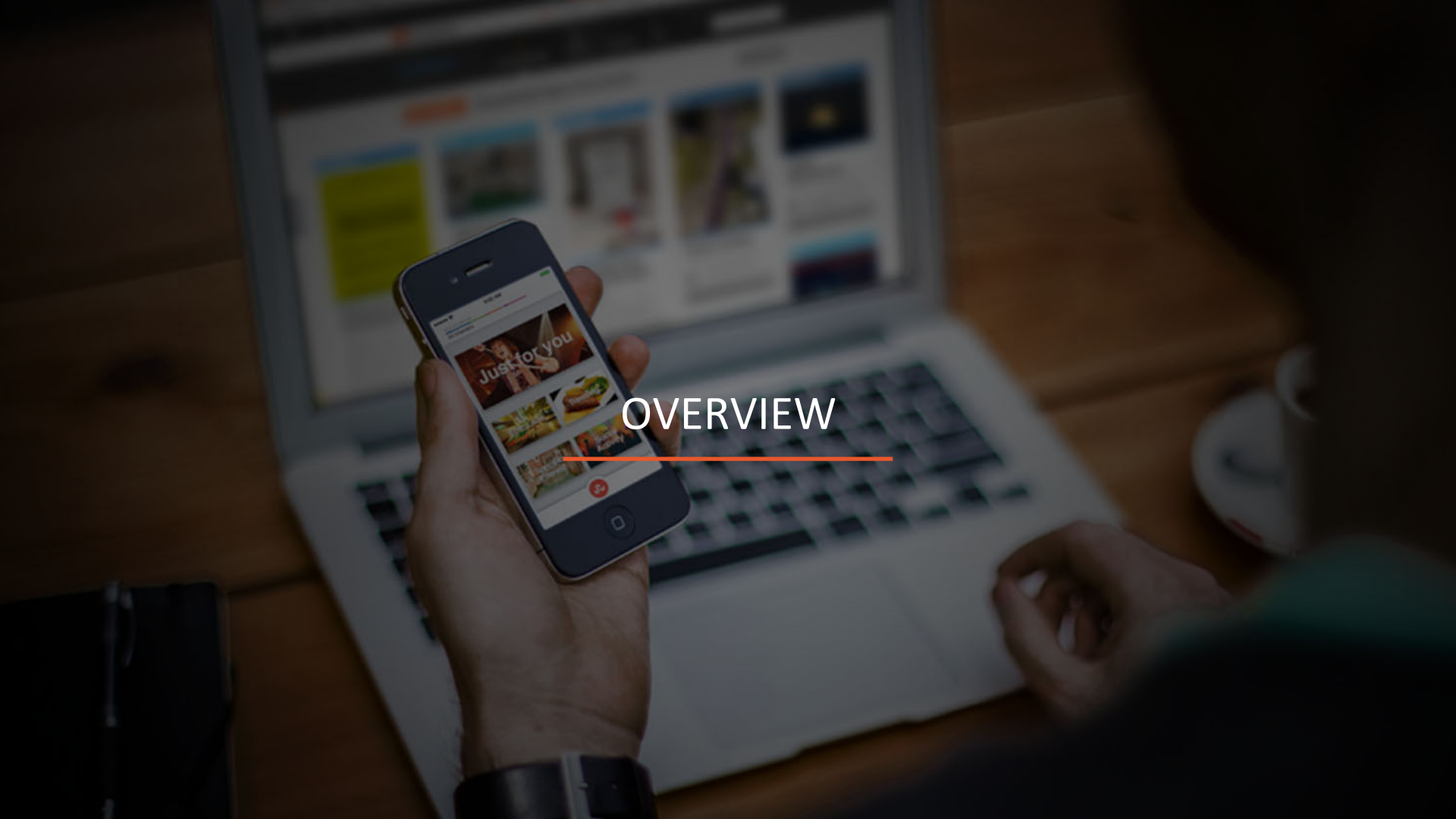
Near Real-time Webpage Recs “One at a time” Using Content Features

Presented at Text-By-The-Bay, SF

Ashok Venkatesan
Sr. Research Engineer, StumbleUpon

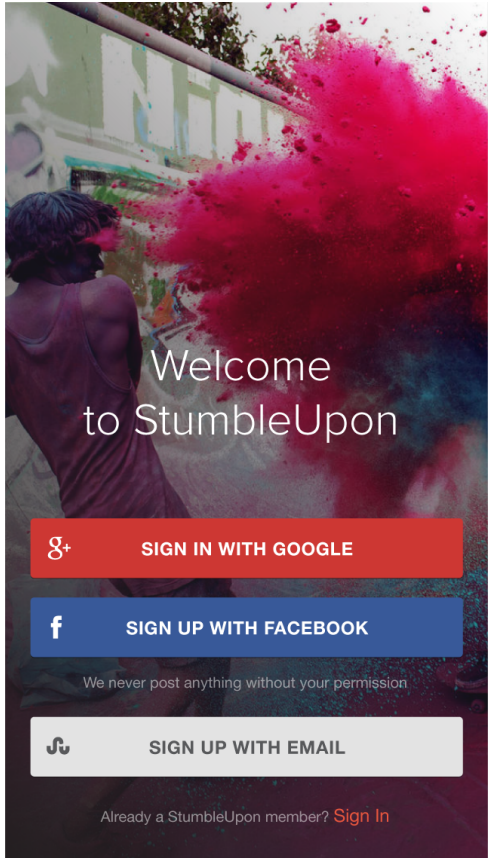
Agenda

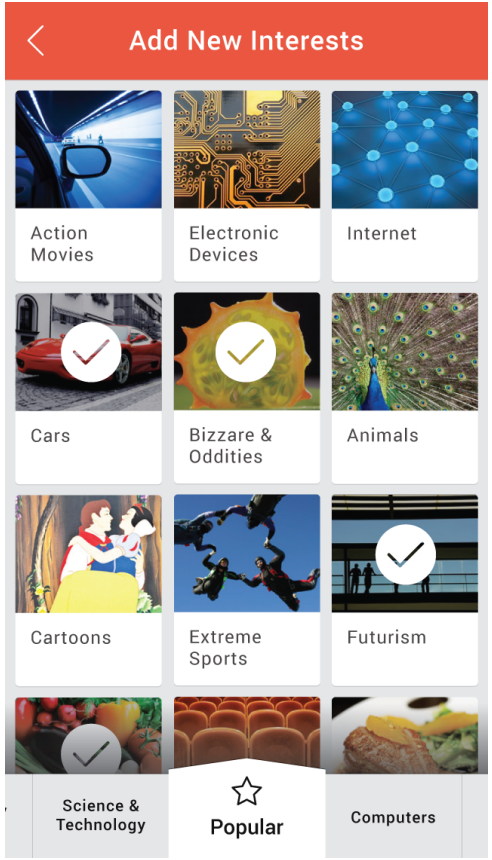
- Overview About us
- Recommendations (Recs) What, How and More
- Content Understanding Overview, Content Categorization
- Serendipitous Recs Methodology
- Future

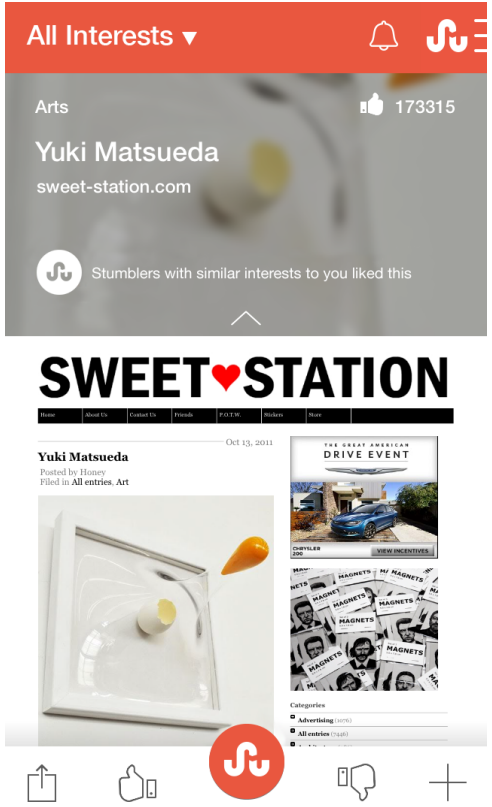


OVERVIEW

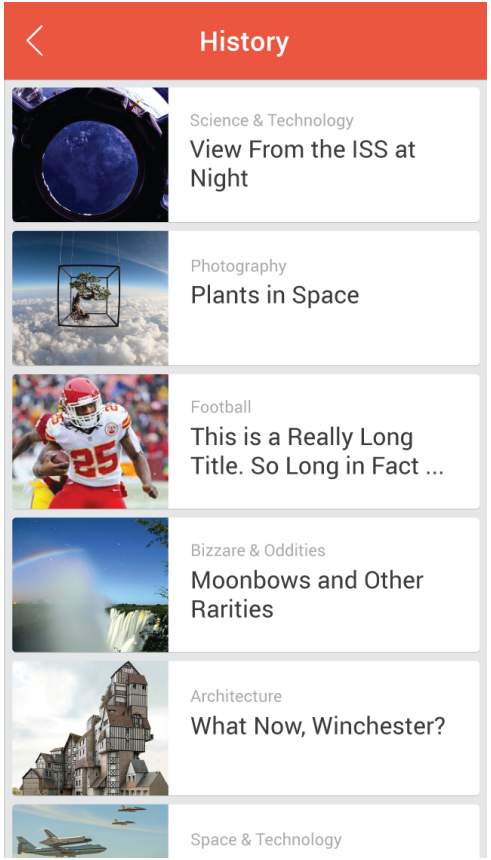
StumbleUpon – Choose Topics, Discover Content

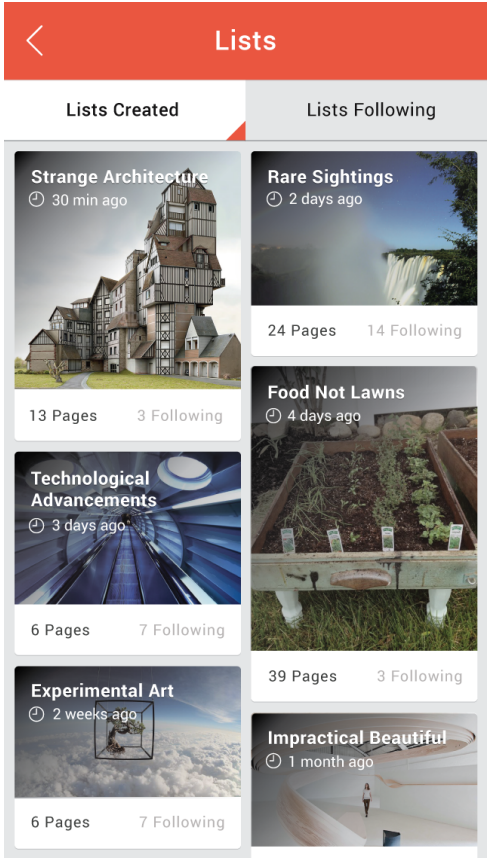


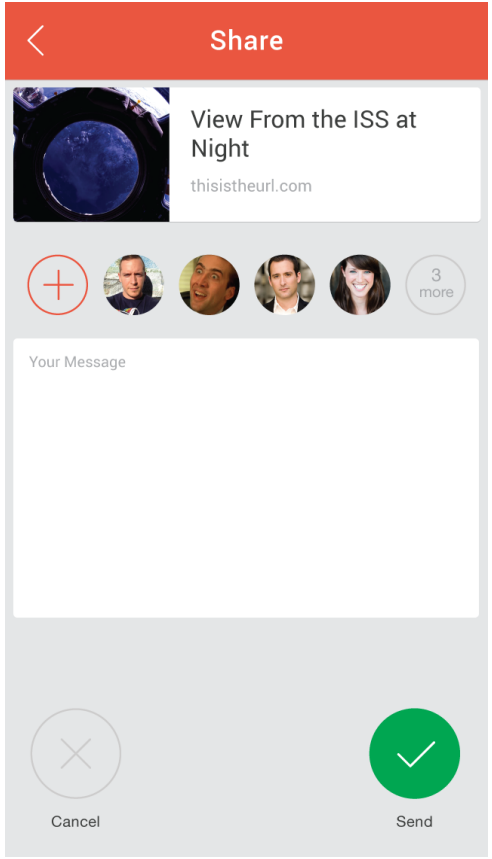


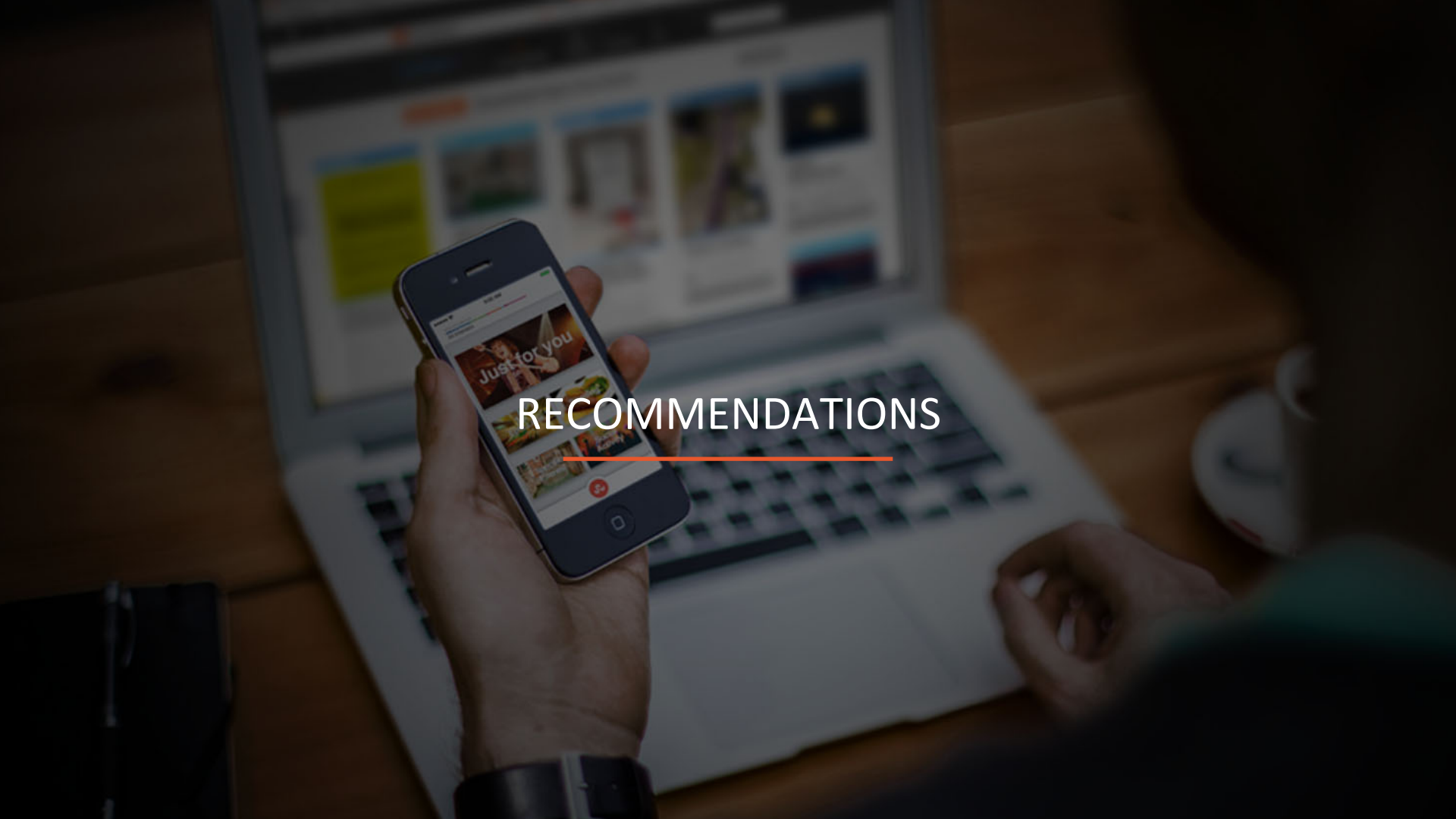


Bookmark, Organize and Share



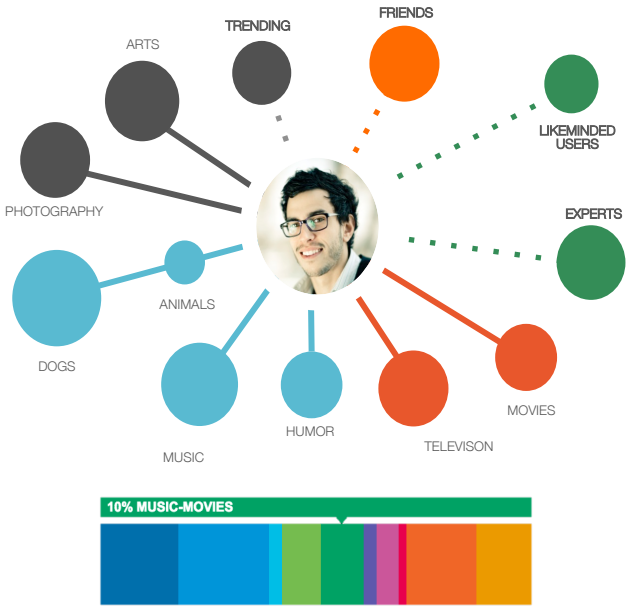


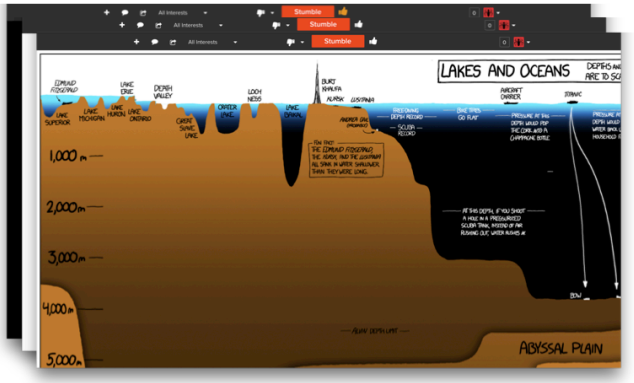




RECOMMENDATIONS

Recommendations – Matching User With Content



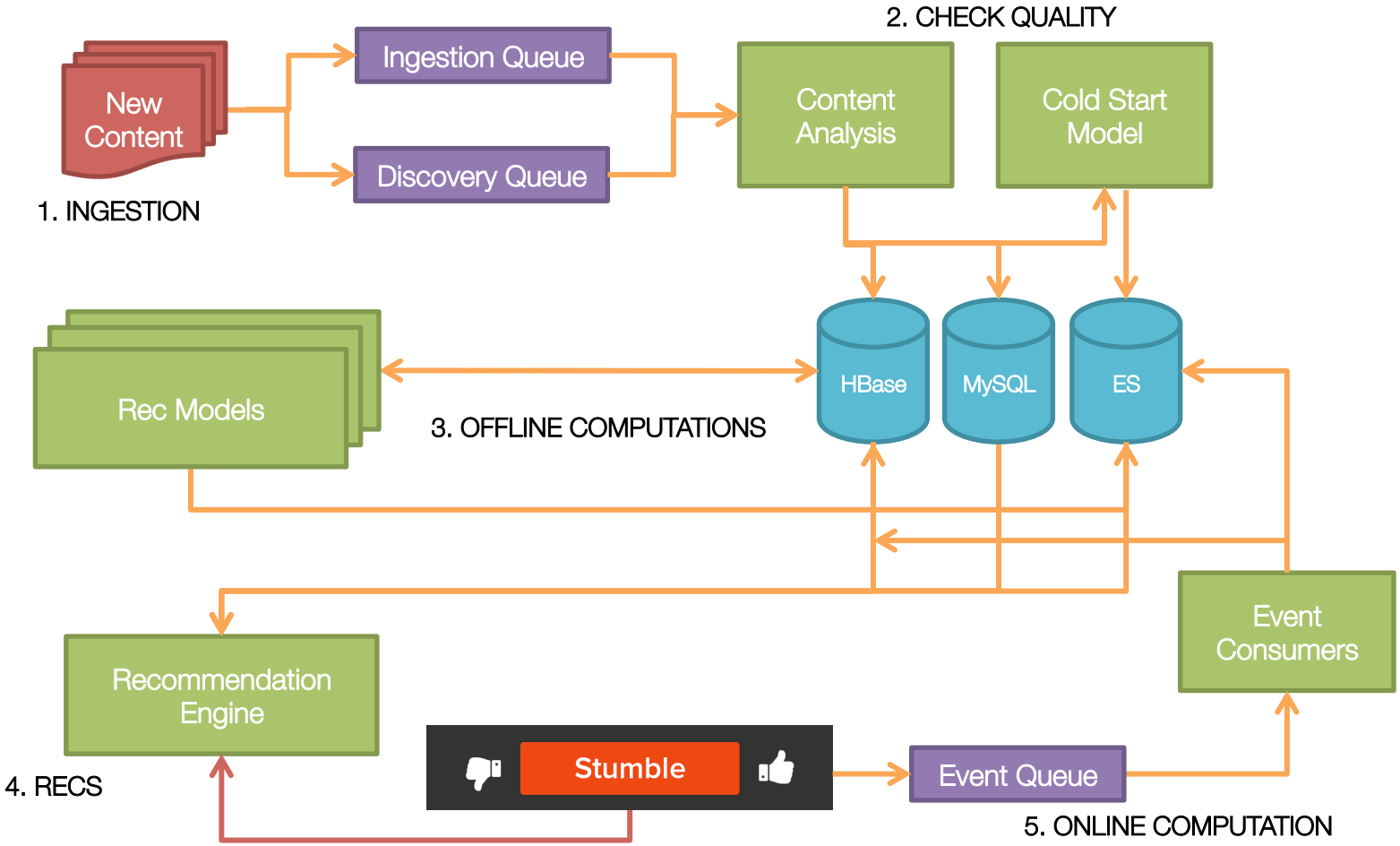


1. Understand User
2. Understand Content
3. Recommend
4. Get Feedback

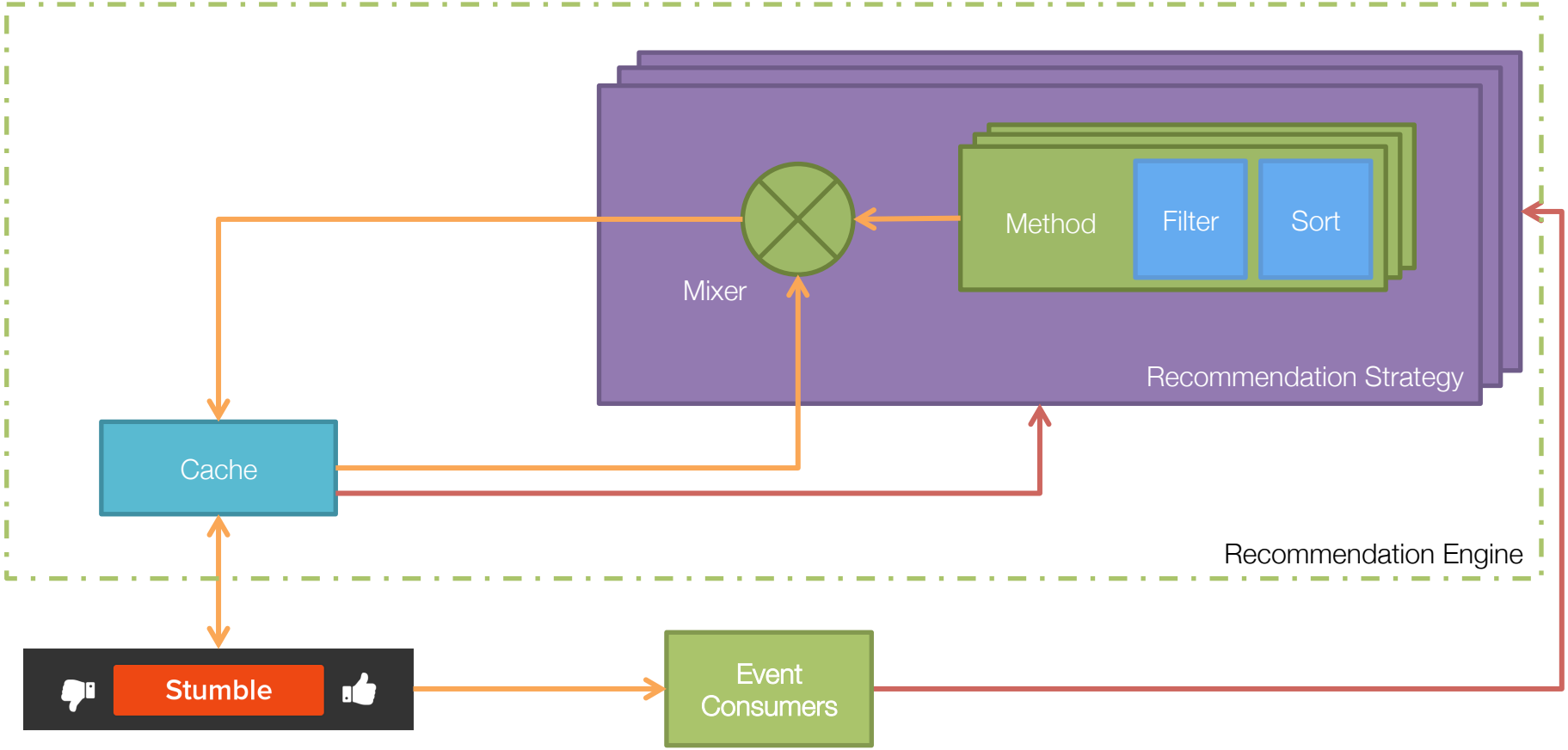
Problems and Challenges

- Understand User
  - User Quality
  - Latent Interests
- Understand Content
  - Content Quality
  - {Spam, Dupes} Detection
  - Spotting Dead + Parked Pages
  - Handling different content types
- Recommendations
  - Rec Quality
  - Managing item supply/demand + churn
  - Keep learning
- Business Related
  - One rec at a time
  - Don’t show already seen content
  - Constantly changing product/use-cases

Architecture I

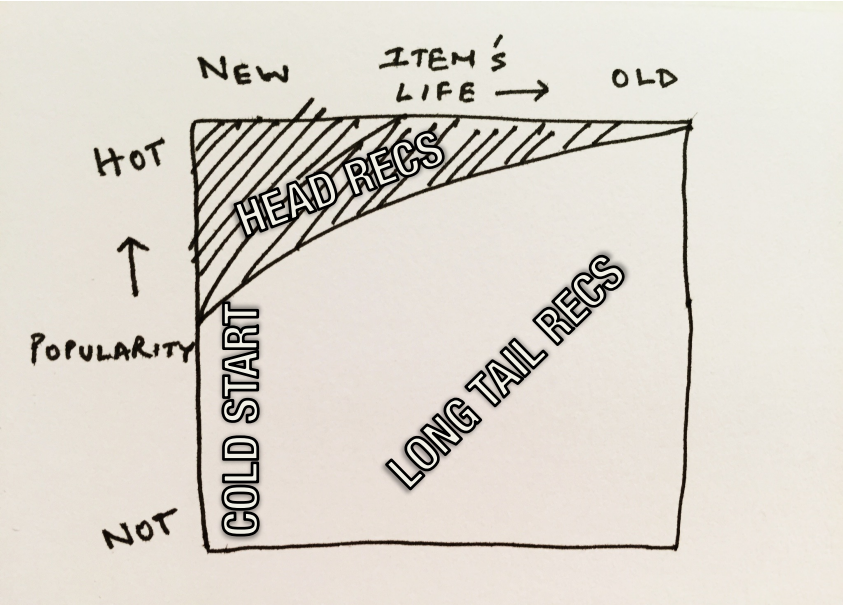


Architecture II



Item Lifecycle in a Recommender System

1. Ingestion Entry point for items; Feature extraction
2. Initial Recs
   - *Cold start* Optimize for maximum expected positive ratings and satisfy item demand
3. Head Recs
   - *Trending* Popular in the short run
   - *Timeless* Popular in the long run
4. Tail Recs
   - *Collaborative Filtering* Nearest neighbors based on user signals
   - *Serendipitous Recs* Unexpected but relevant

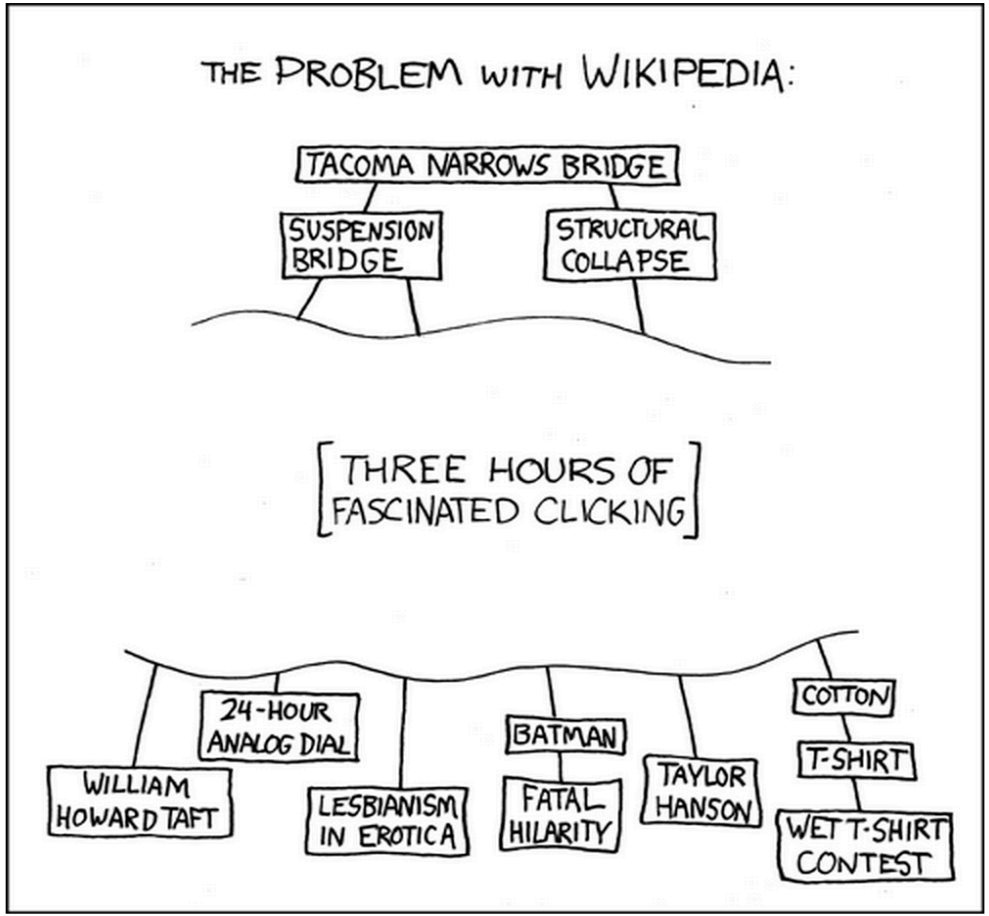


Serendipitous Recs

- What?
  - Recommend the “unexpected but useful”
  - “Go beyond relevance” and look for “interestingness”
- Why?
  - “Helps avoid tunnel vision”
  - Allows exploration, enables true discovery
- Challenges
  - Figuring out how to serve good content that is not random, is unexpected but useful…Hmm
  - Measuring/Controlling serendipity

Bordino I. et al., Penguins in sweaters, or serendipitous entity search on user-generated content, CIKM 2013

Serendipitous Surfing



CONTENT UNDERSTANDING

Characterizing Content

- Some relevant features
  - Topics
  - Keywords
  - Language
  - Mime Type
  - Content Type
  - Number of Ads
  - Number of Links
  - Responsive Design
  - Overlays/Popups and more…

Topic Models

- What
  - Discover underlying topic structure
  - Annotate documents
  - Index/Search documents

- Applications
  - Content Categorization
  - Dimensionality Reduction
  - User modeling

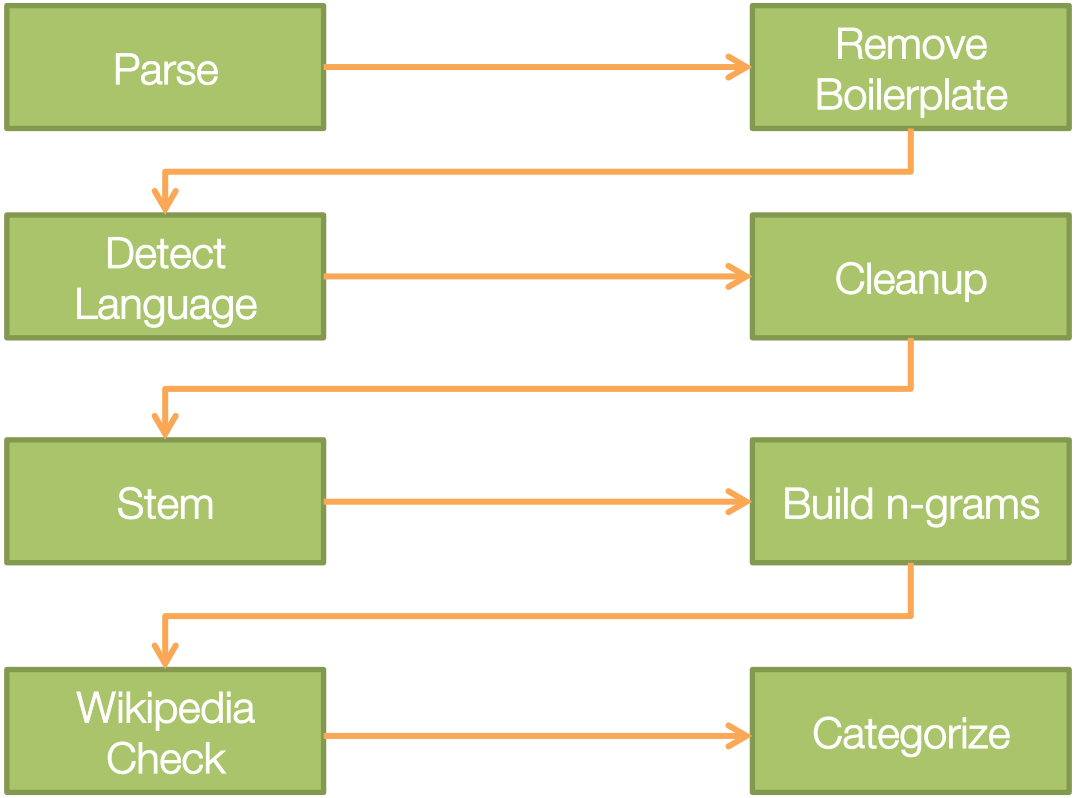
Blei, David M. "Probabilistic topic models." Communications of the ACM 55.4 (2012): 77-84.

Content Categorization

- Requirements
  - Categorize documents to a predefined topic taxonomy
  - Cluster documents into broader groups

- Applications
  - Search and recommendations
  - Evaluate user's categorization/re-categorization of documents
  - Discovering related interests

Feature Extraction

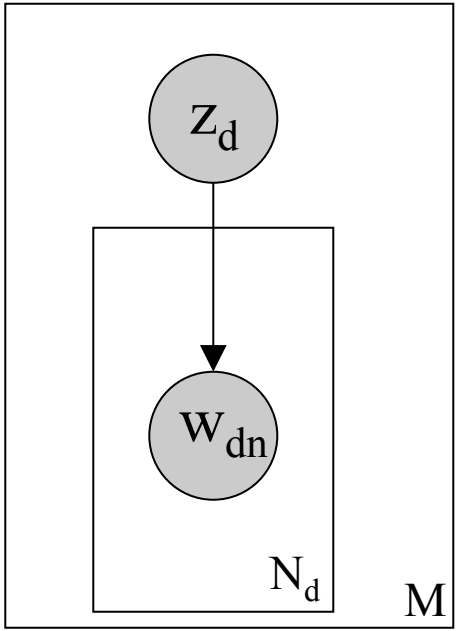


Milne, David, and Ian H. Witten. "An open-source toolkit for mining Wikipedia." Artificial Intelligence 194 (2013): 222-239.

Content Categorization: Discovering topics

- Naïve Bayes Classifier
  - Words are generated from one mixture
  - Words are independently distributed given topic

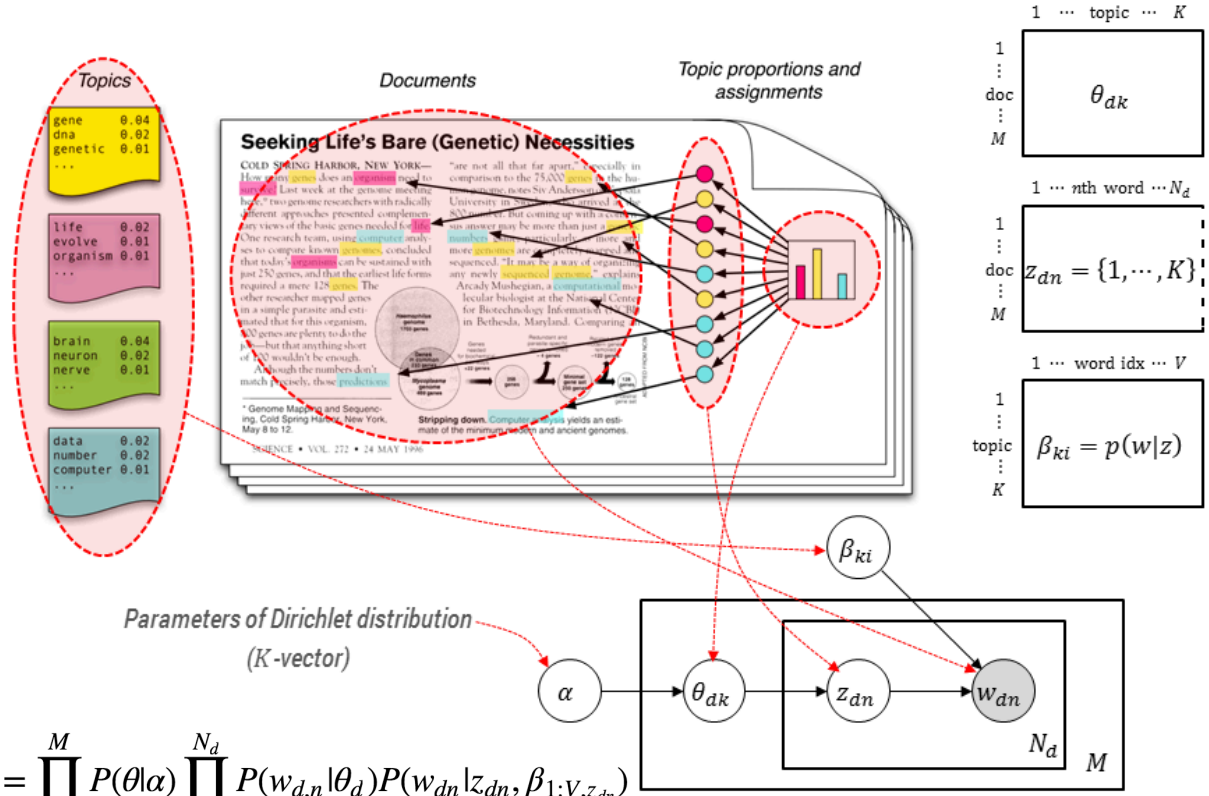
$$P(z_d|\mathbf{w}_d) = P(z_d)\prod_{n=1}^{N_d} P(w_{dn}|z_d)$$



Properties

- Supervised and generative (fully probabilistic)
- Efficient in both training and classification
- Easy to Implement an online version
- Dependent on a static vocabulary (retrain if vocabulary changes)
- Very restricted means of identifying semantic groups from a mixture model

Content Categorization: Discovering Semantic Groups



$$P(\mathbf{W}, \mathbf{Z}, \theta|\alpha, \eta) = \prod_{d=1}^{M} P(\theta|\alpha) \prod_{n=1}^{N_d} P(w_{d,n}|\theta_d) P(w_{dn}|z_{dn}, \beta_{1:V, z_{dn}})$$

Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." the Journal of machine Learning research 3 (2003): 993-1022.

image courtesy: http://parkcu.com/blog/latent-dirichlet-allocation/

Properties

- Unsupervised (Classic LDA) and generative
- Well suited for domain adaptation (taxonomy shift)
- Allows making topic clusters as loose/tight as needed
  - α controls the peak-ness of the document-topic distributions
  - η controls the peak-ness of the topic-word distributions
- Can be extended to discover relations, hierarchies, etc.,

Evaluation + Relearning

- Periodically evaluate the model
  - Perplexity
    - Measure of how surprised the model is on an average when having to guess between k equally probable choices.
    - The average log probability of the trained model having seen the test samples

$$2^{Entropy} = 2^{-\sum p \log p}$$

  - Use human judgment from word intrusion and topic intrusion tasks

- Good topic associations can be initialized from previous trainings or from separate topic clustering

Topic Mixtures

recipe
sugar
baking
cooking
oven
butter
food
vanilla
cream
black_pepper

photography
art
design
photograph
artist
camera
digital
architecture
illustration
painting

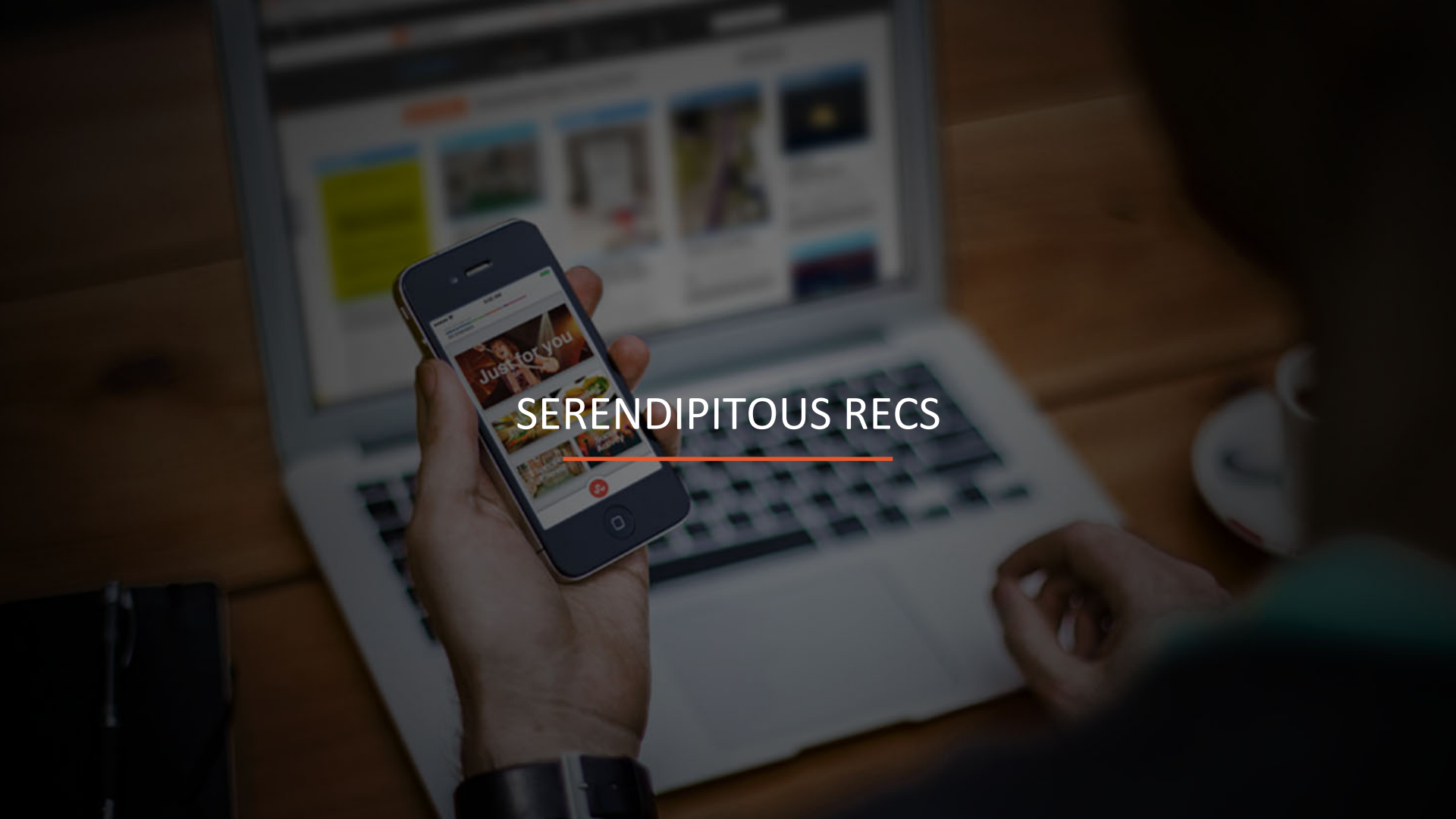
nasa
earth
moon
planet
sun
space
solar_system
universe
astronomer
science

israel
jews
egypt
bible
jesus
syria
iran
russia
new_testament
god

facebook
internet
twitter
advertising
youtube
marketing
myspace
google
social_media
mass_media

cannabis_(drug)
police
crime
law
prison
cannabis
drug
murder
gun
federal_bureau_of_investigation

energy
water
carbon_dioxide
sustainability
gas
nuclear_weapon
electricity
sun
climate_change
solar_energy

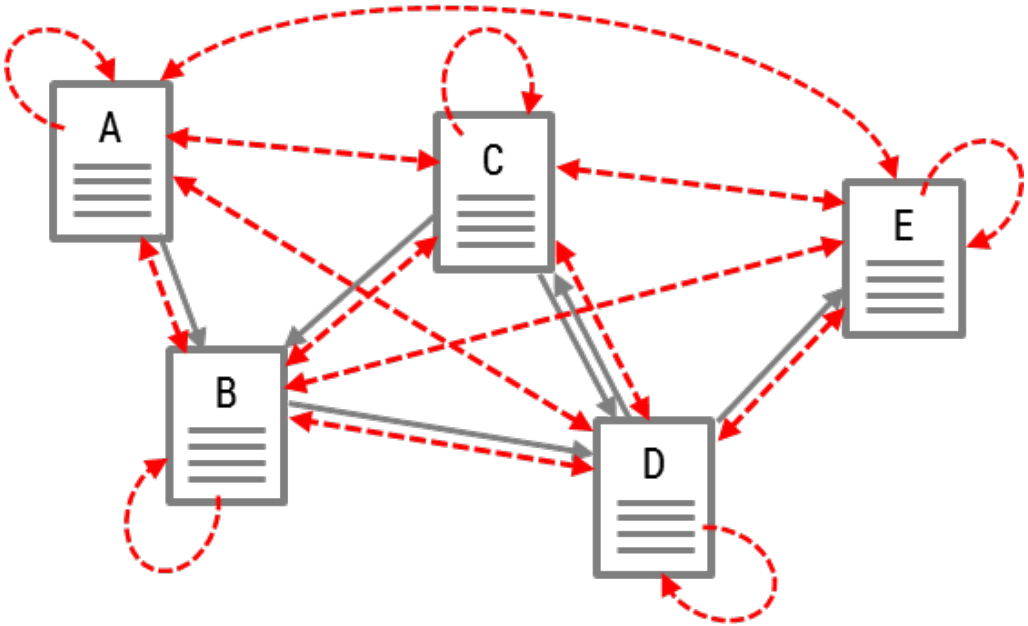


SERENDIPITOUS RECS

Methodology

- Dimensionality Reduction
  - Build LDA model using “Head” URLs
  - Use the model to classify “Tail” URLs in Latent Topic Space
- Document Graph
  - Compute pairwise similarity between documents with topic overlaps Cosine Similarity, Weighted Jaccard
  - Build a graph where documents make up the nodes and the similarity score make up the edge weights.
- Page Rank
  - Run topic sensitive page rank over the document graph.
  - Spot influential documents per topic and index for fast retrieval

Classic Pagerank



$$R_{i+1} = (1-\alpha)(S^T + \mathbf{w} \times \mathbf{d}^T)R_i + \alpha\mathbf{w}$$

Image courtesy: http://parkcu.com/blog/pagerank/

Page, Lawrence, et al. "The PageRank citation ranking: Bringing order to the web." (1999).

Topic Sensitive or Personalized Pagerank

$$\boldsymbol{R}_{i+1} = (1-\alpha)(S^T + w \times d^T)\boldsymbol{R}_i + \alpha \begin{bmatrix} \frac{1}{N} \\ \vdots \\ \frac{1}{N} \end{bmatrix}$$

$$\boldsymbol{R}_{i+1} = (1-\alpha)(S^T + w \times d^T)\boldsymbol{R}_i + \alpha \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

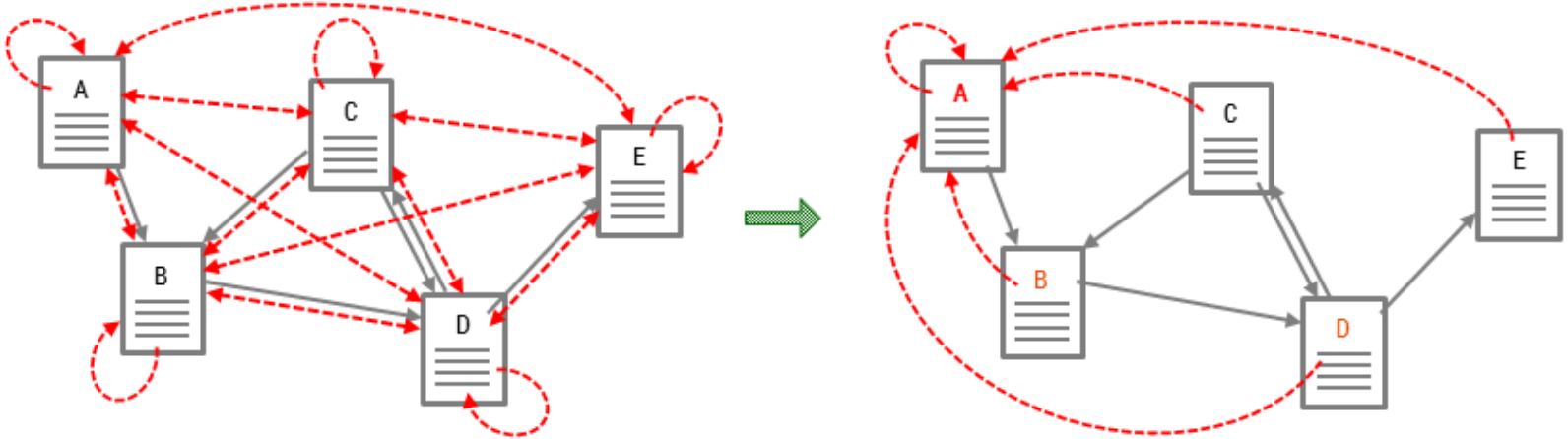


Image courtesy: http://parkcu.com/blog/pagerank/

Haveliwala, Taher H. "Topic-sensitive pagerank." Proceedings of the 11th international conference on World Wide Web. ACM, 2002.

Evaluation

- A/B Testing
  - Measure the difference in user behavior (implicit/explicit signals and retention):
    - “A Recommended item” vs. “Randomly picked item from the set”
    - “Serendipity free stumbling session” vs. “Sessions with serendipitous recommendations”

Ongoing Work

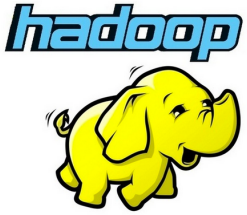
- More online computations
- Model/Feature Improvements
  - E.g. authorship, published date, device optimizations etc.,
- Dupe detection improvements
- Better Recs! ☺

Stack

APACHE
HTTP SERVER

elasticsearch.

APACHE
HBASE



THANKS