Bring **everyone** the **inspiration** to create a life they love
The perfect path to cold brew
Caffeinated Inc.
Omar Seyal
Cravings
Andrew Zhai

A greater collection of ideas.
People on Pinterest each month: 400m+

- Pins: 330b+
- Boards: 7b+
- Languages: 35+
- 91% say Pinterest is a place filled with Positivity
Inspirational Engagement

Top pins by view time

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

Front End

User Activity Tracking

Content Ingestion

Content and User Understanding

Batch Pipelines

Streaming Pipelines

Stores

Signal Service

User Signals

Content Signals

Ranking Service

Multi-Objective Blender

Utility Prediction Scorer

Candidate Generators (CGs)

Light-Weight Scorer (LWS)

Token-Based Indexes

Embedding-Based Indexes (HNSW)

Graph-Based Random Walk

Explore/Exploit Candidate Sources
Predict a wide variety of user actions for each (user, item) pair through multi-head deep neural network.
Multi Objective Optimization

\[
\max_x \text{PinnerUtility}(x) \\
\text{s.t. CreatorUtility} \geq \theta_1 \\
\text{MerchantUtility}(x) \geq \theta_2 \\
\text{AdUtility}(x) \geq \theta_3
\]

\[
\max_x \text{PinnerUtility}(x) \\
\quad + w_1 \text{CreatorUtility}(x) \\
\quad + w_2 \text{MerchantUtility}(x) \\
\quad + w_3 \text{AdUtility}(x)
\]

- Estimate utility values for different parties on Pinterest based on predicted action probabilities
- Tune the weights to achieve a desired tradeoff
- Real system - Functional form contains non-linearities are present
System Architecture for scalability - 00s of thousands of users and few billion Pins (content)

Front End

User Activity Tracking

Content Ingestion

Content and User Understanding

Batch Pipelines

Streaming Pipelines

Stores

Signal Service

User Signals

Content Signals

Ranking Service

Multi-Objective Blender

Utility Prediction Scorer

Candidate Generators (CGs)

Light-Weight Scorer (LWS)

Token-Based Indexes

Embedding-Based Indexes (HNSW)

Graph-Based Random Walk

Explore/Exploit Candidate Sources
Determining visual similarity

Distance Function

Similarity (A,B)
Embeddings

Encode very different types of data (images, pin, user)
Application 1

Application 2

Application 3

Application N

Color
blue=38.78,
purple=16.46,
green=11.07,
black=8.26,
multi=7.52

Fabric
cotton=42.24,
chiffon=14.61,
silk=7.90,
wool=5.61
**“Unified” Visual Backbone**

Output: **20+ tasks** across exact product matching, neardup, skin tone classifier

**Benefits**
- Scalable maintainence (most important)
- Joint learning across dataset
- Share foundational improvements

Zhai et al. “Learning a Unified Embedding for Visual Search at Pinterest”, KDD’19
## “Unified” Visual Backbone

Zhai et al. “Learning a Unified Embedding for Visual Search at Pinterest”, KDD’19

<table>
<thead>
<tr>
<th>Model</th>
<th>STL P@1</th>
<th>Flashlight Avg P@20</th>
<th>Lens Avg P@20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Old Shop-the-Look</td>
<td>33.0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Old Flashlight</td>
<td>-</td>
<td>53.4</td>
<td>-</td>
</tr>
<tr>
<td>Old Lens</td>
<td>-</td>
<td>-</td>
<td>17.8</td>
</tr>
<tr>
<td>ImageNet</td>
<td>5.6</td>
<td>33.1</td>
<td>15.0</td>
</tr>
<tr>
<td>Ours</td>
<td>52.8</td>
<td>60.2</td>
<td>18.4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>STL P@1</th>
<th>Flashlight Avg P@20</th>
<th>Lens Avg P@20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shop-the-Look (S)</td>
<td>49.2</td>
<td>42.1</td>
<td>14.7</td>
</tr>
<tr>
<td>Flashlight (F)</td>
<td>11.0</td>
<td>53.4</td>
<td>16.1</td>
</tr>
<tr>
<td>Lens (L)</td>
<td>26.2</td>
<td>47.8</td>
<td>18.2</td>
</tr>
<tr>
<td>All (S + F + L)</td>
<td>52.8</td>
<td>60.2</td>
<td>18.4</td>
</tr>
</tbody>
</table>

Multi-Task Embedding > Single-Task Embedding
All Dataset > Single Dataset
Billion-Scale Pretrain
Lifts All

Pretrain
● 1.3B image pretraining
● Funnel Hybrid ViT

Finetune
Billion-Scale Pretrain Lifts All

<table>
<thead>
<tr>
<th>Model</th>
<th>Pretraining</th>
<th>VS</th>
<th>F</th>
<th>L</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>RN-101</td>
<td>IN-1k</td>
<td>39.6</td>
<td>59.7</td>
<td>17.2</td>
<td>85.2</td>
</tr>
<tr>
<td>RN-101</td>
<td>IG-940M</td>
<td>46.7</td>
<td>67.6</td>
<td>20.2</td>
<td>87.9</td>
</tr>
<tr>
<td>RN-101</td>
<td>ANN-1.3B</td>
<td>52.4</td>
<td>70.8</td>
<td>22.7</td>
<td>88.8</td>
</tr>
<tr>
<td>ViT-B/32</td>
<td>IN-1k</td>
<td>29.2</td>
<td>44.7</td>
<td>15.2</td>
<td>82.3</td>
</tr>
<tr>
<td>ViT-B/32</td>
<td>ANN-1.3B</td>
<td>46.4</td>
<td>68.9</td>
<td>24.9</td>
<td>86.5</td>
</tr>
<tr>
<td>ViT-B/16</td>
<td>ANN-1.3B</td>
<td>54.7</td>
<td>74.3</td>
<td>26.7</td>
<td>89.7</td>
</tr>
</tbody>
</table>

Billion-scale Pretraining Lifts majority of application performance
Challenge: How to represent all dimensions of our content?
The perfect path to cold brew

Caffeinated Inc.

Omar Seyal

Cravings

Image

Title

The perfect path to cold brew

Creator

Omar Seyal

Pin-board Graph
Harnessing the Pin Board Graph
PinSAGE: Graph Neural Network

Graph with 3 billion nodes and 18 billion edges

Graph Convolutional Neural Networks for Web-Scale Recommender Systems, Ying et al., 2018
From pin **features and graph**, encode into **embeddings** trained so pins that are “related” have **similar** embeddings

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

PinSage V1 (~triplet loss)

\[ L = \frac{1}{|D|} \sum_{(q,p,n) \in D} \max(0, e_q^T e_n - e_q^T e_p + m) \]
V1: Graph Sampling on the Fly

- **Sample Method**: K-hop neighborhood sampling
  - Pin -> board -> pin
- **Train Infra**: Graph sampling on the fly
  - 1.5TB RAM GPU machine (custom hardware)
  - Only 2 available at Pinterest….
- **Inference Infra**: Hardwire architecture as Hadoop Jobs
V1: Graph Sampling on the Fly

- **Sample Method**: K-hop neighborhood sampling
  - Pin -> board -> pin
- **Train Infra**: Graph sampling on the fly
  - 1.5TB RAM GPU machine (custom hardware)
  - Only 2 available at Pinterest….
- **Inference Infra**: Hardwire architecture as Hadoop Jobs

**Pro:**
- It works! Best performing content embedding at 3B nodes and 18B edges scale

**Con:**
- Not scalable to more developers nor flexible for iterations
- Train & serve completely separate stacks
V2: Offline Graph Sampling

- Scalability challenges due to graph sampling on the fly
  - Solution: Move sampling out of training / inference

- Sample Method: Random walks (50 neighbors)

- Data Prep:
  - Compute $3B \times 50$ random walk in a daily workflow
  - Materializes self + neighbor features for each pin example

- Train & Inference Infra:
  - Stream example through model
V2: Offline Graph Sampling

- Scalability challenges due to graph sampling on the fly
  - **Solution**: Move sampling out of training / inference

- **Sample Method**: Random walks (50 neighbors)

- **Data Prep**:
  - Compute $3B \times 50$ random walk in a daily workflow
  - Materializes `self + neighbor features` for each pin example

- **Train & Inference Infra**:
  - Stream example through model

---

**Pro:**
- Leverage commodity hardware
- +46% offline performance

**Con:**
- Harder to iterate on graph sampling algorithm
V3: Multi-Task GNN Transformer

- Multi-Task - 16 objectives to optimize different content formats
- TransformerEncoder - why not early fuse neighbor and self features?
GNNs produce the Best Content Representation

70+ launches across recommendation systems, T&S, knowledge understanding, shopping, advertisement, ...

- PinSAGE
- Text
- Visual
- Random Walk
User Modeling
PinnerSage

Hierarchical Clustering (WARD)

declutter

Interest embeddings (medoid)

Pro:
- Simple and effective. 10+ launches (e.g. +3% HF repin/click volume)
- Interpretable, debuggable

Con:
- Multiple embeddings challenging to use
- No parameter sharing across users
- No explicit objective learning
User sequence activity for past year
PinnerFormer

Random Split

P2P: click pinid X

HF: repin pinid Y

Search: repin pinid Z
Encode last $K$ actions. $K=255$ currently

- P2P: click pinid X
- HF: repin pinid Y
- Search: repin pinid Z
PinnerFormer

Encode last 255 actions

Predict actions

P2P:click pinid X

HF:repin pinid Y

Search:repin pinid Z
**Input**: Last K user activity sequence across all of Pinterest

**Output**: one user embedding summarizing activity jointly for **short** and **long-term** activity prediction.
**Dense All Action** leads to best performance
- Optimize for all pos actions within 28d, densely across input seq to Transformer
## PinnerFormer Results

<table>
<thead>
<tr>
<th>Model Description</th>
<th>R@100</th>
</tr>
</thead>
<tbody>
<tr>
<td>(oracle) PinnerSAGE (5 clusters)</td>
<td>0.125</td>
</tr>
<tr>
<td>(oracle) PinnerSAGE (20 clusters)</td>
<td>0.205</td>
</tr>
<tr>
<td>PinnerFormer (1 embedding)</td>
<td>0.255</td>
</tr>
</tbody>
</table>

### Site-wide impact
- +1-2% timespent
- +3-4% repins
- -2.6% hides
- +1.8% revenue

10+ launches
Personalized Ranking
Predict a wide variety of user actions for each (user, item) pair through multi-head deep neural network

Combine 100s of features, served on CPU
Ranking: User Action Prediction

- Two Trends for Performance:
  - Increase parameters, complexity for model expressivity
### Ranking: Scaling It Up

<table>
<thead>
<tr>
<th>Model</th>
<th>Expected Saves Gain</th>
<th>Latency Increase</th>
</tr>
</thead>
<tbody>
<tr>
<td>2x Wider Fully Connected</td>
<td>5%</td>
<td>+10%</td>
</tr>
<tr>
<td>+ Transformers</td>
<td>4%</td>
<td>+300%</td>
</tr>
</tbody>
</table>
Ranking: User Action Prediction

- Two Trends for Performance:
  - **Increase parameters, complexity** for model expressivity
  - **End-to-end learn** from raw (er) features
Ranking: User journey modeling (E2E Learning)

<table>
<thead>
<tr>
<th>Model</th>
<th>Expected Saves Gain</th>
<th>Latency Increase</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ RT activity seq (early fuse)</td>
<td>9%</td>
<td>+100%</td>
</tr>
</tbody>
</table>
**Ranking: User Action Prediction**

- **Two Trends for Performance:**
  - Increase parameters, complexity for model expressivity
  - End-to-end learn from raw (er) features

- **Challenge:**
  - Latency (~10ms P99)
  - Throughput (~10M inferences / sec)
  - Cost (+10% latency ~ $400k / year)
Ranking: GPU serving
Save Volume Lift

- First ML model
  - 2014: 6%
  - 2016: 7%

- GBDT model
  - 2017: 7%

- NN model
  - 2018: 7%

- Multitask NN model
  - 2020: 8%

- PinSAGE
  - 2021: 11%

- PinnerFormer
  - 2020: 8%

- Wide Networks
  - 2021: 5%

- Transformer
  - 2021: 4%

- RT Activity Sequence
  - 2021: 8%
Challenges
ML Systems are Dynamic

- Model degrades over time (e.g. Concept Drift)
- **Retraining** recovers performance
- Evaluating a “Good” model is at least 2-dimensional
ML Systems are Dynamic

- In practice, long chains of model dependencies
- What is the ABI for ML models?
Curse of the Power Law Distribution

- Power law distributions exist for both users and content
  - Not much feedback for majority of content and users
- Methods
  - Dataset Sampling
  - Explore-Exploit
  - Counterfactual Learning
  - Content/User Embeddings
  - Self Supervision
  - ...
Dataset is an Important Lever

- **Research**: model-centric
- **Industry**: data-centric
- **Trends**: Software 2.0, Data-centric ML
- **How can we build systems and algorithms to iterate on datasets faster?**
## User Journey Optimization

To maximize long-term “reward”

<table>
<thead>
<tr>
<th>Aspiration</th>
<th>Inspiration</th>
<th>Consideration</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Off platform: I want to remodel my kitchen</td>
<td>Search for “kitchen remodel” in Pinterest</td>
<td>● Save lifestyle images to boards ● Explore products in shopping</td>
<td>Purchase products for my kitchen to complete my remodel</td>
</tr>
</tbody>
</table>

- **User problem**: Want to find inspiration and complete project (e.g. summer vacation planning, cooking dinner). If Pinterest does well, plan more of life on Pinterest.
- **Today**: Utility function of immediate actions (e.g. save, click, closeup, hides).
  - Manual “gradient descent” (analysis, implement, ab experiment, feedback)
User Journey Optimization
To maximize long-term “reward”

- **User problem**: Want to find inspiration and complete project (e.g. summer vacation planning, cooking dinner). If Pinterest does well, plan more of life on Pinterest.
- **Today**: Utility function of immediate actions (e.g. save, click, closeup, hides).
  - Manual “gradient descent” (analysis, implement, ab experiment, feedback)
- **Challenge**: Enable ML systems to optimize directly for “pinner satisfaction”
  - Causal inference for actions -> long-term satisfaction?
  - Off-policy Reinforcement Learning?
    - Reward function incredibly complex from multi-objective optimization
Next Gen Inspirational AI Products

Multimodal Conditional Image Synthesis with Product-of-Experts GANs 2021

A bowl of soup that is a portal to another dimension as digital art

https://openai.com/dall-e-2/
A lot more going on…

- **Representation Learning** for videos, products, creators, search queries, notifications
- **Web Mining** through GNNs to extract attributes (e.g. recipe for food pins) from websites to create rich content at scale
- **Inspirational Knowledge Graph** to enable a vocabulary to communicate between ML and users to assist their journey
- **Learned Retrieval** to holistically learn candidate generation for recommendations and search
- **Notification Uplift Modeling** to learn the optimal intervention policy for share inspiration to Pinners outside of Pinterest
Takeaways

- **Pinterest** is a unique curated dataset of how people describe and organize things.
- **ML** is leveraged throughout our inspiration funnel to enable us to bring *everyone* the *inspiration* to create a life they love.
- **Deep Learning methods** (Transformers, GNN, Sequence) leading the way for performance.
- **Scalability** of systems and ML algorithms are baked deeply into our culture and a continued trend for improvement.
- A lot of technical **challenges** exist. Not even close to a solved problem.
Thank you!

andrew@