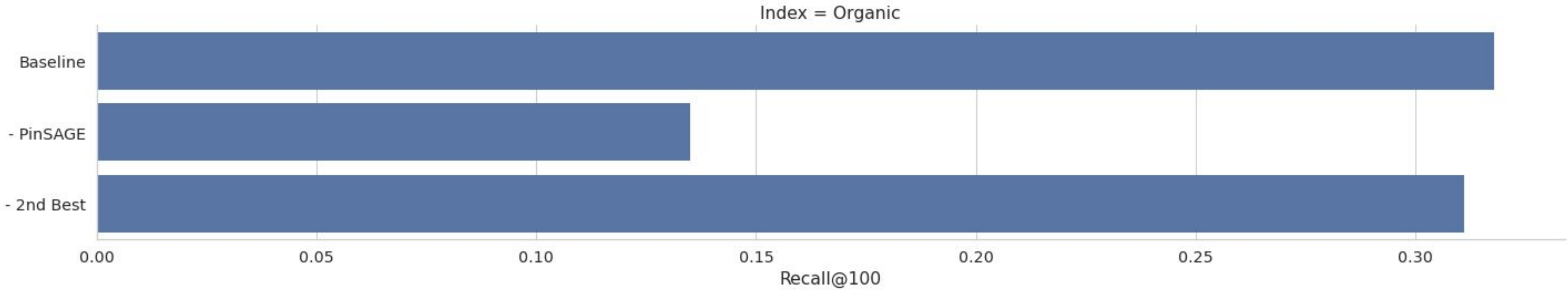


User Embedding Results

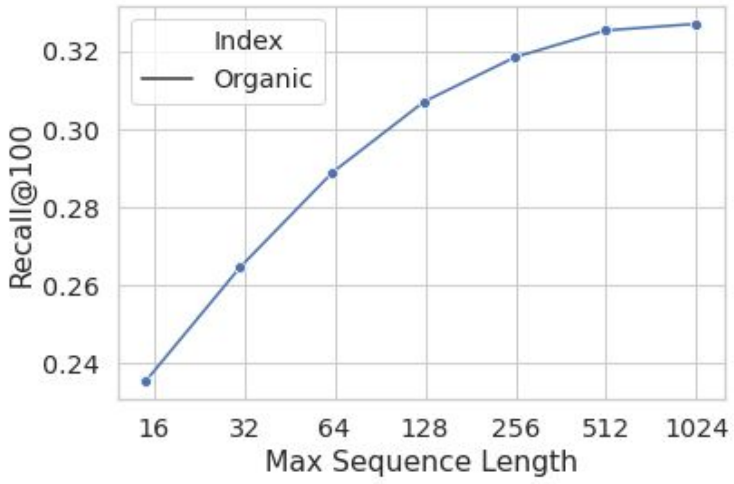
	all_action R@100
(oracle) PinnerSAGE (5 clusters)	0.125
(oracle) PinnerSAGE (20 clusters)	0.205
Our method (1 embedding)	**0.255**

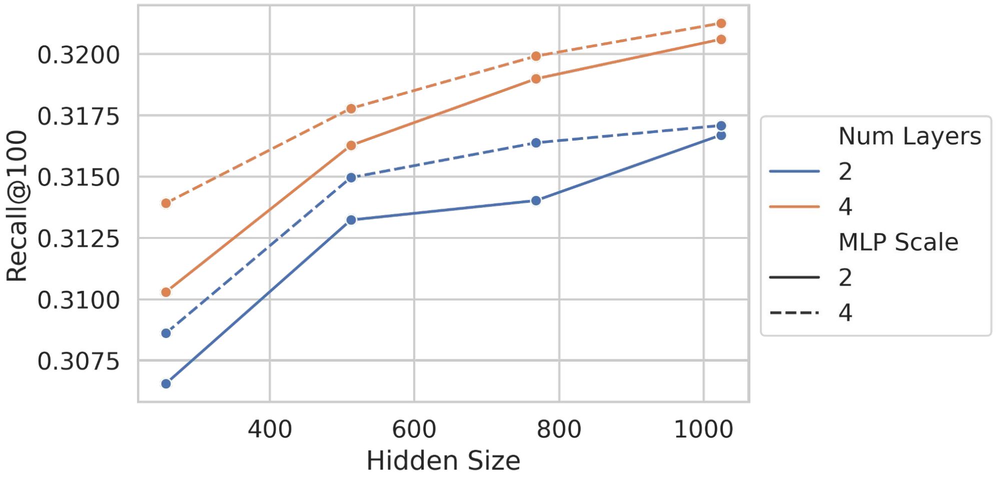
- Online experiment in Homefeed ranking, replacing prior method
  - **+1-2%** timespent on Pinterest, **+3-4%** engagement lift, **-2.6%** content hides. Wins across in shopping, creators, organic

Ablation: Feature Importance



Ablation: Larger is better





Pinterest

Summary

- Representations are critical to recommendation performance
  - Often times where the most complex ML models reside
- Large capacity models with **huge** data is very useful
- We use **Transformers** everywhere, general feature interaction module
- Build on top of each other (Visual -> PinSage -> PinnerSage)
  - Separate models due to training cost
- Team jointly optimizes **vertically** (users, pin, image, video, text, creator, products, ...) and **horizontally** (softmax retrieval, transformers, …)

Come join us! https://www.pinterestcareers.com/