

When Advanced Machine Learning Meets Intelligent Recommender Systems

Presenter: Liang Hu and Songlei Jian, UTS

Authors: Liang Hu, Songlei Jian, Shoujin Wang, UTS

Prof. Longbing Cao, UTS, Prof. Jian Cao, SJTU

Goal

Providing a comprehensive understanding of how to apply state-of-the-art machine learning approaches to build more sensible recommender systems with various heterogeneous data and complex relations.

Agenda

- Preliminaries for recommender systems and machine learning (45 mins)
- 3C information for recommender systems (2 hrs and 15 mins)
 - Complementary information (50 mins)
 - Comprehensive information (30 mins)
 - Contextual information (55 mins)
- Recommender systems in real world (30 mins)

Section I: (Presented by Liang Hu)

- Backgrounds
- Traditional Recommender Systems and Challenges
- Non-IIDness in Recommender Systems
- Representation: The Foundation of Machine Learning and Recommender Systems

Section II: (Presented by Songlei Jian)

- Leveraging Complementary Information
 - Cross-domain Recommender Systems
 - Social Recommender Systems
- Leveraging Comprehensive Information
 - Multimodal Recommender Systems
 - Multi-objective Recommender Systems

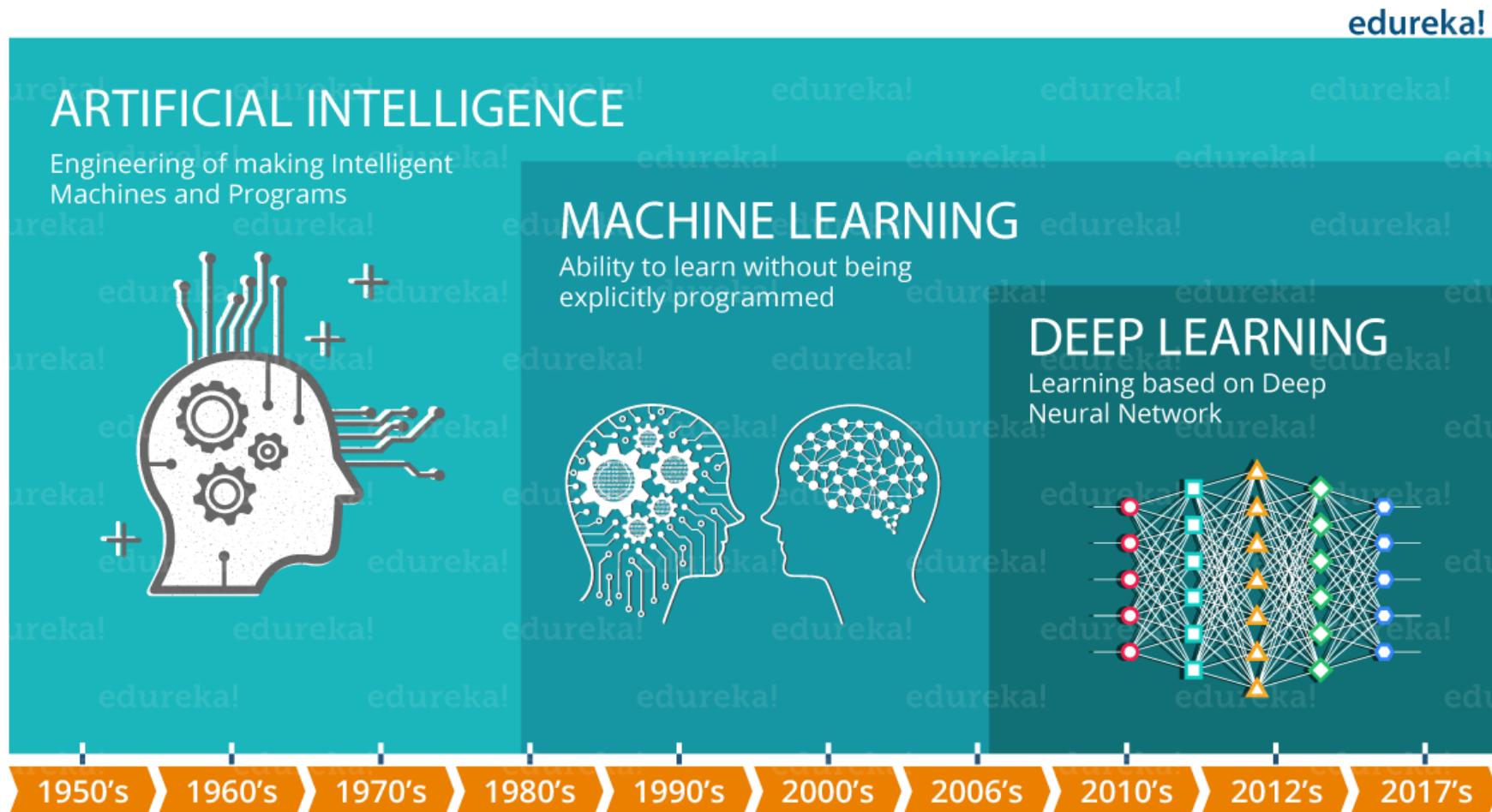
Section III: (Presented by Liang Hu)

- Leveraging Contextual Information
 - Context Modeling in Recommender Systems
 - Session-based Recommender Systems
 - Group-based Recommender Systems
- Real Recommender Systems in Practice

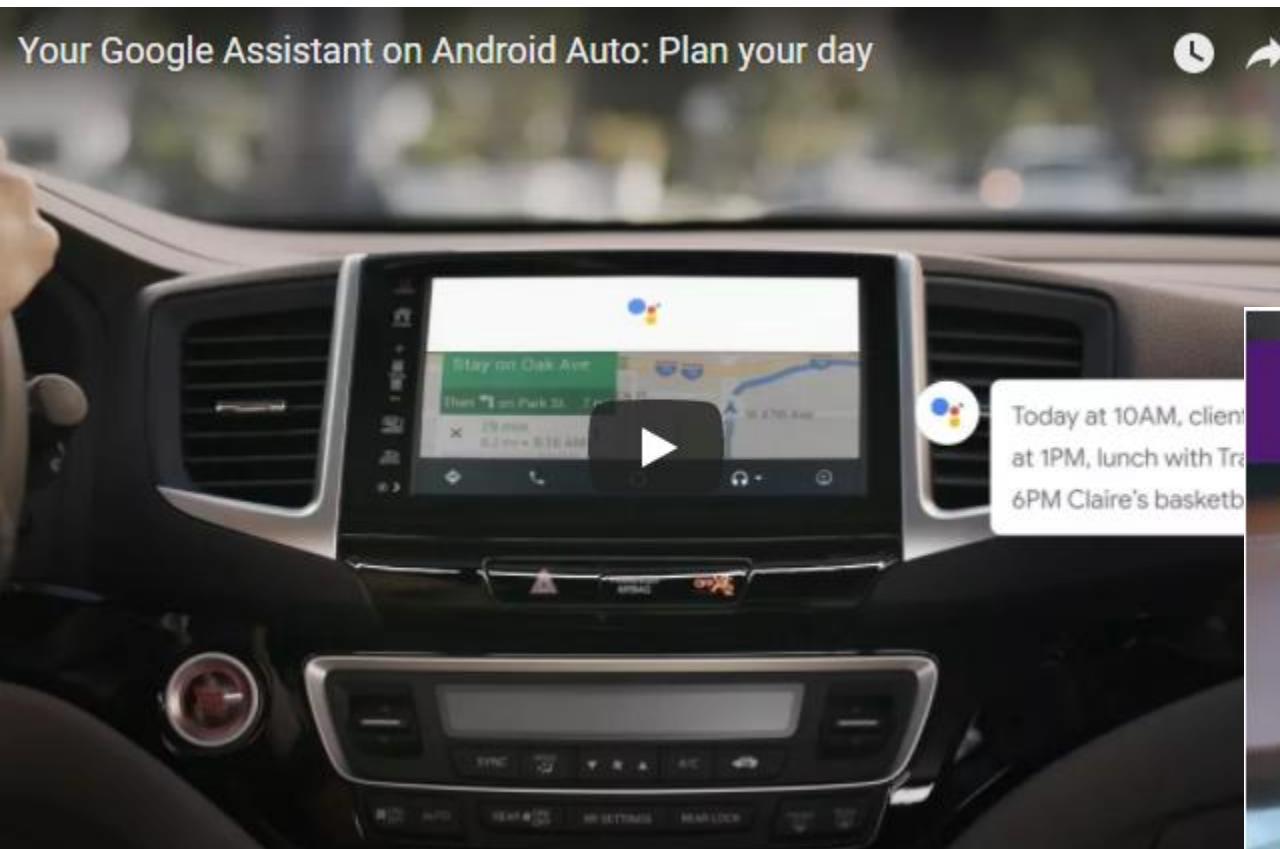
Section I:

- **Backgrounds**
 - Artificial Intelligence in Daily Life
 - Recommender Systems in Daily Life
 - AI + Big Data Era: The Marriage of Machine Learning and Recommender Systems
- **Recommender Systems and Challenges**
- **Non-IIDness in Recommender Systems**
- **Representation: The Foundation of Machine Learning and Recommender Systems**

AI, Machine Learning and Deep Learning



AI in Every Corner



Section I:

- **Backgrounds**
 - Artificial Intelligence in Daily Life
 - **Recommender Systems in Daily Life**
 - AI + Big Data Era: The Marriage of Machine Learning and Recommender Systems
- **Recommender Systems and Challenges**
- **Non-IIDness in Recommender Systems**
- **Representation: The Foundation of Machine Learning and Recommender Systems**

Information Age to Recommendation Age

Anderson, C. (2006). *The long tail: Why the future of business is selling less of more*

- With the information explosion, quickly finding the desirable information has become more important.
 - Modern recommender systems are the evolution of intelligent systems and information retrieval systems.



Recommendation in commercial

NEW & INTERESTING FINDS ON AMAZON **EXPLORE**

All

amazon My Prime

Departments ▾ Your Amazon.com Today's Deals Gift Cards & Registry Sell Help

EN Hello, Sign in Account & Lists Orders Try Prime Cart 0

Today's Deals **SAVE UP TO 15% WITH SUBSCRIBE & SAVE** Shop now Cascade ©2017 P&G Ad feedback

Never miss another deal ▾

Showing 1-48 of 8840 results Sort by Relevance

Department	Product Image	Deal Type	Price	Rating	Availability
<input type="checkbox"/> Amazon Devices <input type="checkbox"/> Amazon Home Services <input type="checkbox"/> Amazon Video <input type="checkbox"/> Arts, Crafts & Sewing <input type="checkbox"/> Automotive & Motorcycle <input type="checkbox"/> Baby <input type="checkbox"/> Baby Clothing & Accessories <input type="checkbox"/> Beauty <input type="checkbox"/> Books <input type="checkbox"/> Boys' Fashion <input type="checkbox"/> Camera & Photo <input type="checkbox"/> Cell Phones & Accessories See more		DEAL OF THE DAY	\$269.99 Price: \$344.00 (22% off)	★★★★★ 20	Add to Cart
Deal Type		DEAL OF THE DAY	\$17.90 - \$74.99 Ends in 16:11:33 Save 25% on T-fal Kitchen Products	★★★★★ 828	See details
Deal of the Day Lightning Deals Savings & Sales Coupons Prime Early Access Deals		DEAL OF THE DAY	\$103.99 Price: \$148.42 (30% off) Ends in 16:11:33 Save Big on Blossom Smart Watering Controller	★★★★★ 27	Add to Cart
Availability		DEAL OF THE DAY	\$95.99 - \$129.99 Ends in 16:11:34 Save on TeChef 7 and 8-Piece Cookware Sets	★★★★★ 165	See details
		DEAL OF THE DAY	\$99.99 Ends in 16:11:33 40% off Gerber Center-Drive Multi-tool, Pen & Dual Sheath bundle	★★★★★ 153	Add to Cart
			\$5.00 New Imagine Dragons Album: \$5 Ships from and sold by Amazon Digital Services LLC.	★★★★★ 53	See details

Recommender Systems have occupied our life

What to eat

Food & Drink

Sort by **Relevance**

Up to 67% off Room Hire with Drinks for Four
Dynasty Karaoke
Up to ten party-goers can add more zing to their night out with a private room hire featuring a glass of house wine, beer or soft drink each
Haymarket • 2.2 km
★★★★★ (52)

\$72 From \$24 View Deal

Up to 60% off Japanese BBQ Special for Two
Taisno Wagyu Japanese BBQ
Zetland • 5 km
90+ bought \$98 From \$58 ★★★★★ (2,181) \$154 From \$88

Up to 57% off Darts + Beer or Mix Drinks for 2
The Century Bar
Sydney • 1.9 km
★★★★★ (7) \$44 From \$19 ★★★★★ (29) \$299.50 From \$149

View On Map

Categories

- All Deals
- Food & Drink (66)
- Restaurants (318)
- Cafes & Treats (46)
- Bars (8)
- Breweries, Wineries & Distilleries (8)
- Groceries & Markets (3)

Which to dress

Men's Compression Shorts
20+ bought From \$15

Skechers Shoes for Women and Men
70+ bought From \$59

Customised Build-A-Brick Cap
30+ bought From \$19

Women's Knitted Ugg Boots
90+ bought \$219 From \$65

Ralph Lauren Polo Shirt
250+ bought \$120 From \$59

Two-Piece Men's Thermal Wear Set
20+ bought From \$25

Where to go

Sort by **Relevance**

UTS Building 25 AAI, 12/2 Blackfriars St
Burwood, New South Wales 2134
Add destination

Route options

Avoid

- Highways
- Tolls
- Ferries

Distance units

- Automatic
- miles
- km

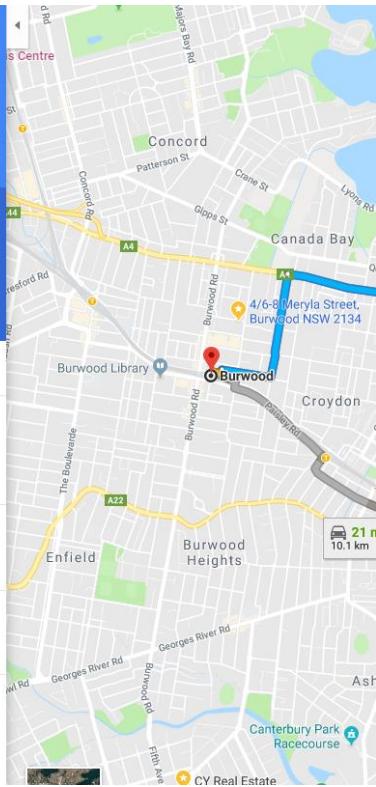
CLOSE

Send directions to your phone

via Great Western Hwy/Parramatta Rd 18 min
Fastest route, the usual traffic 10.5 km
DETAILS

via Great Western Hwy/Parramatta Rd/A22 21 min
10.1 km

via A4 21 min
11.4 km



Section I:

- **Backgrounds**
 - Artificial Intelligence in Daily Life
 - Recommender Systems in Daily Life
 - **AI + Big Data Era: The Marriage of Machine Learning and Recommender Systems**
- **Recommender Systems and Challenges**
- **Non-IIDness in Recommender Systems**
- **Representation: The Foundation of Machine Learning and Recommender Systems**

Machine Learning Methods Dominate RS Competitions

Alibaba Competitions

The screenshot shows a table of competitions from Alibaba. The columns include Name, Area, Attribute Types, Evaluation, Year, Provider, and a 'Full Data' link. The competitions listed are:

Name	Area	Attribute Types	Evaluation	Year	Provider	Full Data
IDCAI-15 Competition	E-commerce	Mixed	Without	2015	Tmall	Apply
REC-TMALL	E-commerce	Mixed	Without	2015	Tmall	Apply
AI_Mum_Baby	E-commerce	Mixed	Without	2015	Taobao & Tmall	Apply
AI_Mobile_Rec	E-commerce	Mixed	With	2015	Alibaba	Apply
IDCAI-15 Dataset	E-commerce	Mixed	With	2015	Tmall	Apply
Purchase_Redemption_	Finance	Mixed	With	2015	Ant Financial Services	Apply
Intelligent_Transpor	Transportation	Categorical	Without	2015	Guizhou Province	Apply

RecSys 2018 – Challenge – RecSys

<https://recsys.acm.org/recsys18/challenge/> ▾ 翻译此页

RecSys Challenge 2018. The RecSys Challenge 2018 will be organized by Spotify, The University of Massachusetts, Amherst, and Johannes Kepler University, Linz. Spotify is an online music streaming service with over 140 million active users and over 30 million tracks. One of its popular features is the ability to create ...

RecSys 2017 – Challenge – RecSys

<https://recsys.acm.org/recsys17/challenge/> ▾ 翻译此页

RecSys Challenge 2017. The RecSys Challenge 2017 is organized by XING, Politecnico Milano and Free University of Bozen-Bolzano. XING is a social network for business. People use XING, for example, to find a job and recruiters use XING to find the right candidate for a job. At the moment, XING has more than 18 ...

The screenshot shows a list of recommended competitions on Kaggle. The competitions listed are:

- hotel recommend kernel - Expedia Hotel Recommendations
- personalized recommending system for techcrunch kernel - TechCrunch Posts Compilation
- Recommending Chess Openings kernel - Chess Game Dataset (Lichess)
- Recommend Movie with Clustering kernel
- spark recommend product kernel - Santander Product Recommendation
- Non-Negative Recommended System kernel - Steam Video Games
- Music Recommend --LGBM kernel - WSDM - KKBox's Music Recommendation Challenge
- A small, personalized recommending system 59de... kernel - TechCrunch Posts Compilation
- spark recommend hotels kernel - Expedia Hotel Recommendations
- Any books, tutorials recommended? topic - Data Science Bowl 2017

Tip: narrow your results by adding in: followed by the content type or tag: in:kernels in:datasets in:topics in:comments in:users in:jobs tag: 'artificial intelligence'

New to Kaggle? Get started with our most popular challenges for beginners!

Competitions machine learning problems

Make a Submission Upload your prediction file for real-time scoring & a spot on the leaderboard.

RecSys 2016 – Challenge – RecSys

<https://recsys.acm.org/recsys16/challenge/> ▾ 翻译此页

In this year's edition of the RecSys Challenge, the task is: given a XING user, predict those job postings that a user will click on. Submitted solutions will be evaluated offline and online. A detailed description of the challenge can be found on the website of the RecSys Challenge 2016. Accepted contributions will be ...

Big Data Era: Tell Truths from Data

Recommendation Era: Recommend Truths from Data

Data with Machine Learning Methods

- Attributes
 - Regression, Clustering, Factor Analysis

- Labels
 - Classification, Learning to Rank

- Images, Videos
 - Computer Vision

- Text
 - Natural Language Processing (NLP), Sentiment analysis

- Sequence
 - Time series forecasting

- Network
 - Link Prediction, Network Embedding

Data with Recommender Systems

- User/Item features

- Ratings

- Item pictures, Multimedia Data

- Reviews

- Transaction

- User/Item Network

Conclusion

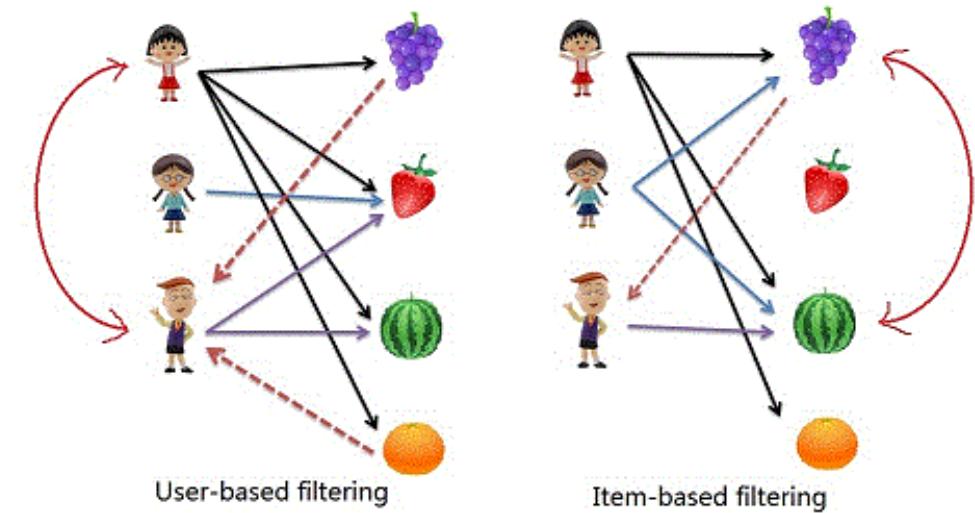
- Data act as the matchmaker to bring advanced machine learning methods to recommender systems

Section I:

- **Backgrounds**
 - Artificial Intelligence in Daily Life
 - Recommender Systems in Daily Life
 - AI + Big Data Era: The Marriage of Machine Learning and Recommender Systems
- **Recommender Systems and Challenges**
- Non-IIDness in Recommender Systems
- Representation: The Foundation of Machine Learning and Recommender Systems

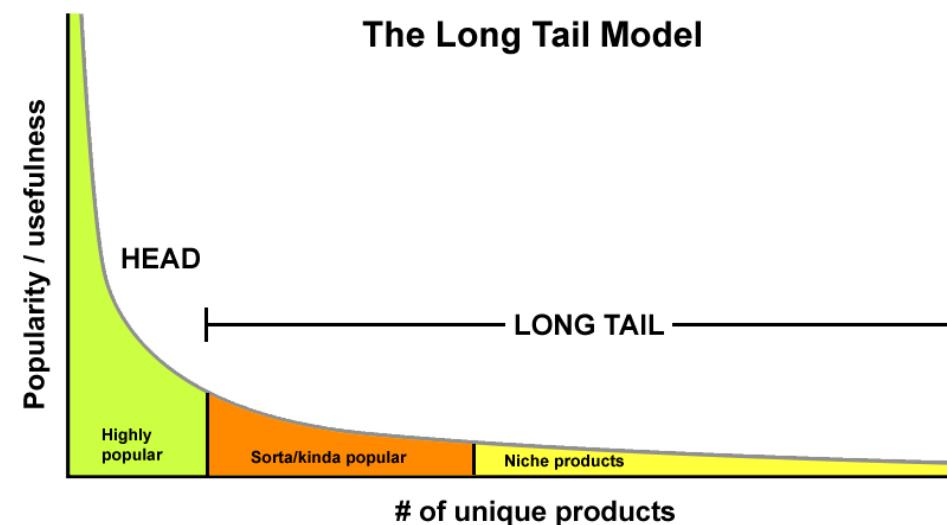
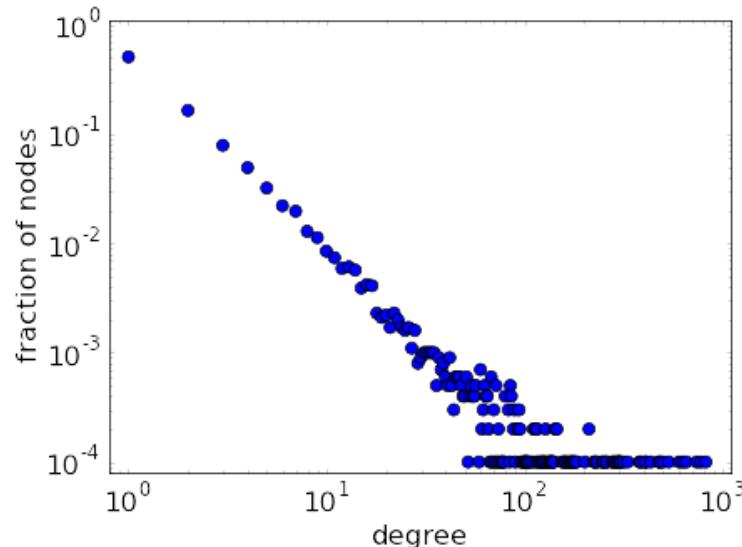
Collaborative Filtering (CF)

- Intuition (user-based filtering): If user **A** related to user **B** and **A** bought x and y, then **B** who bought x will buy y.
- Famous examples: Amazon.com's recommender system
- Facebook, MySpace, LinkedIn use collaborative filtering to recommend new friends, groups, and other social connections.



Data Characteristics in Recommender Systems

- Power law or Long tail distribution
 - Data associated with the **majority** of users are **insufficient** and even **absent** in real world.
 - In most recommender systems, the **majority** of users/items only associated with very **few data** while only **minority** of users/items have **sufficient data**

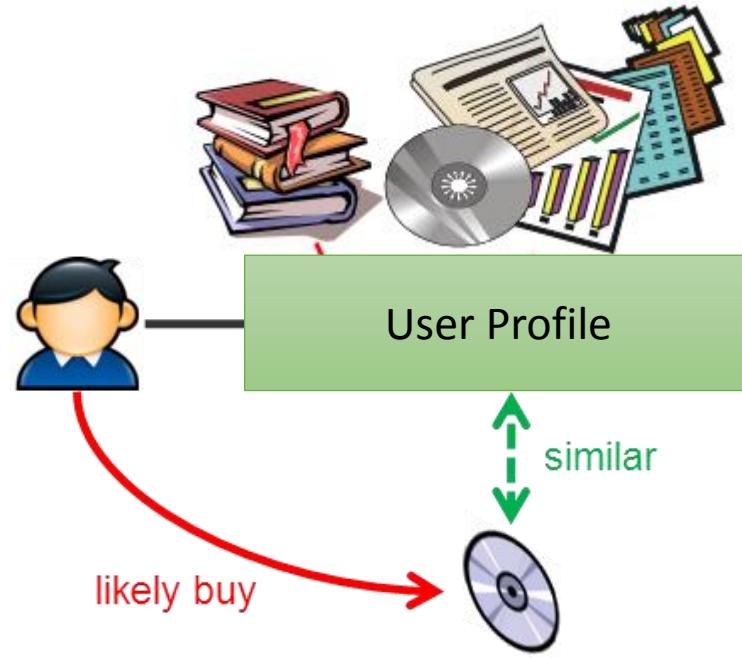


Challenges in Collaborative Filtering

- Data Sparsity
 - In real-world recommender systems, the data generally follow the power-law distribution.
- Cold Start
 - When new users or new items are added, the system cannot recommend to these users and these items due to the lack of data.
- Scalability
 - There are millions of users and products in real systems. Thus, a large amount of computation power is often necessary to calculate recommendations.

Content-based Filtering (CBF)

- CBF is based on the features of items
 - Attributes of items
 - Description of items
 - Text of article
- User profile is built with the features of historical items
- Recommend items through the similarity with user profile



Challenges in Content-based Filtering

- Limited Content Analysis
 - Limited content analysis occurs when the system has a limited amount of information on its users or the content of its items.
- Over-specialization
 - The system can only recommend items that highly similar with user's profile, the user is limited to being recommended items similar to those already rated

Main Cause of These Challenges

- Data Sparsity
- Cold Start
- Limited Content Analysis
- Over-sepcialization

Insufficiency of Data

Modern Recommender Systems

- Built on More Complex Data
 - Multiple data types
 - Ratings
 - Images
 - Text
 - Multisource
 - Multiple domains
 - Multiple systems
 - Social data
 - Acquire data from user social media
 - Multiple criteria
 - Multi-objectives: accuracy, novelty
 - Multi-metric:



Challenges in Modern Recommender Systems

- Heterogeneity
 - Items may belong to **different domains and systems**, so they have heterogeneous features.
 - User behavior also has heterogeneity due to the differences of **background, characteristic, and social relations**.
- Multi-information Integration
 - How to effectively link multiple data sources ?
 - How to deal with multiple types of data ?
 - How to integrate multiple domains and multiple systems ?

Challenges in Modern Recommender Systems

- Social Influence
 - People are often influenced by various social factors when making decisions.
 - Social information is helpful in improving the recommendation results and relieve the data sparsity and cold-start problem.
- Vulnerability
 - As the open environment of the Internet, recommender systems have to confront with false information and malicious attacks.
 - **Shilling attack** refers to a group of spam users intentionally providing fake feedback, e.g., **much higher or lower** ratings than a true rating to bias the ratings and the recommendations for them.

Section I:

- Backgrounds
- Recommender Systems and Challenges
- **Non-IIDness in Recommender Systems**
- Representation: The Foundation of Machine Learning and Recommender Systems

Big data challenges existing theories and systems

The screenshot shows a news article from The Guardian. At the top, there's a red circle highlighting the breadcrumb navigation: "News > World news > Greece". Below it, the main headline reads: "Violence continues in Greece as rioters firebomb buildings". A sub-headline provides context: "Protesters in Athens torch offices and cars amid clashes with police after memorial for teenager". The author is listed as "And Dawar" with the publication date "Sunday 21 December 2008 17.05 GMT". There's also a link to "Article history". On the right side of the article, there's a sidebar with a blue background containing the text: "What Ad would you place here?". Below the main headline, there's a photograph showing a youth in riot gear attacking a police officer. A caption under the photo states: "A youth assaults a police officer in Athens, during a week of riots after the shooting of a teenager. Photograph: Bela Szandelszky/REUTERS".

Irrelevant and
Damaging to Brand

What's wrong of the recommendation?

- There may be many reasons,
 - Content understanding
 - Understand the semantic hidden in contents
 - Analyze the relevance between news and ads from every possible aspect
 - Treat each piece of news differently
 - ...
- Modern recommender systems are build on non-iid assumption
 - Relationships between objects, syntactically, semantically,
 - Heterogeneity between objects, sources, ...

Non-IIDness in Complex Data

- Heterogeneity:
 - Data types, attributes, sources, aspects, ...
 - Formats, structures, distributions, relations, ...
 - Learning outcomes
 - Coupling relationships:
 - Within and between values, attributes, objects, sources, aspects, ...
 - Structures, distributions, relations, ...
 - Methods, models, ...
 - Outcomes, impact, ...
-
- The diagram consists of four main elements: 1) A large green bracket on the right side of the slide, spanning from the 'Not identically distributed.' box down to the 'Not independent.' box. 2) A green callout box containing the text 'Not identically distributed.' in red, with a thin green arrow pointing from the left towards it. 3) A green callout box containing the text 'Not independent.' in red, with a thin green arrow pointing from the bottom towards it. 4) A green rectangular box on the far right labeled 'Non-IIDness' in red text, with a vertical line connecting its top edge to the top of the large green bracket.

Non-IIDness in Recommender Systems

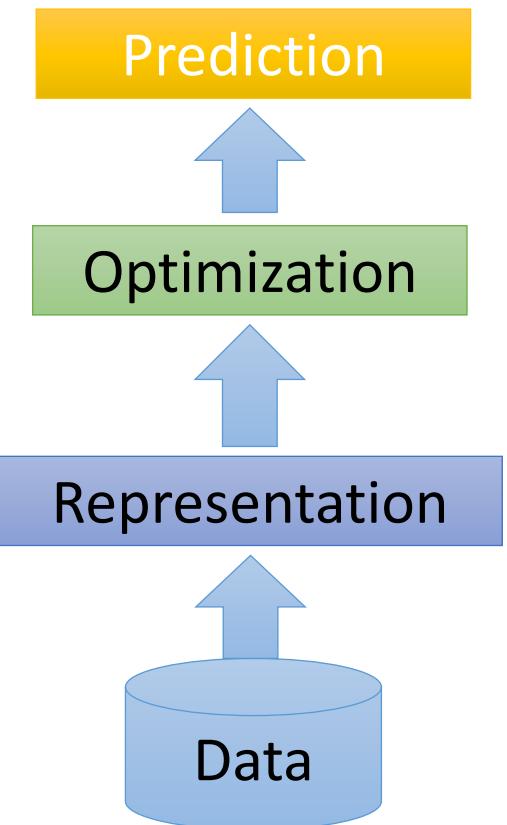
- Heterogeneity in various aspects:
 - The heterogeneity over users (**social RS, group-based RS**)
 - The heterogeneity over items (**cross-domain RS, multi-modal RS**)
 - The heterogeneity of data types (**multi-modal RS**)
 - The heterogeneity of domains (**cross-domain RS**)
- Coupling in various aspects:
 - The coupling between users (**social RS, group-based RS**)
 - The coupling between items (**session-based RS, cross-domain RS**)
 - The coupling between different data types (**multi-modal RS**)
 - The coupling between different domains (**cross-domain RS**)

Section I:

- Backgrounds
- Recommender Systems and Challenges
- Non-IIDness in Recommender Systems
- **Representation: The Foundation of Machine Learning and Recommender Systems**

Machine Learning

- Machine learning concerns the construction and study of systems that can learn from data.
- Machine learning focuses on prediction, based on *known* properties learned from training data
- The *core* of machine learning deals with *representation* and generalization.
- Good features are essential for successful ML: **90%** of effort



Feature Engineering VS Feature Learning

- **Feature engineering**
 - Human ingenuity and prior knowledge
 - Domain-specific hand-crafted features
 - GIST, SIFT features in computer vision
 - MFCC features in speech perception
- **Feature learning (representation learning)**
 - Learning from data
 - Dealing heterogeneity in attributes automatically
 - Without human prior
 - CNN features in computer vision

Typical data in recommender systems

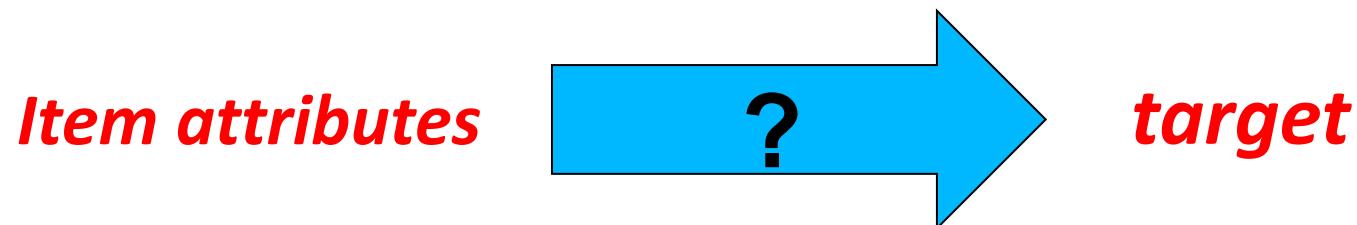
- User/Item Attributes (Clustering, Similarity Learning)
- Ratings (Label: Classification, Numeric: Regression)
- Reviews (Text: NLP, Sentiment Analysis)
- Item pictures (Image: Computer Vision)
- Multimedia Data (Video: Computer Vision, Music:)
- User/Item Network (Network Data: Social Network Analysis, Random Walk)
- Transaction (Sequence: Time Series Analysis, Recurrent Model)

Representation in RS

- Representing attributes
 - Representing rating table
 - Representing review
 - Representing image
 - Representing network
 - Representing sequence
 - Representing multimodal data

Representing attributes

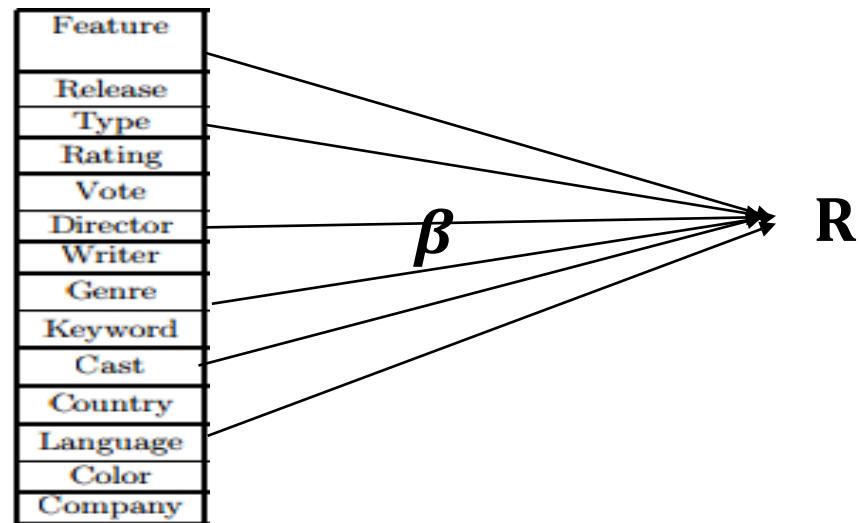
- Attributes are most common used in RS
 - User feature or Item feature
 - Categorical feature or Numerical feature
- Modeling the relationship between ***target*** and given ***attributes***.



Regression-based Rating Prediction

- β is the parameters used to model the importance of each feature.
- R is the ratings given by a user

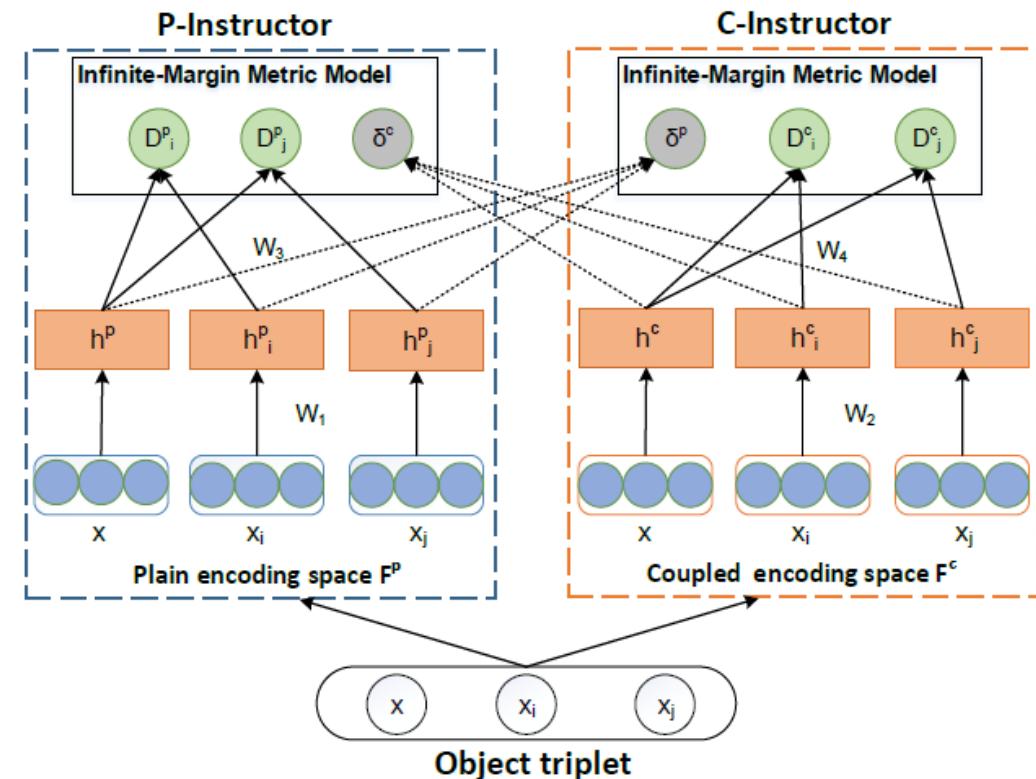
$$R = f(\beta | X) = X\beta$$



Statistical estimation and inference focuses on β

Metric-based Auto-Instructor for Learning Mixed Data Representation

- Representing **categorical feature** and **numerical feature** in one unified feature space
 - Learning couplings between categorical and numerical feature
 - Construct two feature spaces
 - Learning from each other



Representation in RS

- Representing attributes
- **Representing rating table**
- Representing review
- Representing image
- Representing network
- Representing sequence
- Representing multimodal data

User-item rating

- A full matrix $Y \in \mathbb{R}^{N \times M}$

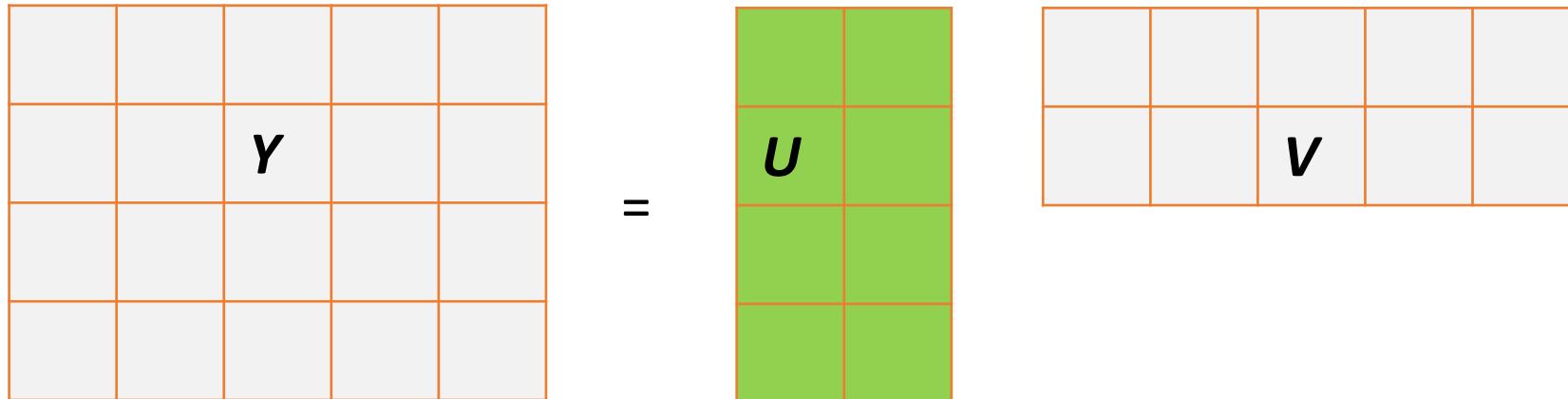
		5			4		2	
	5				5		1	
		4						
	2		3	2	3	2	3	1
	1			5		4		1
		4						
	2				4			
		3						5

Is there other way to represent rating table?

$O(NM)$, if 100,000 rows, 50,000 columns, 4GB memory is needed

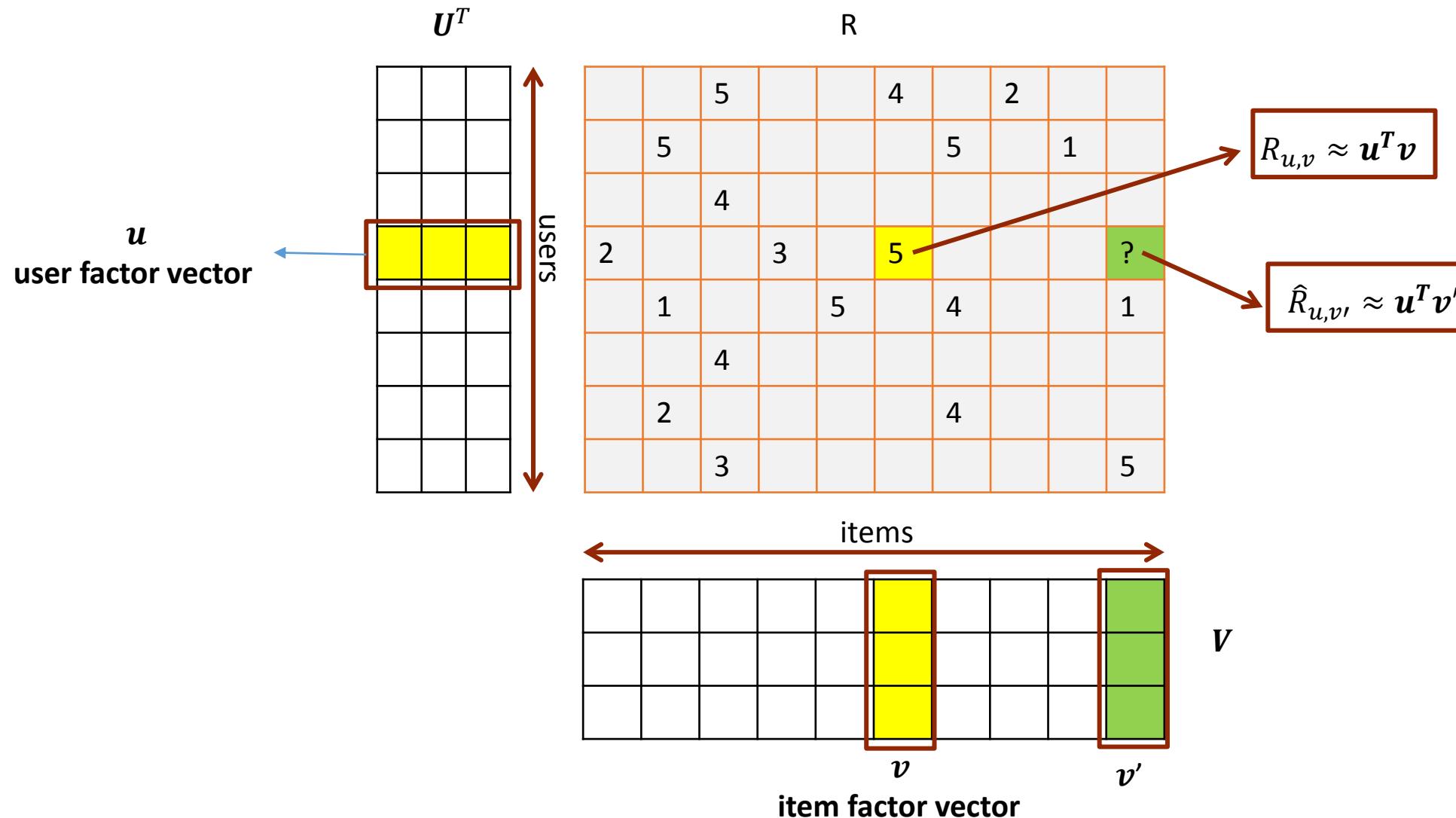
Matrix Factorization

- Approximated by low-rank matrices
 - Given a matrix $Y \in \mathbb{R}^{N \times M}$, we have
 - $Y = U^T V$ where $U = [u_1, \dots, u_N], V = [v_1, \dots, v_M], u_i, v_j \in \mathbb{R}^D$



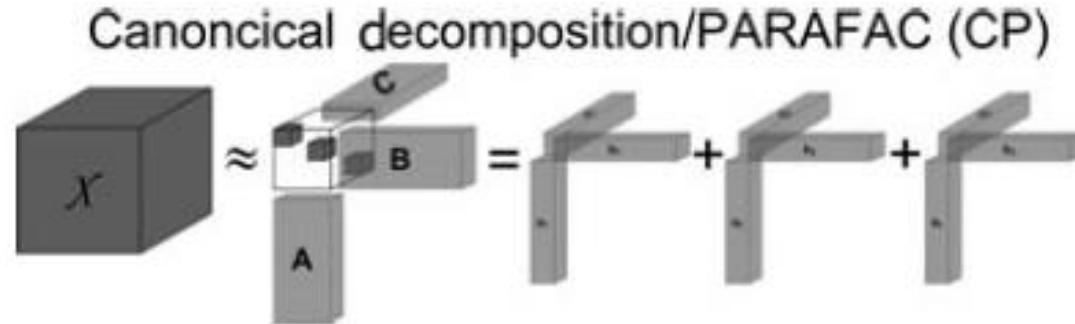
$O(ND + MD)$, if 100,000 rows, 50,000 columns, only 8MB memory is needed

Applying MF for Recommender Systems



Tensor Factorization

- CP Model: $Y = A \circ B \circ C$



Full Storage: $O(\prod_i N_i)$, if $N_i=100,000$, 8PB (10^{15}) memory is needed

Low-rank Storage: $O(D \sum_i N_i)$, if $D=10$, only 24MB memory is needed

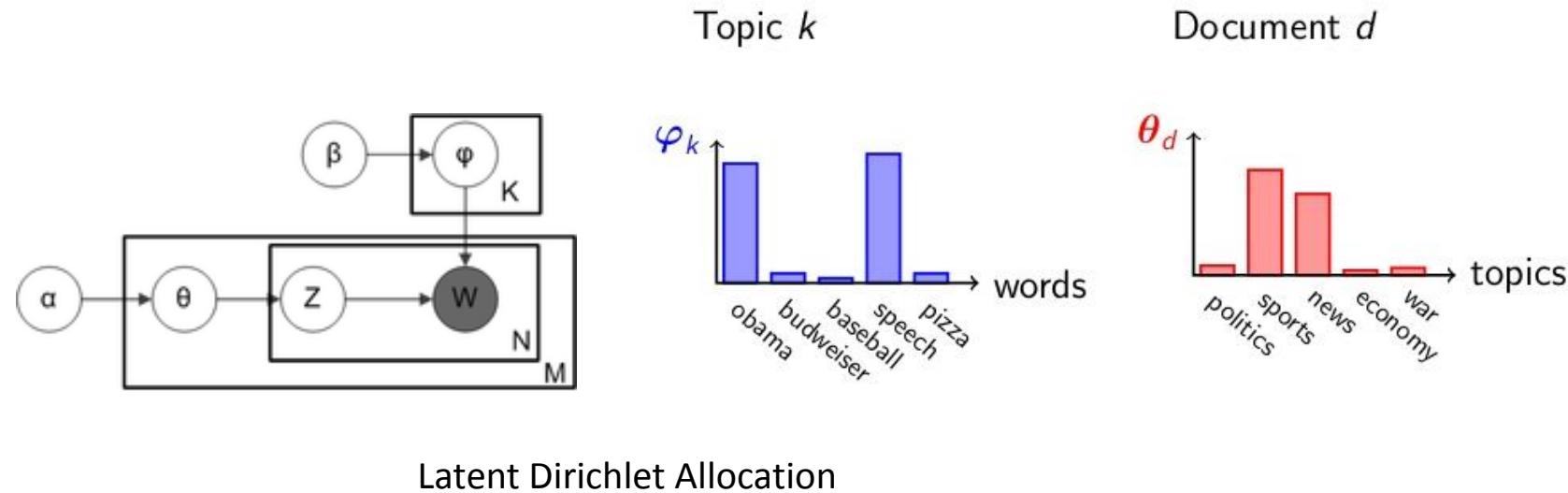
Representation in RS

- Representing attributes
- Representing rating table
- **Representing review**
- Representing image
- Representing network
- Representing sequence
- Representing multimodal data

Representing Text Data

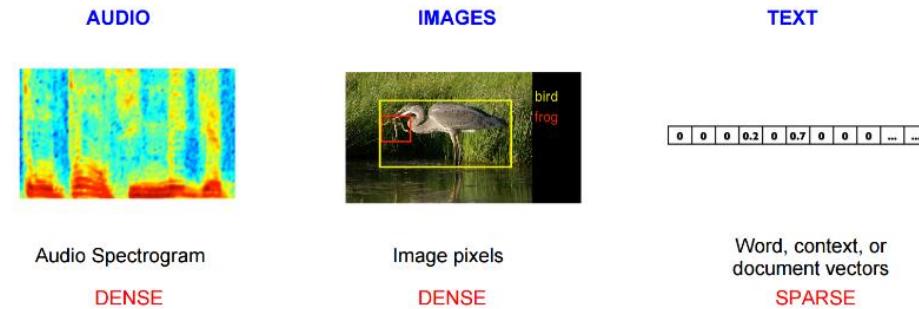
- TF-IDF
- Topic model
 - LSA
 - LDA
 - HDP
- Word embedding
 - Skip-gram
 - CBOW

Recommended TF-IDF weighting schemes		
weighting scheme	document term weight	query term weight
1	$f_{t,d} \cdot \log \frac{N}{n_t}$	$\left(0.5 + 0.5 \frac{f_{t,q}}{\max_t f_{t,q}}\right) \cdot \log \frac{N}{n_t}$
2	$1 + \log f_{t,d}$	$\log\left(1 + \frac{N}{n_t}\right)$
3	$(1 + \log f_{t,d}) \cdot \log \frac{N}{n_t}$	$(1 + \log f_{t,q}) \cdot \log \frac{N}{n_t}$



Word Embedding

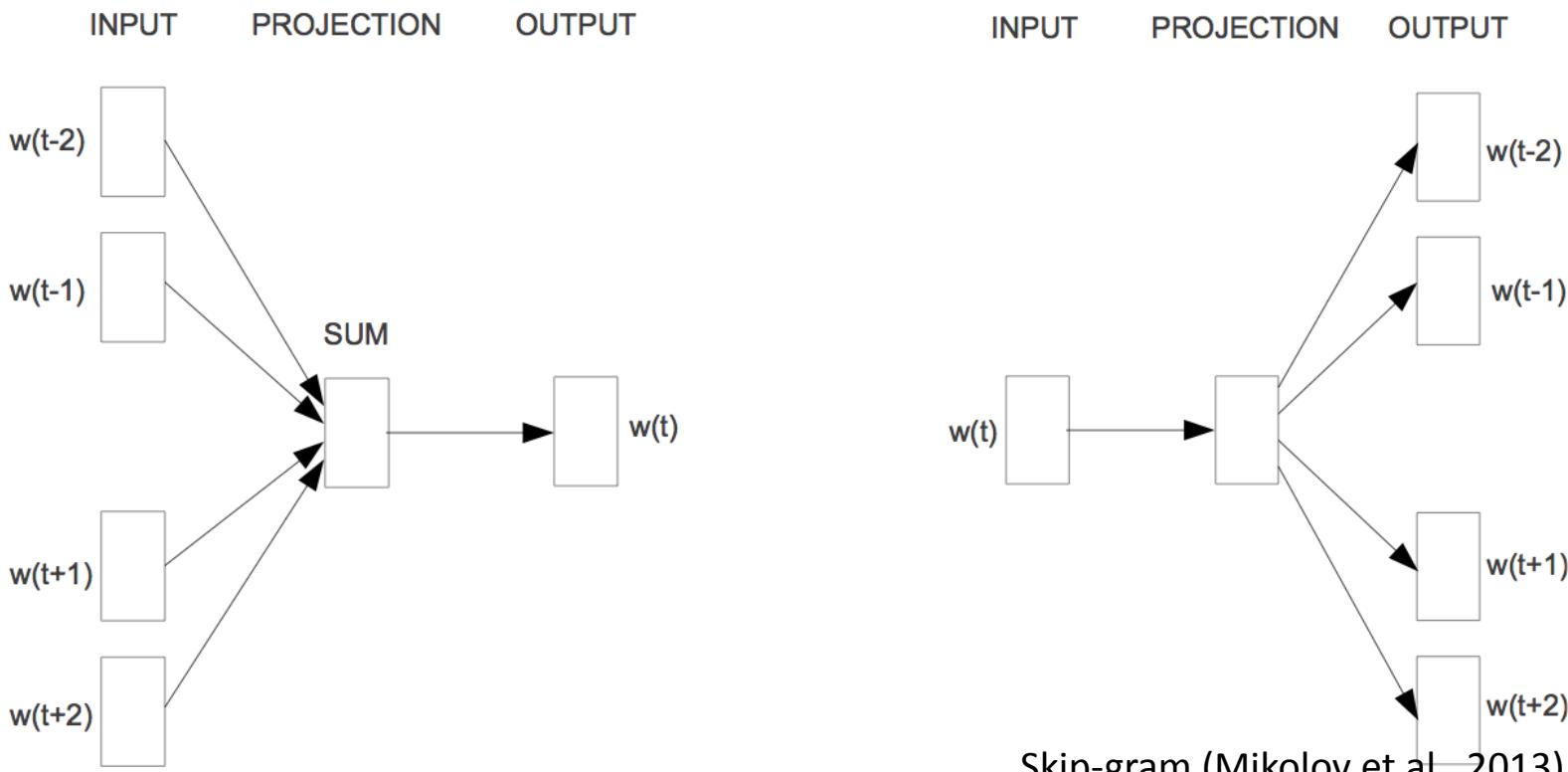
- A word embedding W : words $\rightarrow \mathbb{R}^n$ is a parameterized function mapping words in to low-dimensional vectors.



- *Why Learn Word Embedding?*

- NLP systems traditionally treat words as discrete atomic symbols
 - E.g. 'cats': id 124, 'dogs': id 23, while they are both animals, four-legged, pets, etc.
 - Using vector representations can overcome some of these obstacles.

Word2Vec



Continuous bag-of-words (Mikolov et al., 2013)

Mikolov, T., Corrado, G., Chen, K., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space.

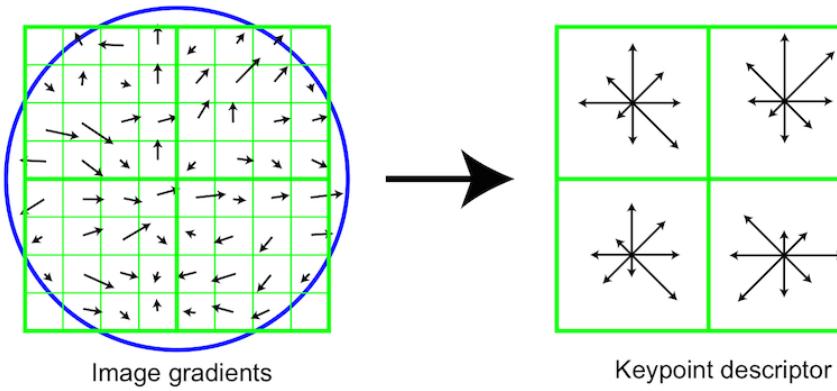
Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionalities

Representation in RS

- Representing attributes
- Representing rating table
- Representing review
- **Representing image**
- Representing network
- Representing sequence
- Representing multimodal data

Representing Image

- SIFT (Scale Invariant Feature Transform)

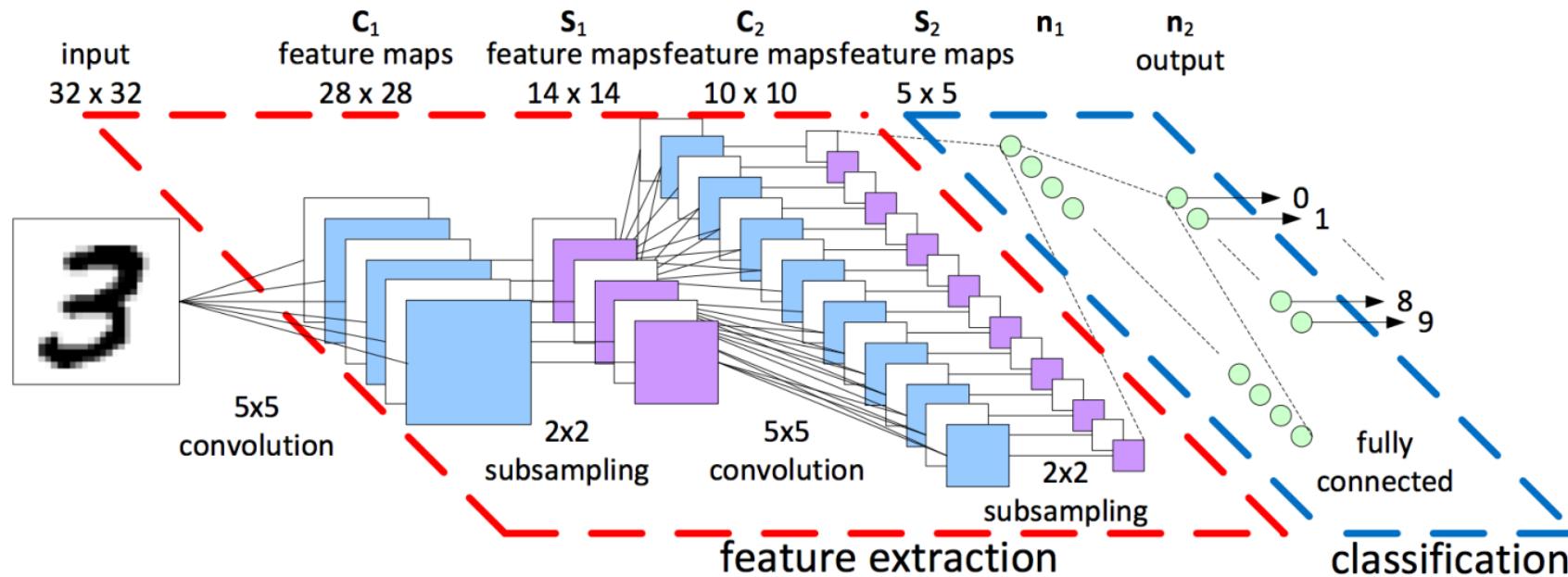


- HOG (Histogram of Oriented Gradients)

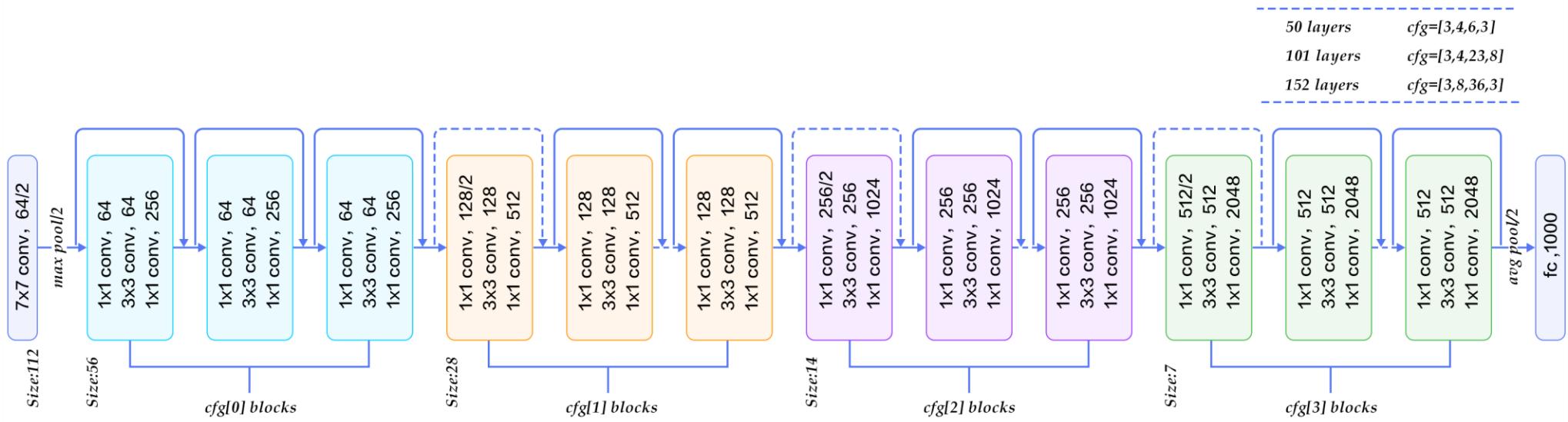


Feature learning for image

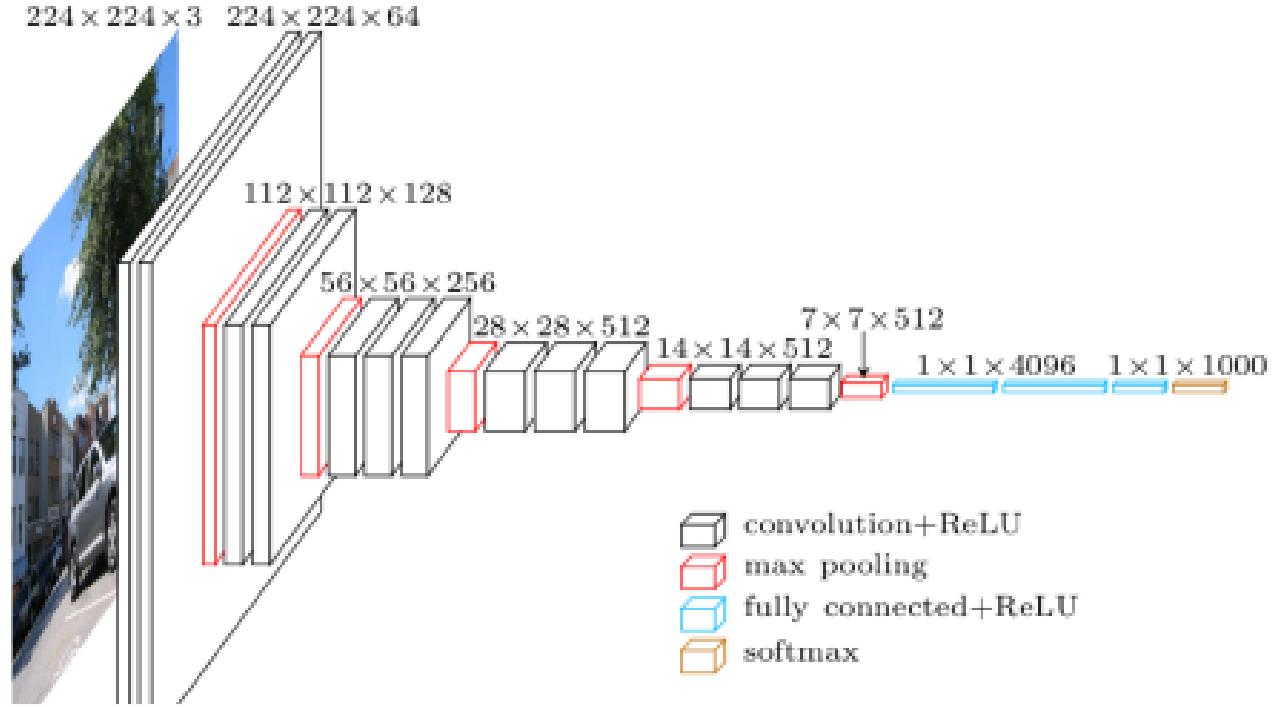
- Deep Learning
 - CNN (Convolutional Neural Networks)



ResNet



VGG



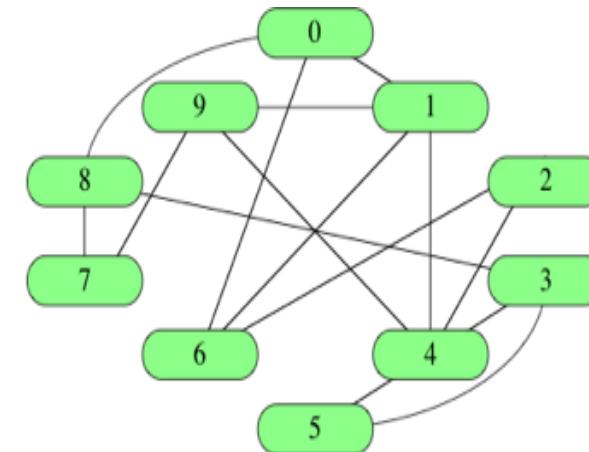
Representation in RS

- Representing attributes
- Representing rating table
- Representing review
- Representing image
- **Representing network**
- Representing sequence
- Representing multimodal data

Representing network

- Adjacency matrix

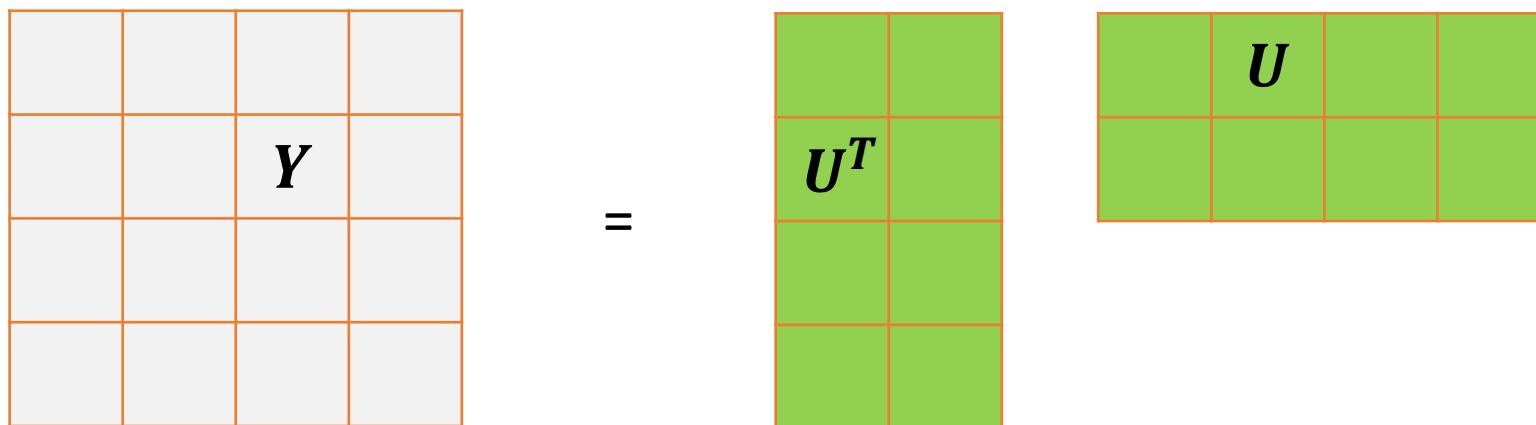
	0	1	2	3	4	5	6	7	8	9
0	0	1	0	0	0	0	1	0	1	0
1	1	0	0	0	1	0	1	0	0	1
2	0	0	0	0	1	0	1	0	0	0
3	0	0	0	0	1	1	0	0	1	0
4	0	1	1	1	0	1	0	0	0	1
5	0	0	0	1	1	0	0	0	0	0
6	1	1	1	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	1	1
8	1	0	0	1	0	0	0	1	0	0
9	0	1	0	0	1	0	0	1	0	0



$O(|V|^2)$, if 100,000 nodes, 40GB memory is needed

Matrix Factorization

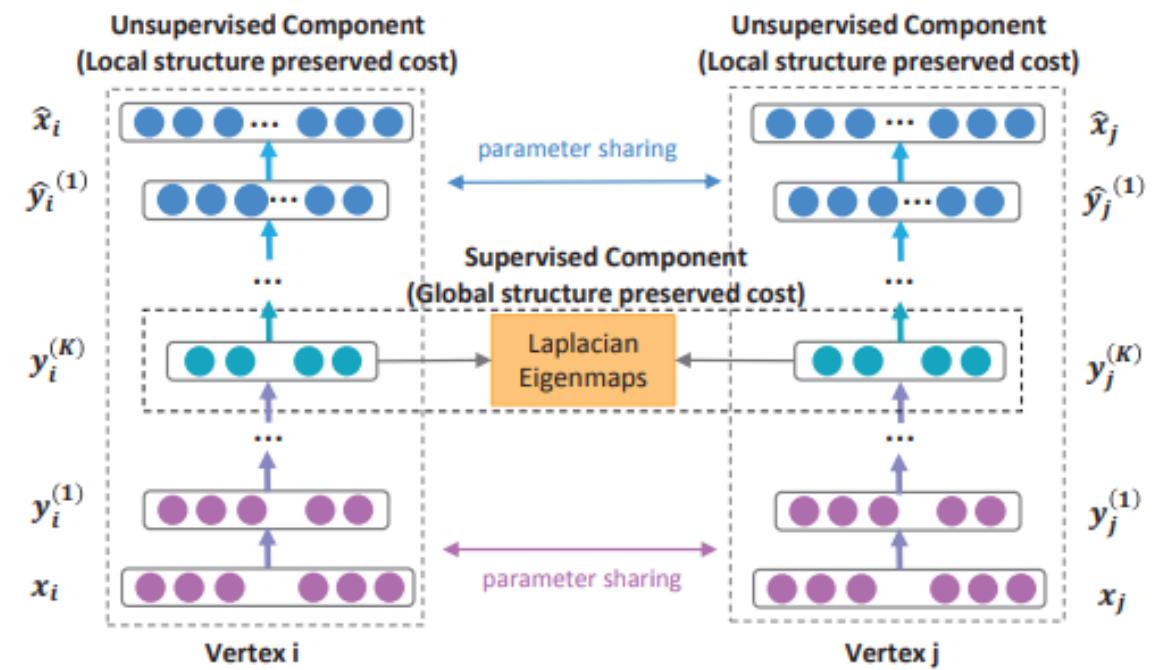
- Approximated by low-rank matrices
 - Given an adjacent matrix $Y \in \mathbb{R}^{N \times N}$, we have $Y = U^T U$



$O(ND)$, if 100,000 nodes and D=10, only 4MB memory is needed

Representing social network

- Network embedding
 - The users in social network can be represented as vectors
 - There are plenty of methods, e.g., LINE, deepwalk, node2vec
 - Deep learning with network embedding

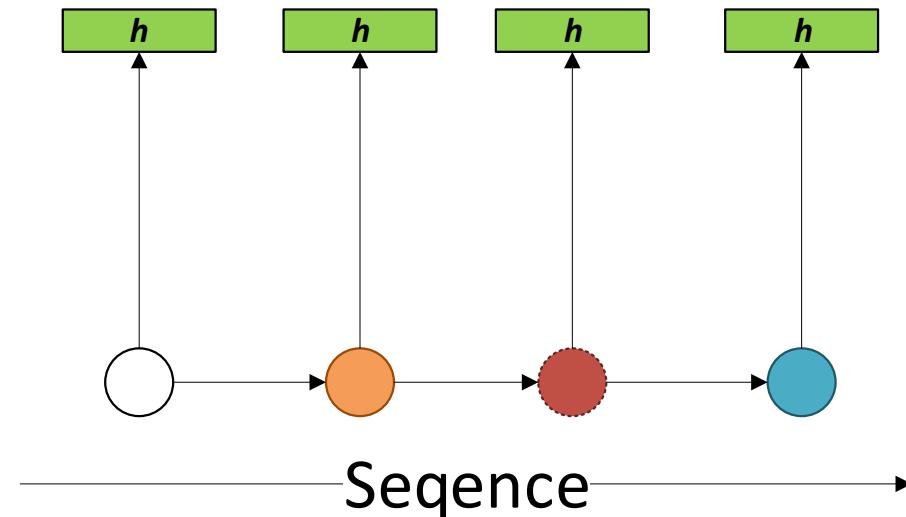


Representation in RS

- Representing attributes
- Representing rating table
- Representing review
- Representing image
- Representing network
- **Representing sequence**
- Representing multimodal data

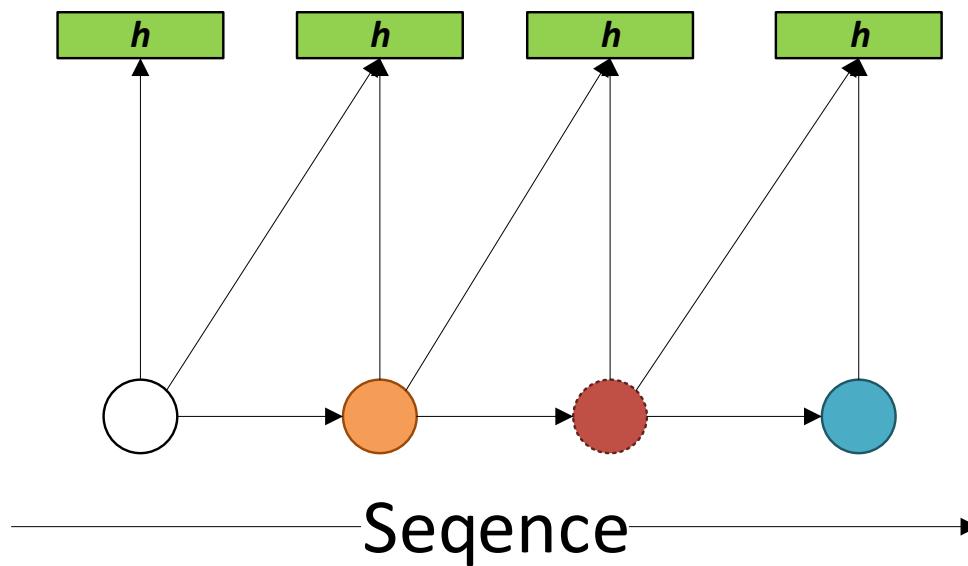
Zero-order Embedding

- Map each item from its into a real-valued vector without considering the sequential dependency
- $h_i = f(o_i)$



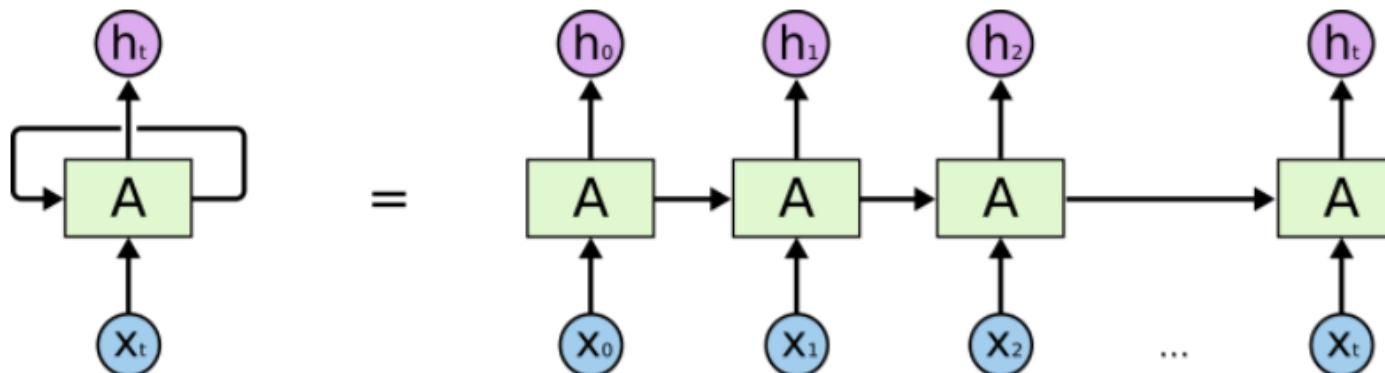
First-order Embedding

- Map an object to embedding conditional on the last object
- $h_t = f(o_t | o_{t-1})$



Embedding higher order information

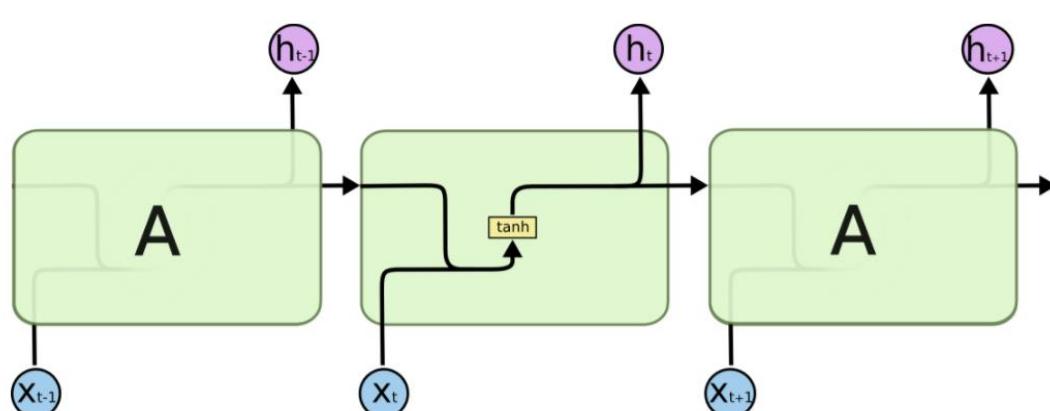
- Raw RNN: Except for the item itself information, the embedding accumulate the information of recent states.
- $h_t = f(o_t|h_{t-1}) = f(o_t|f(o_{t-1}|h_{t-2})) = \dots$



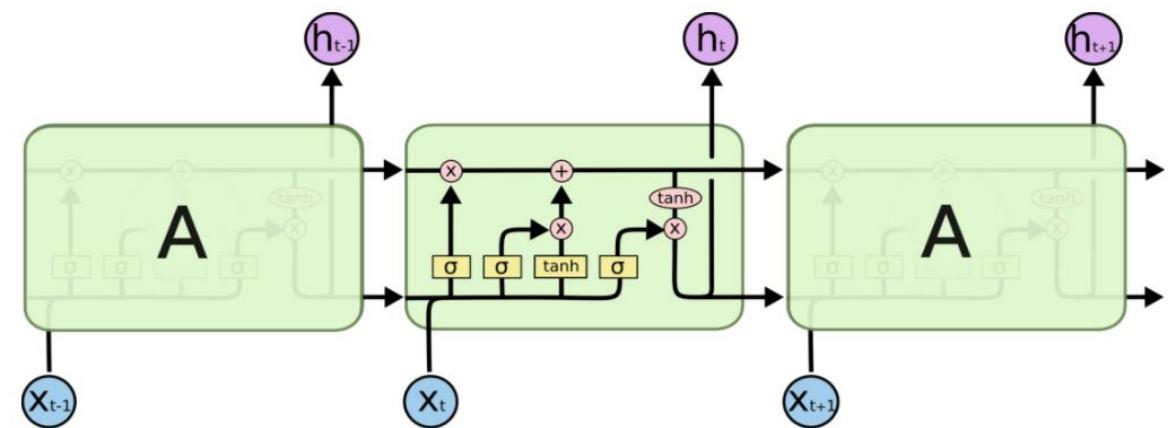
An unrolled recurrent neural network.

Embedding long-short term information

- LSTM models the long and short-term dependencies, where the accumulation of information is controlled by gate modules



The repeating module in a standard RNN contains a single layer.



The repeating module in an LSTM contains four interacting layers.

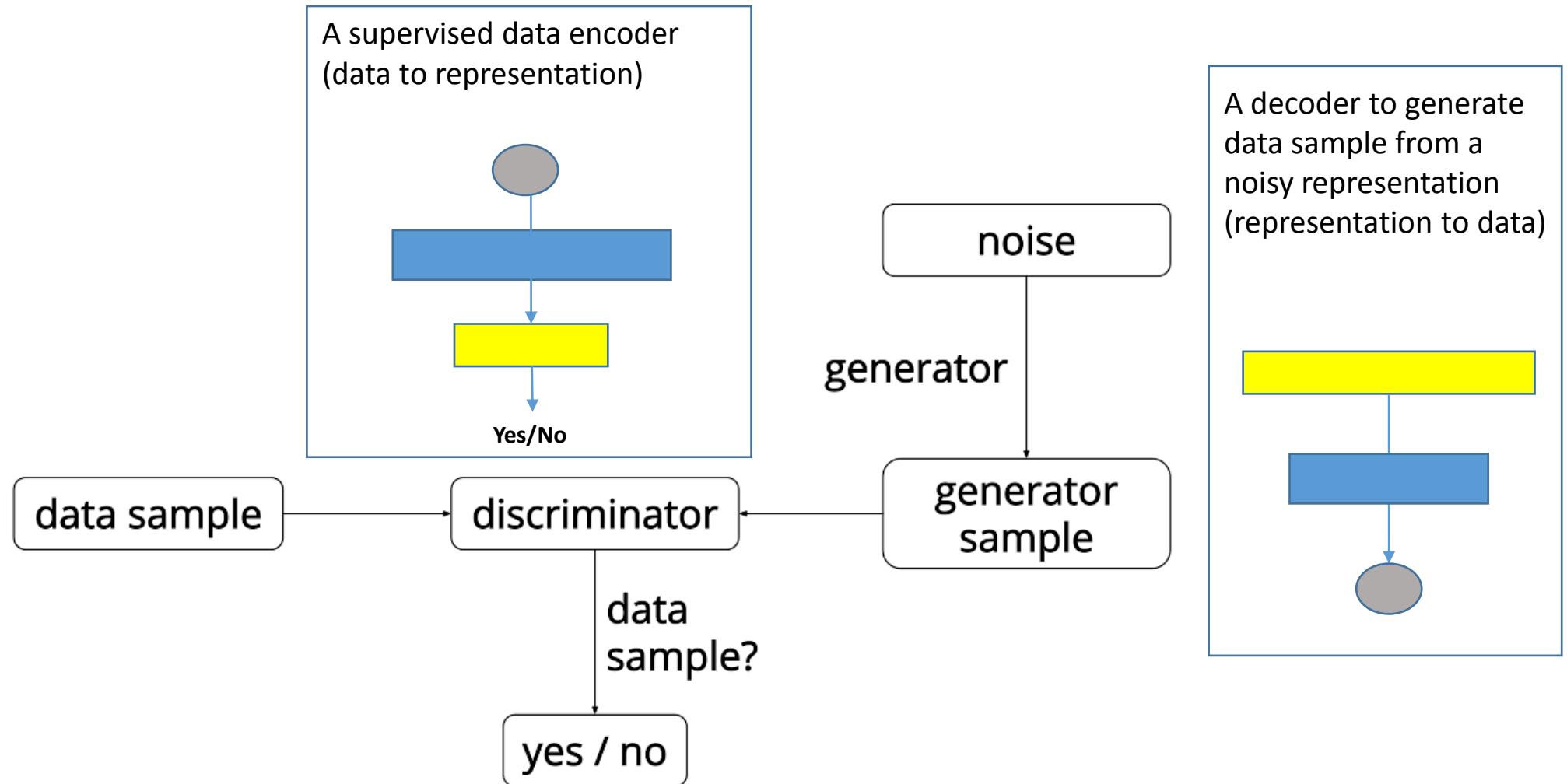
Representation in RS

- Representing attributes
- Representing rating table
- Representing review
- Representing image
- Representing network
- Representing sequence
- **Representing multimodal data**

Generating data from representation: generative model

- Training and sampling from generative models is an excellent test of our ability to **represent** and manipulate high-dimensional probability distributions.
- Generative models can be trained with missing data and can provide predictions on inputs that are missing data, e.g. semi-supervised learning

Generative Adversarial Networks



Text-to-image Synthesis

this small bird has a pink breast and crown, and black primaries and secondaries.



the flower has petals that are bright pinkish purple with white stigma



this magnificent fellow is almost all black with a red crest, and white cheek patch.



this white and yellow flower have thin white petals and a round yellow stamen



This small blue bird has a short pointy beak and brown on its wings



This bird is completely red with black wings and pointy beak



A small sized bird that has a cream belly and a short pointed bill



A small bird with a black head and wings and features grey wings



Figure 23: Text-to-image synthesis with GANs. Image reproduced from Reed *et al.* (2016b).

Figure 25: StackGANs are able to achieve higher output diversity than other GAN-based text-to-image models. Image reproduced from Zhang *et al.* (2016).

GAN for Generating Images by Text

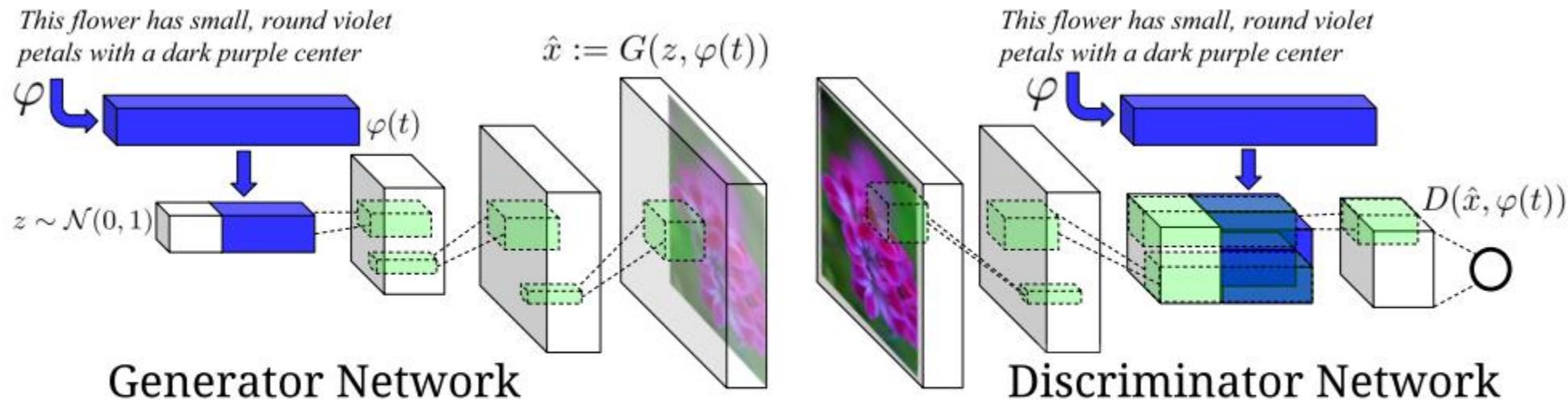


Figure 2. Our text-conditional convolutional GAN architecture. Text encoding $\varphi(t)$ is used by both generator and discriminator. It is projected to a lower-dimensions and depth concatenated with image feature maps for further stages of convolutional processing.

Application of GAN in RS

- Generate virtual data to relieve data sparsity
- Preview virtual images of item from NLP description

Section II:

- Leveraging Complementary Information
 - Cross-domain Recommender Systems
 - Social Recommender Systems
- Leveraging Comprehensive Information
 - Multimodal Recommender Systems
 - Multi-objective Recommender Systems

Complementary Information

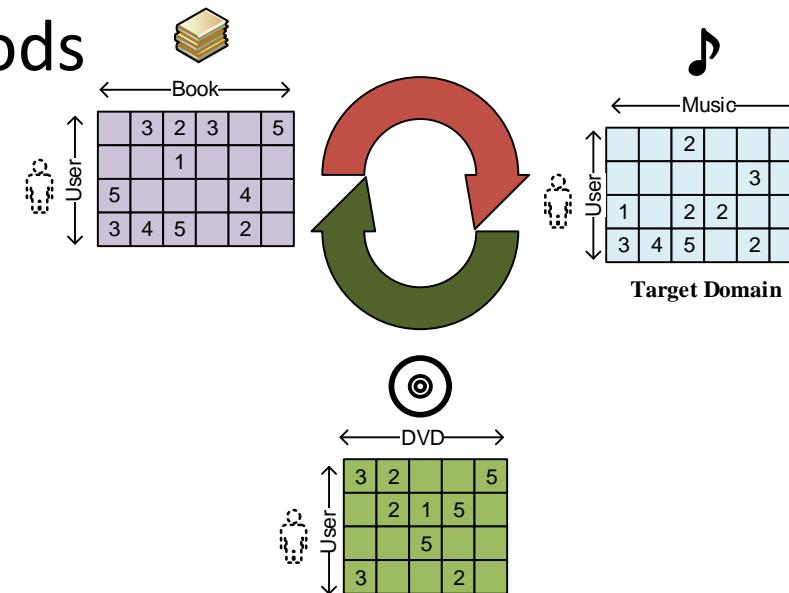
- Single dataset
 - Single data source
 - Single domain
- 
- Multiple datasets
 - Multiple data sources
 - Multiple domains
-
- Typical recommender systems
 - Cross domain RS: Leveraging information from multiple domains
 - Social RS: Propagating information through social relations

Section II:

- Leveraging Complementary Information
 - Cross-domain Recommender Systems
 - Social Recommender Systems
- Leveraging Comprehensive Information
 - Multimodal Recommender Systems
 - Multi-objective Recommender Systems

Cross-Domain Collaborative Filtering

- Leverage information from multiple related domains
 - The existence of *multiple related domains*
 - The user preference from each domain is not independent
- Two main machine learning methods
 - Latent factor models
 - Deep learning models



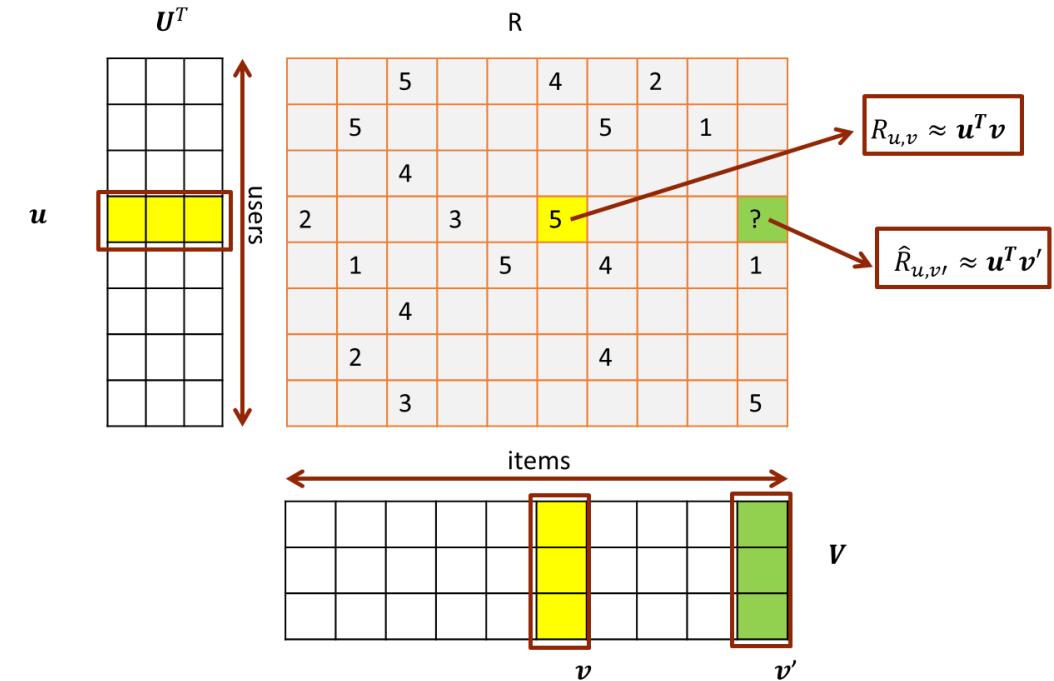
Naïve MF for cross domains

- Concatenating the rating matrices for all domains

User ↑

	3	2	3		5	3	3	2		5								
1		2			1				2	1	5		1	2	2	5		
3	4	5		2	5	4	5	2		3	4	5	2					

← Book → ← DVD → ← Music →

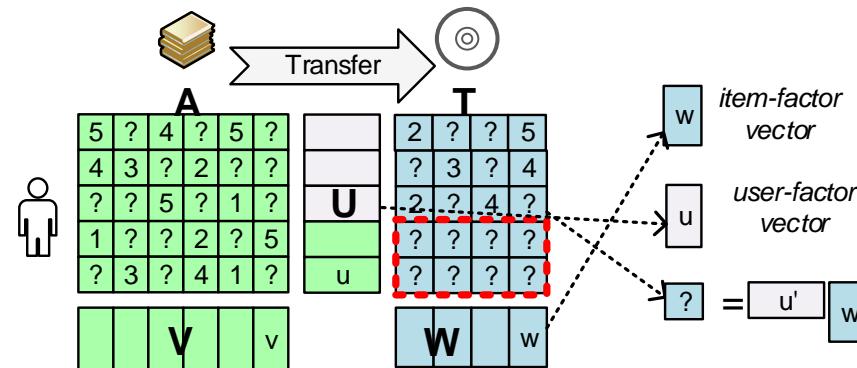


Deficiency

- Each domain has different characteristics
 - The factor of *color* has huge impact on the user preference in *clothes* domain
 - But has little impact on the user preference in *book* domain
- Above methods using the single domain model implicitly assume the *homogeneity* of items.
 - Obviously, such assumption may decrease the prediction accuracy due to the *heterogeneities* of different domains.

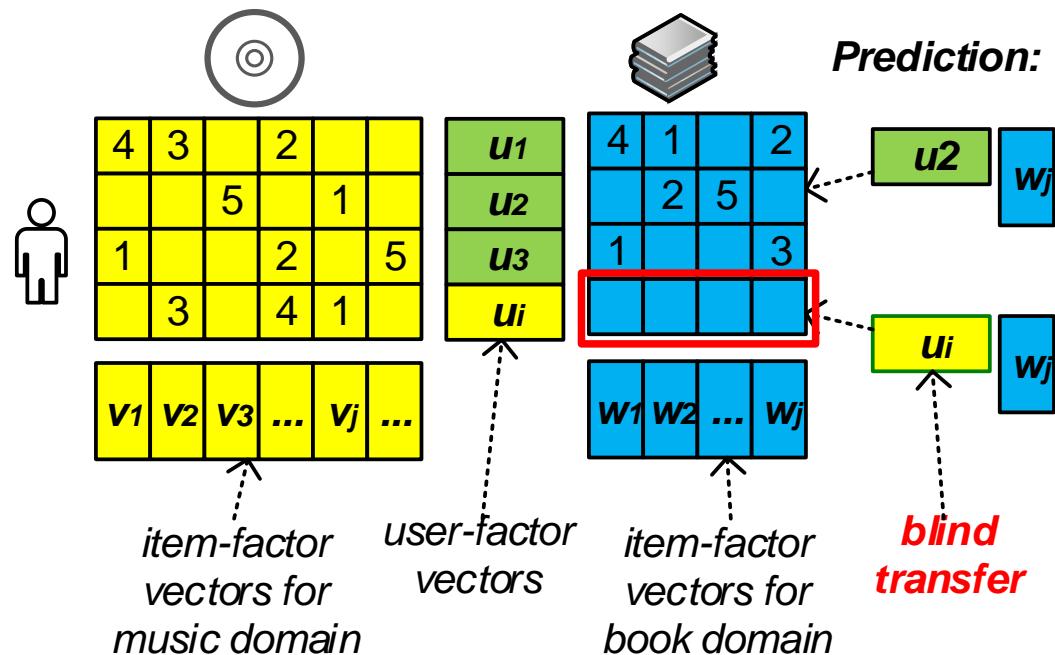
MF based Transfer Learning

- Transfer the knowledge learned from the auxiliary domain to the target domain
 - The user-factor vectors are *co-determined* by the feedback in auxiliary and target domains



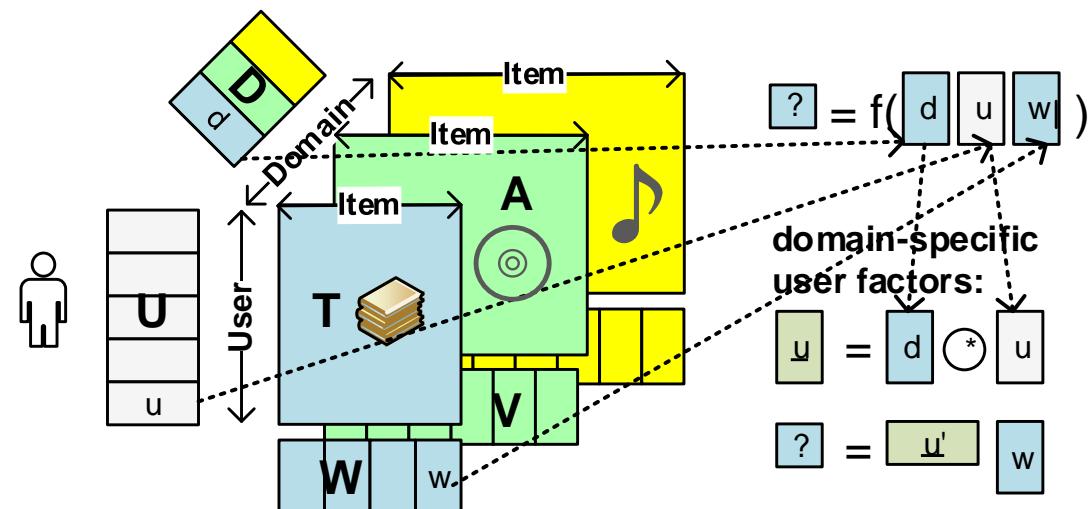
Deficiency

- Blind Transfer
 - If u_i is transferred to the target domain and interacts with heterogeneous item factors, it may yield a poor prediction.



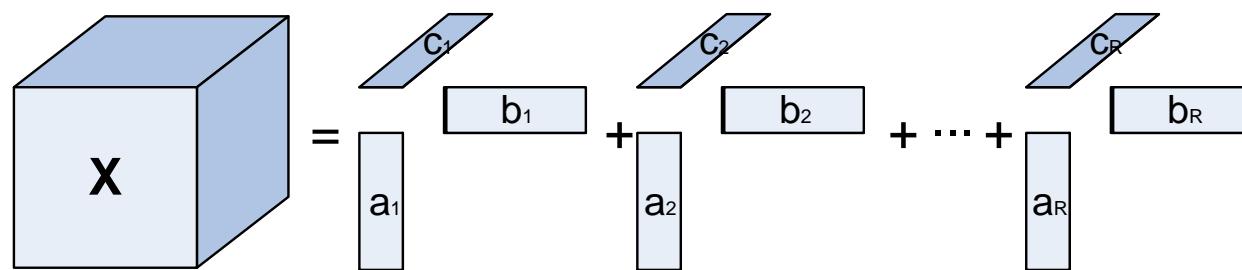
Modeling Domain Heterogeneity

- **Domain factors** is an essential element in cross “domain” problem to model domain heterogeneity
 - Triadic relation ***user-item-domain*** to reveal the domain-specific user preference



Tensor Factorization over Triadic Relation

- Decompose a tensor into a sum of rank-one components
 - $\mathcal{X} = [[\mathbf{A}, \mathbf{B}, \mathbf{C}]] = \sum_{r=1}^R \mathbf{A}_{\cdot,r} \circ \mathbf{B}_{\cdot,r} \circ \mathbf{C}_{\cdot,r}$



Irregular Tensor Factorization

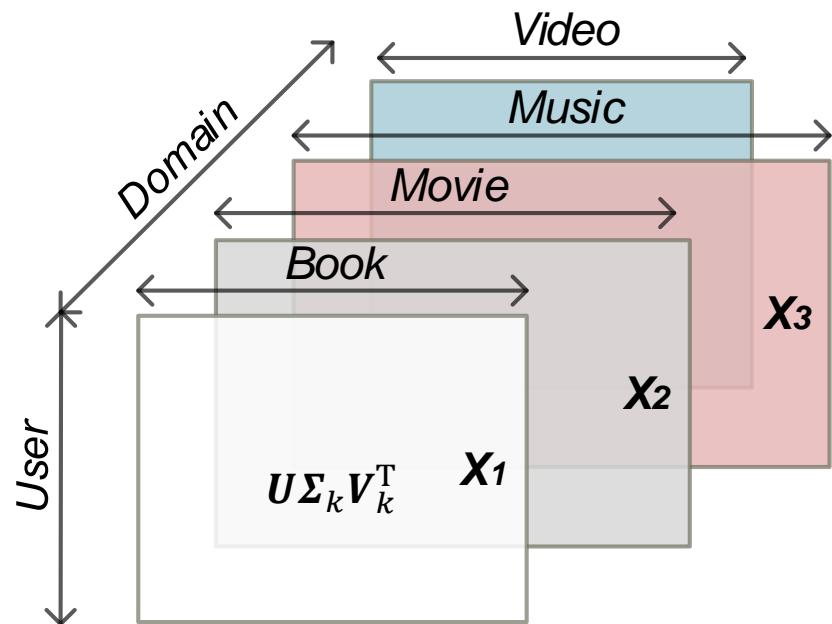
Sum loss over all domains:

$$\underset{U,V,C}{\operatorname{argmin}} \frac{1}{2} \sum_{k=1}^K \left\| W_k \odot (X_k - U \Sigma_k V_k^T) \right\|_F^2 + \frac{\lambda_U}{2} \|U\|^2 + \frac{\lambda_V}{2} \|V\|^2 + \frac{\lambda_C}{2} \|C\|^2$$

With orthonormal constraints, we can obtain equivalent loss:

$$\underset{U,V,C}{\operatorname{argmin}} \frac{1}{2} \left[\underbrace{(\|y - [U, V, C]\|^2 + \lambda_U \|U\|_F^2 + \lambda_V \|V\|_F^2 + \lambda_C \|C\|_F^2)}_{1: \text{Regularized TF Model}} + \underbrace{\sum_k \|\hat{X}_k \odot H_k\|_F^2}_{2: \text{Loss Compensation}} \right]$$

Unfortunately, the number of items for each domain is inconsistent to apply regular tensor factorization.



Epinions dataset (ratings)

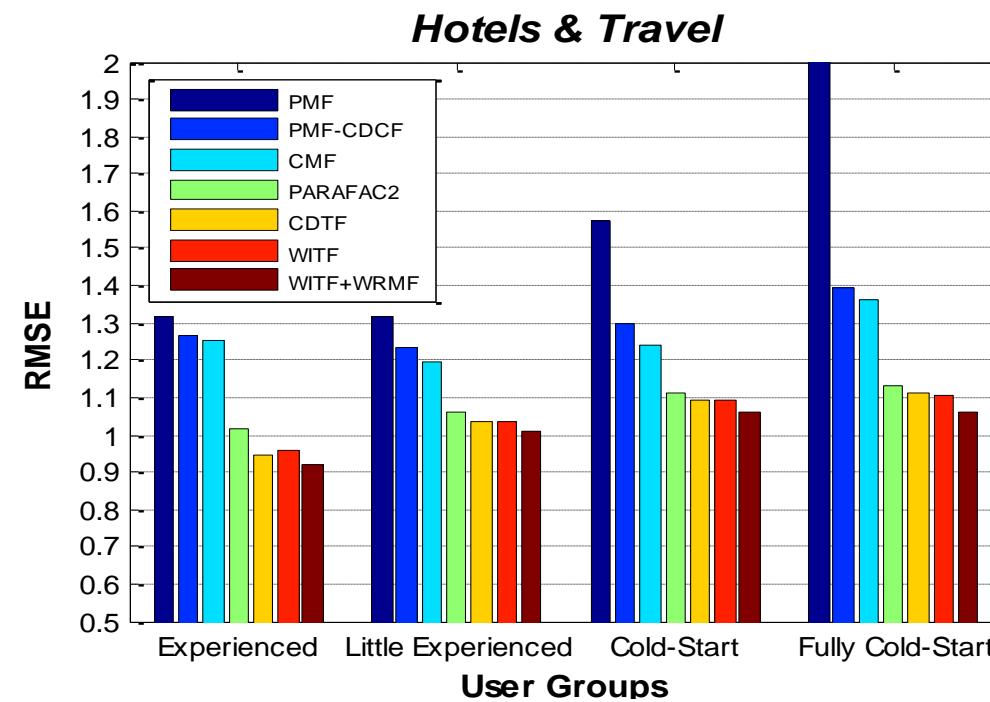
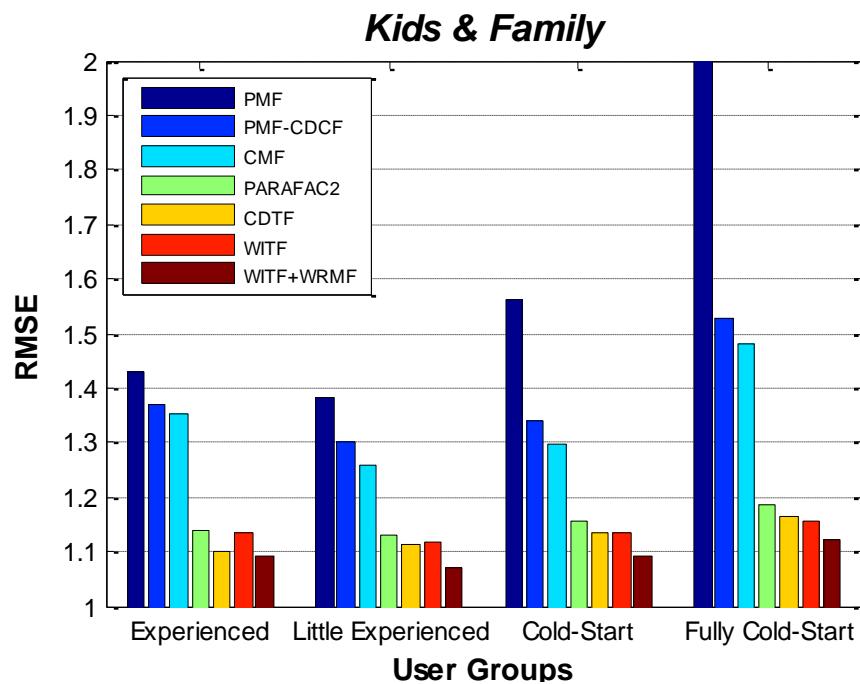
- Covering 5 domains

Domain	# Items	# Ratings / # Users	# Ratings / # Items	Sparsity
Kids & Family*	3,769	4.9309	9.9077	0.0013
Hotels & Travel*	2,545	3.9210	11.6676	0.0015
Restaurants & Gourmet	2,543	3.3394	9.9446	0.0013
Wellness & Beauty	3,852	3.5481	6.9756	0.0009
Home and Garden	2,785	2.6003	7.0707	0.0009

<http://liris.cnrs.fr/red/>

The Prediction Performance over Different Numbers of Training Ratings

- RMSE of comparative methods (the smaller the better)



Tmall.com dataset (clicks)

- One-class preference problem

Domain	# Items	# Clicks / # Users	# Clicks / # Items	Sparsity
$D1^*$	8,179	23.2003	19.7170	0.0028
$D2^*$	6,940	18.5455	18.5749	0.0027
$D3$	5,561	22.5005	28.1246	0.0040
$D4$	6,145	16.0606	18.1671	0.0026

<https://tianchi.aliyun.com/datalab/dataSet.htm?id=5>

The Mean AP@5,10 and nDCG@5,10

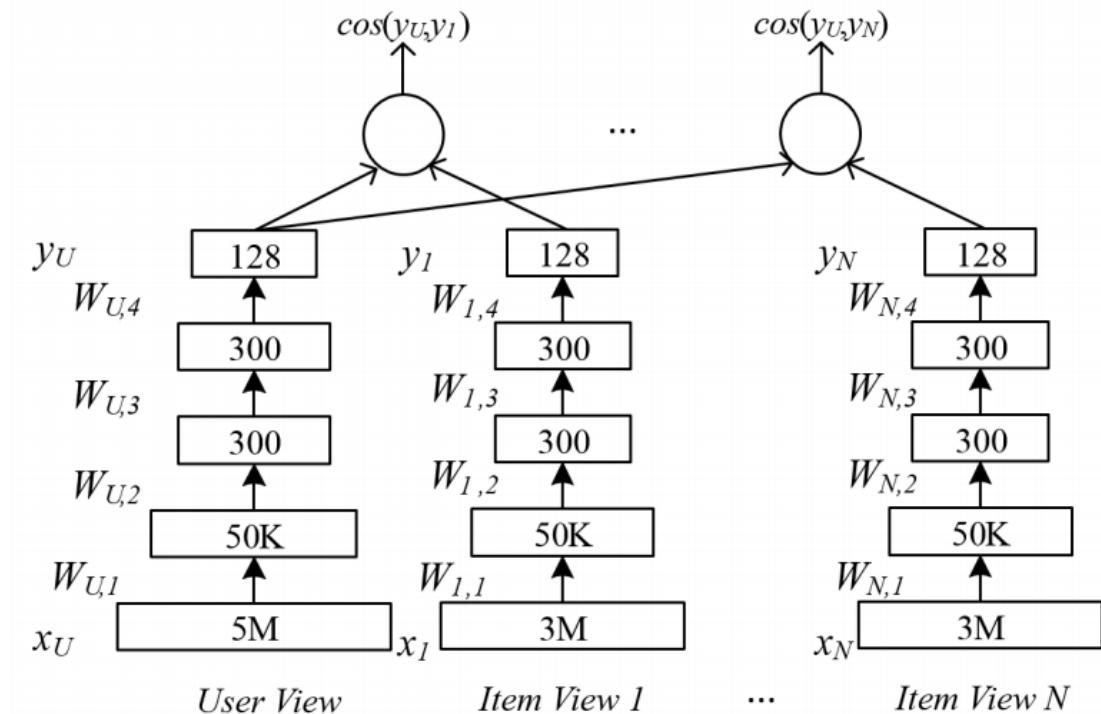
Target Domain Method	D1							
	TR-80%				TR-50%			
	AP@5	AP@20	nDCG@5	nDCG@20	AP@5	AP@20	nDCG@5	nDCG@20
Most-Pop	0.0161^	0.0175^	0.0269^	0.0382^	0.0322^	0.0223^	0.0567^	0.0577^
N-CDCF	0.0252*	0.0240*	0.0441*	0.0465*	0.0352*	0.0210	0.0604*	0.0534
MF-IF	0.0263*	0.0293*	0.0432*	0.0631*	0.0455*	0.0324	0.0813*	0.0854*
MF-IF-CDCF	0.0242*	0.0258*	0.0399*	0.0552*	0.0431*	0.0296	0.0763*	0.0775*
PARAFAC2	0.0213*	0.0226*	0.0350*	0.0476*	0.0395*	0.0267	0.0691*	0.0687*
CDTF-IF	0.0258*	0.0276*	0.0425*	0.0587*	0.0423*	0.0294	0.0758*	0.0767*
WITF	0.0267*	0.0285*	0.0451*	0.0623*	0.0484*	0.0340	0.0849*	0.0872*
WITF+WRMF	0.0271**	0.0290**	0.0462**	0.0643**	0.0486**	0.0343**	0.0851**	0.0879**
Target Domain Method	D2							
	TR-80%				TR-50%			
	AP@5	AP@20	nDCG@5	nDCG@20	AP@5	AP@20	nDCG@5	nDCG@20
Most-Pop	0.0175^	0.0194^	0.0288^	0.0424^	0.0297^	0.0231^	0.0530^	0.0591^
N-CDCF	0.0281*	0.0261*	0.0435*	0.0520*	0.0228	0.0243*	0.0380	0.0357
MF-IF	0.0320*	0.0354*	0.0528*	0.0747*	0.0501*	0.0370*	0.0872**	0.0924**
MF-IF-CDCF	0.0240*	0.0262*	0.0397*	0.0563*	0.0380*	0.0285*	0.0675	0.0724*
PARAFAC2	0.0215*	0.0234*	0.0356*	0.0506*	0.0327*	0.0251*	0.0589*	0.0638*
CDTF-IF	0.0326*	0.0337*	0.0526*	0.0662*	0.0454*	0.0316*	0.0761*	0.0750*
WITF	0.0338*	0.0363*	0.0552*	0.0753*	0.0538*	0.0383*	0.0905*	0.0909*
WITF+WRMF	0.0343**	0.0369**	0.0556**	0.0758**	0.0542**	0.0386**	0.0907**	0.0915*

A Multi-View Deep Learning Approach

- Multi-learning Framework
 - One user view VS. multiple item views
 - DNN to map high-dimensional sparse features (e.g., raw features of users and items) into low dimensional dense features in a joint semantic space

Type	DataSet	UserCnt	Feature Size	Joint Users
User View	Search	20M	3.5M	/
Item View	News Apps Movie/TV	5M 1M 60K	100K 50K 50K	1.5M 210K 16K

Table 1: Statistics of the four data sets used in this paper. The *Joint Users* column indicates the number of common users between each item view and the user view.



User log for Microsoft products

- The data sets:
 - Search engine logs from Bing Web vertical
 - News article browsing history from Bing News vertical
 - App download logs from Windows AppStore
 - Movie/TV view logs from Xbox.

Data Set	Training			Testing		
	Number Of unique users	Number of unique items	Number of training pairs	Number of new users	Number of test pairs for old users	Number of test pairs for new users
Apps Data	200K	55k	2.5M	1K	11K	2K
News Data	1.5M	5M	> 1B	5K	50K	10K
Xbox Data	16K	10K	45K	1K	10K	3K

Mapping between URL domains, News articles and Apps

User View with Single Domain ID Feature	Top Matched News	Top Matched Apps
barackobama.com	Obama to Delay <u>Obamacare</u> Again to Help Democrats Froma Harrop: Democrats should not run away from <u>Obamacare</u> Democratic Senator: I am willing to defy <u>Obama</u> Governor Jindal proposes Republican alternative to <u>Obamacare</u>	7 Minutes Fitter Relax Meditate Escape Sleep Sleep Tracker U.S. Constitution
spiegel.de	Nazi-Era Jerseys on View in <u>World Cup</u> Exhibit 2014 <u>World Cup</u> Day 3 Lessons: Colombia Fun In The Sun... Belgium Vs. Algeria <u>World Cup</u> 2014: Live Stream... Colombia vs. Ivory Coast: <u>Tactical</u> Preview ...	ESPN Cricinfo Golf News RSS Pulse News Dinamalar - Tamil News Paper
linkedin.com	RectorSeal, ... <u>Acquires</u> Assets of Resource Conservation... Berkshire Partners Teams With Glen T. Senk To <u>Co-Invest</u> ... TF Financial: National Penn Bancshares, Inc. to <u>Acquire</u> ... H.I.G. Capital <u>Portfolio</u> Company Surgery Partners to Acquire ...	LinkedIn App LinkedIn Touch The Economist on Windows The Wall Street Journal
babycenter.com	Jenelle Evans' <u>Baby Name</u> : What We Know Catelynn Lowell ... Are Reportedly <u>Pregnant</u> With Baby #2! Jenelle Evans Can Take Drugs During <u>Pregnancy</u> If She Wants Pregnant Jenelle Evans: What Should She Name Her Baby?	Parents Pregnancy & Baby Guide ANIMALS FOR KIDS GAME Minecraft Fan Hub GS Preschool Games

MRR and Precision@1

	Algorithm	All Users		New Users	
		MRR	P@1	MRR	P@1
<i>I</i>	Most Frequent	0.298	0.103	0.303	0.119
	CF	0.337	0.142	/	/
	CCA (TopK) [29]	0.295	0.105	0.295	0.104
	CTR [32]	0.448	0.277	0.319	0.142
<i>II</i>	SV- Kmeans	0.359	0.159	0.336	0.154
	SV-LSH	0.372	0.169	0.339	0.158
	SV-TopK	0.497	0.315	0.436	0.268
<i>III</i>	MV-Kmeans	0.362	0.16	0.339	0.156
	MV-TopK	0.517	0.335	0.466	0.297
	MV-TopK w/ Xbox	0.527	0.347	0.473	0.306

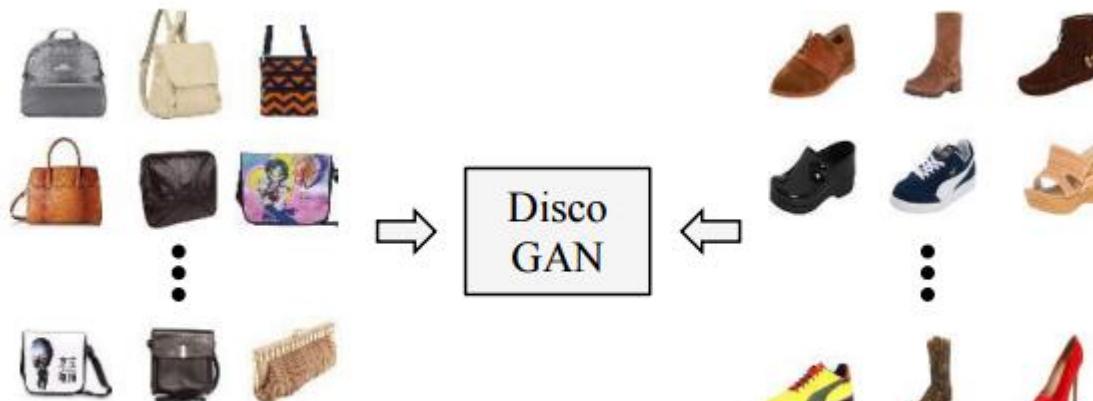
Table 3: Results for different algorithms on Windows Apps Data Set. Type *I* algorithms are baseline methods we compare with. Type *II* are single user-item view methods trained using the original DSSM framework. Type *III* are multi-view DNN models we proposed. The best performance is achieved by training a MV-DNN on all three user-item views with TopK as feature selection method.

	Algorithm	All Users		New Users	
		MRR	P@1	MRR	P@1
<i>I</i>	Most Frequent	0.301	0.111	0.305	0.111
	CTR [32]	0.427	0.215	0.276	0.123
	SV-Kmeans	0.386	0.192	0.294	0.143
<i>II</i>	SV-LSH	0.45	0.247	0.34	0.186
	SV-TopK	0.486	0.286	0.358	0.208
	MV-Kmeans	0.391	0.194	0.296	0.145
<i>III</i>	MV-TopK	0.494	0.303	0.368	0.222
	MV-TopK w/ Xbox	0.503	0.321	0.398	0.245

Table 4: Results for the News Data Set. Similarly, the best performance is achieved by our multi-view models. Note that due to the extreme big size of this data set ($> 1\text{B}$ entries), traditional algorithms like CF (SVD) and CCA failed to handle it due to memory constraint.

Content-based Cross-domain Recommendation with Generative Adversarial Networks

- Discovering cross-domain relations given unpaired data.



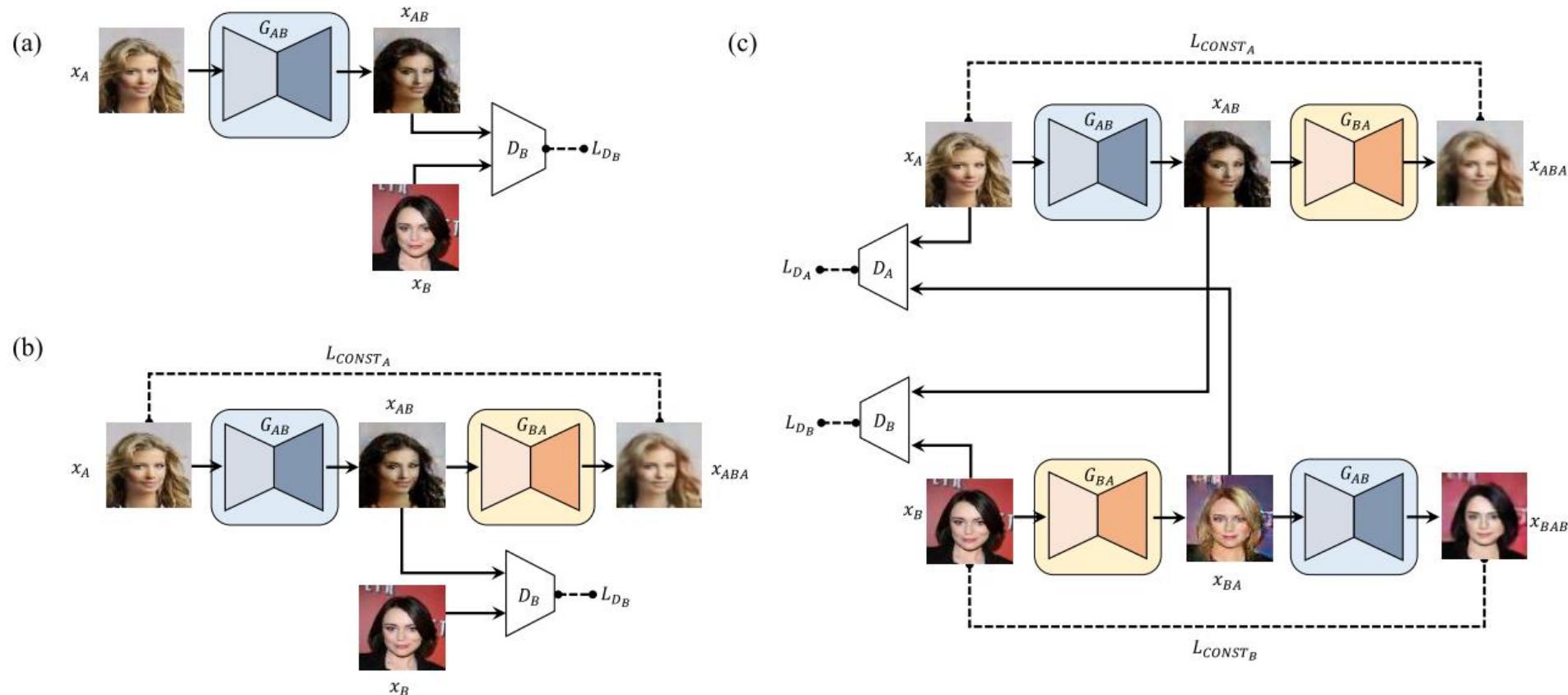
(a) Learning cross-domain relations **without any extra label**



(b) Handbag images (input) & **Generated** shoe images (output)

(c) Shoe images (input) & **Generated** handbag images (output)

DiscoGAN for unpaired, unlabeled datasets

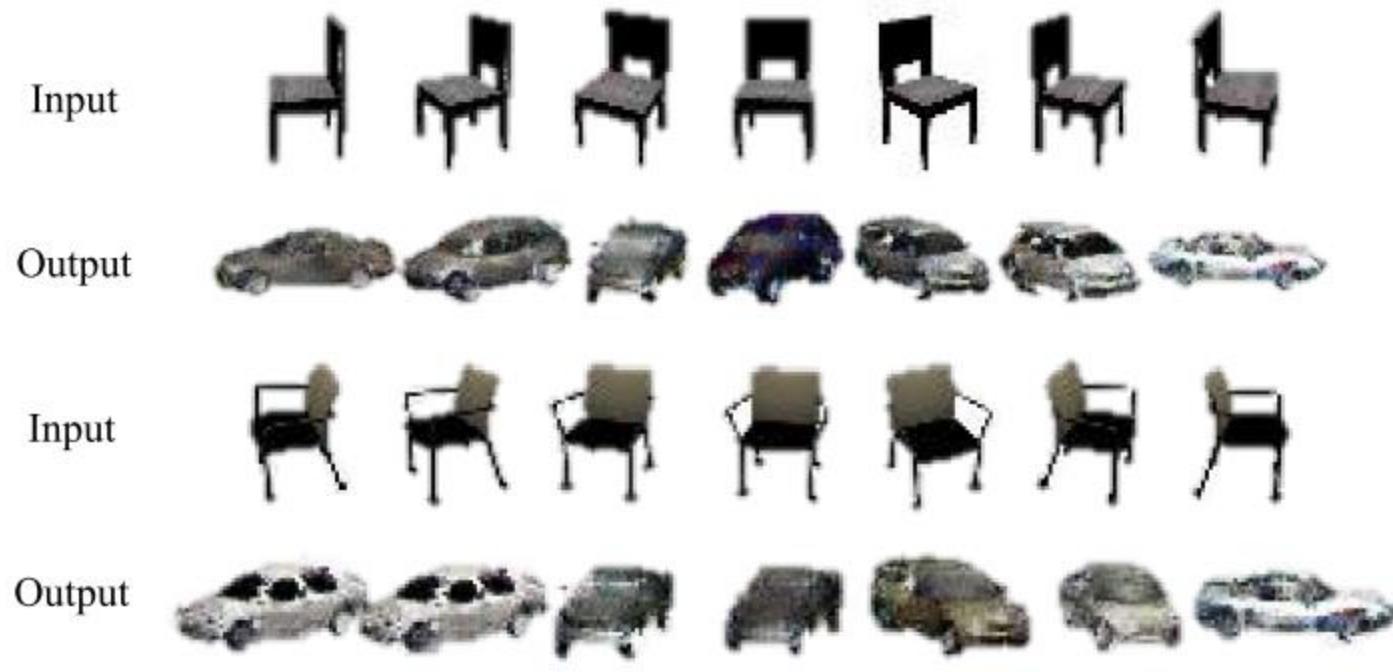


(a) Standard GAN (b) GAN with a reconstruction loss, (c) DiscoGAN designs two coupled GAN between two unpaired, unlabeled datasets.

Datasets

- Car dataset (Fidler et al., 2012)
 - Fidler, S., Dickinson, S., and Urtasun, R. 3d object detection and viewpoint estimation with a deformable 3d cuboid model. In NIPS, 2012.
- Chair dataset (Paysan et al., 2009)
 - Aubry, M., Maturana, D., Efros, A. A., Russell, B., and Sivic, J. Seeing 3d chairs: Exemplar part-based 2d-3d alignment using a large dataset of cad models. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
- Handbags dataset
 - Zhu, Jun-Yan, Krähenbühl, Philipp, Shechtman, Eli, and Efros, Alexei A. Generative visual manipulation on the natural image manifold. In proceedings of European Conference on Computer Vision (ECCV), 2016.
- Shoe dataset
 - Yu, A. and Grauman, K. Fine-grained visual comparisons with local learning. In Computer Vision and Pattern Recognition (CVPR), June 2014.

Chair to Car Translation



(a) Chair to Car

Discovering relations of images from visually very different object classes.
DiscoGAN is trained on chair and car images

Recommend Items from Sketches



(a)



(b)



(c)

- (a) colored images of handbags are generated from sketches of handbags
- (b) colored images of shoes are generated from sketches of shoes
- (c) sketches of handbags are generated from colored images of handbags

Open issues and directions

- Information and Influence adaption
 - What information should be imposed from which domains?
 - How much information should be imposed?
 - How to integrate the heterogeneous information from multiple domains?
- Non-overlap cross-domain learning
 - Joint learning complementary information without overlapped users and items
- Zero-shot learning for cross-domain RS
 - Deal with unseen new items

Section II:

- Leveraging Complementary Information
 - Cross-domain Recommender Systems
 - Social Recommender Systems
- Leveraging Comprehensive Information
 - Multimodal Recommender Systems
 - Multi-objective Recommender Systems

Social Recommendation

- The growth of social media usage

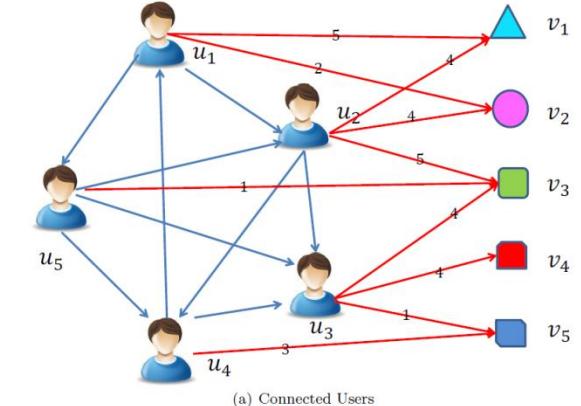


Social Recommendation

- Specific application of cross domain RS
- Recommendation + social relations
 - Latent factor model:
 - Co-factorization
 - Regularization methods
 - Deep learning model

	v_1	v_2	v_3	v_4	v_5
u_1	5	?	2	?	?
u_2	4	4	5	?	?
u_3	?	?	4	4	1
u_4	?	?	?	?	3
u_5	?	?	1	?	?

(b) Traditional Recommender Systems



	v_1	v_2	v_3	v_4	v_5
u_1	5	?	2	?	?
u_2	4	4	5	?	?
u_3	?	?	4	4	1
u_4	?	?	?	?	3
u_5	?	?	1	?	?

(c) Social Recommender Systems

	u_1	u_2	u_3	u_4	u_5
u_1	0	1	0	0	1
u_2	0	0	1	1	0
u_3	0	0	0	0	0
u_4	1	0	1	0	0
u_5	0	1	1	1	0

Social recommendation using probabilistic matrix factorization

- Integrating social network structure and the user-item rating matrix
 - Connecting through the shared user latent feature space

$$\min \sum_{i=1}^n \sum_{u_k \in \mathcal{N}_i} (\mathbf{S}_{ik} - \mathbf{u}_i^\top \mathbf{z}_k)^2,$$

$$\begin{aligned} & \min_{\mathbf{U}, \mathbf{V}, \mathbf{Z}} \|\mathbf{W} \odot (\mathbf{R} - \mathbf{U}^\top \mathbf{V})\|_F^2 + \alpha \sum_{i=1}^n \sum_{u_k \in \mathcal{N}_i} (\mathbf{S}_{ik} - \mathbf{u}_i^\top \mathbf{z}_k)^2 \\ & + \lambda(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2 + \|\mathbf{Z}\|_F^2) \end{aligned}$$

MF with social trust propagation

- Based on the assumption of trust-aware recommender
 - users have similar tastes with other users they trust
 - the transitivity of trust and propagate trust to indirect neighbors in the social network.

$$\begin{aligned} & \min \sum_{i=1}^n (\mathbf{u}_i - \sum_{u_k \in \mathcal{N}_i} \mathbf{S}_{ik} \mathbf{u}_k)^2 \\ & \min_{\mathbf{U}, \mathbf{V}} \|\mathbf{W} \odot (\mathbf{R} - \mathbf{U}^\top \mathbf{V})\|_F^2 + \alpha \sum_{i=1}^n (\mathbf{u}_i - \sum_{u_k \in \mathcal{N}_i} \mathbf{S}_{ik} \mathbf{u}_k)^2 \\ & \quad + \lambda (\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2) \end{aligned}$$

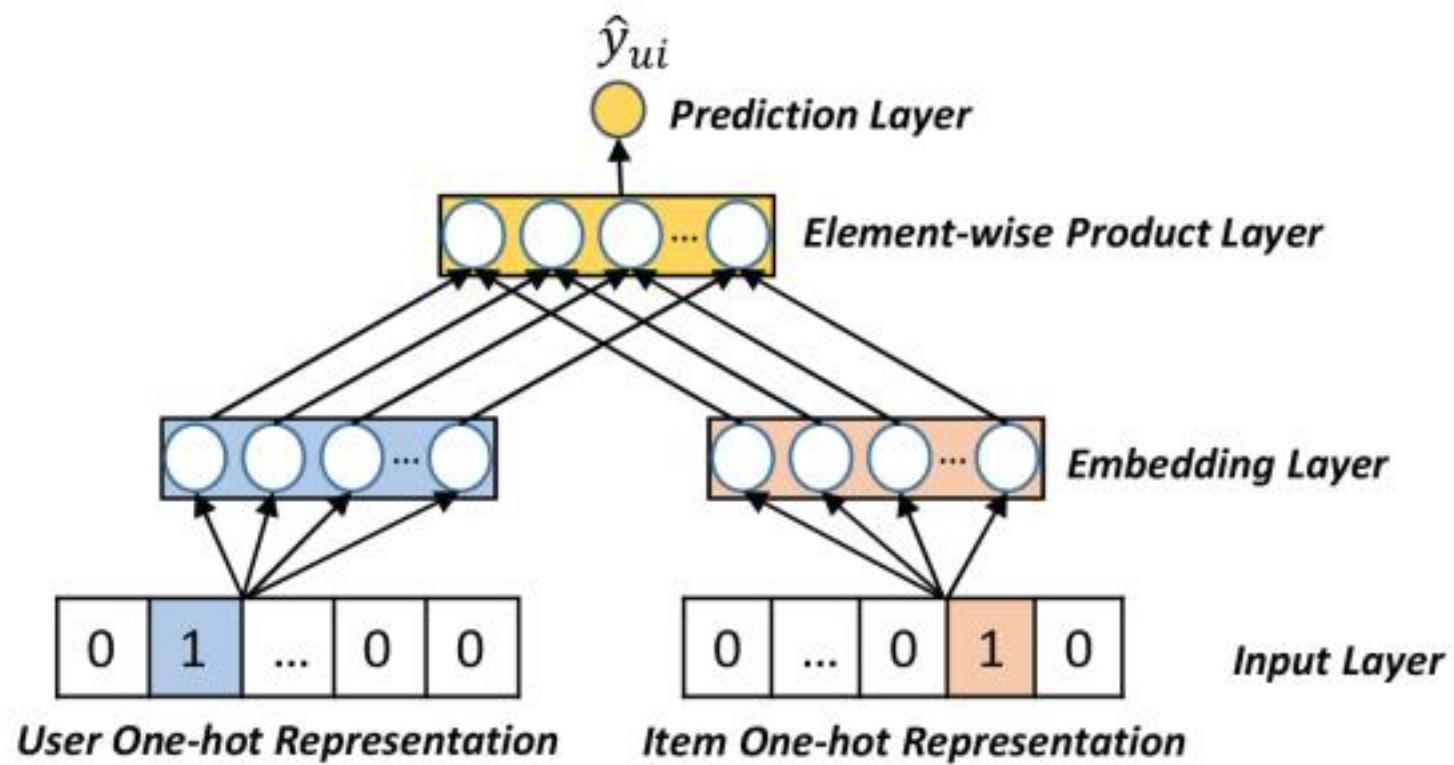
Recommender Systems with Social Regularization

- Two important assumptions
 - “trust relationships” are different from “social friendships” in many aspects.
 - The tastes of one user’s friends may vary significantly

$$\begin{aligned} & \min \sum_{i=1}^n \sum_{u_k \in \mathcal{N}_i} S_{ik} (\mathbf{u}_i - \mathbf{u}_k)^2 \\ & \min_{\mathbf{U}, \mathbf{V}} \|\mathbf{W} \odot (\mathbf{R} - \mathbf{U}^\top \mathbf{V})\|_F^2 + \alpha \sum_{i=1}^n \sum_{u_k \in \mathcal{N}_i} S_{ik} (\mathbf{u}_i - \mathbf{u}_k)^2 \\ & \quad + \lambda (\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2) \end{aligned}$$

Neural network with social recommendation

- Matrix factorization can be regarded as a shallow neural network

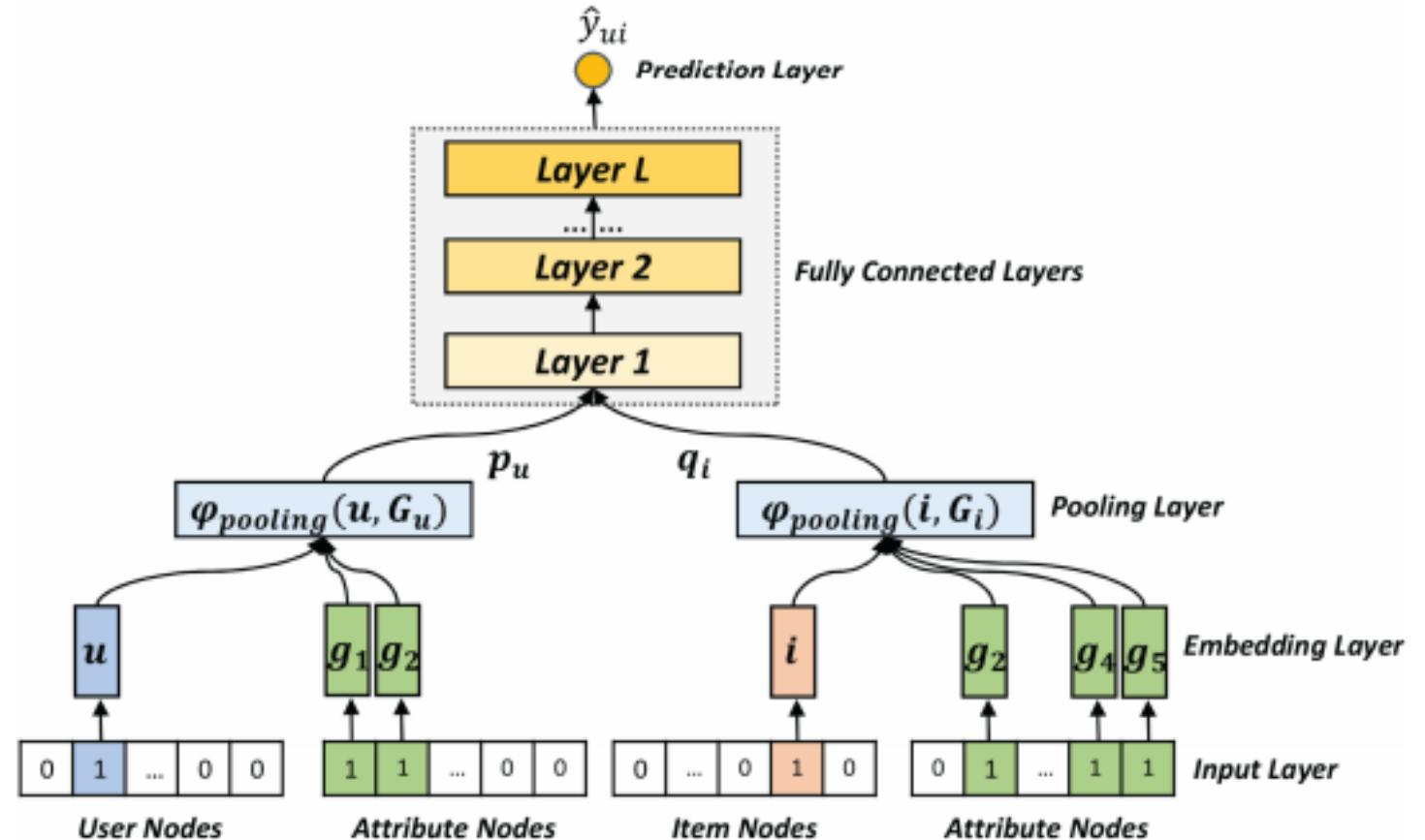


Deep neural network with social recommendation

- The key of integrating rating and social information is representation
 - How to project items and users in rating information and users in social domain into the same embedding space
- Deep neural network is a better option than MF
 - More complex relation
 - Non-linear relation
 - Higher-order correlations

Recommending Items from Information Domains to Social Users

- Unify social users and items with embedding layer and pooling layer
- Bridge the social user information and items with users in rating table (bridge user)



Open issues

- How to deal with the large amount of social data?
 - Billions of nodes
 - Links change every day, every hour, every second
- How to recognize the influential nodes for recommendation?
 - Not all nodes contribute to the recommendation
 - The same nodes may have different influence on different targets
- How to incorporate more social information without harm to users' privacy?
 - More personal information may benefit the recommendation quality
 - Keeping the users' privacy is the top priority

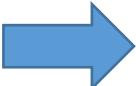
Directions

- Network embedding and learning
 - Mapping user raw information and social relationships to high-level representation
- Memory mechanism
 - Representation learning on social activity sequence
- Dynamic model
 - Using temporal model to capture the shift of social relationships
 - Using neural networks to capture dynamic group coupling (cf. Section III)

Section II:

- Leveraging Complementary Information
 - Cross-domain Recommender Systems
 - Social Recommender Systems
- Leveraging Comprehensive Information
 - Multimodal Recommender Systems
 - Multi-objective Recommender Systems

Comprehensive Information

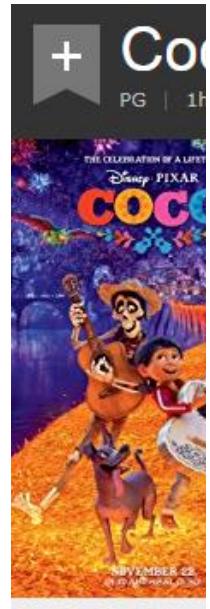
- Incomplete data (e.g. description, images w.r.t. items; lacks of some user attributes)
 - Single criterion, e.g. accuracy is not enough.
- 
- Jointly incorporating information from multiple data types (description + images + attributes)
 - Considering multiple aspects for jointly optimization to improve user experience and business benefits

Typical recommender systems

- Multimodal RS: Modeling multi-type data to obtain more comprehensive information
- Multi-objective RS: Optimize recommendation from multiple aspects.

Learning Comprehensive Information

- Attributes
- Rating
- Review
- Social
- Image
-

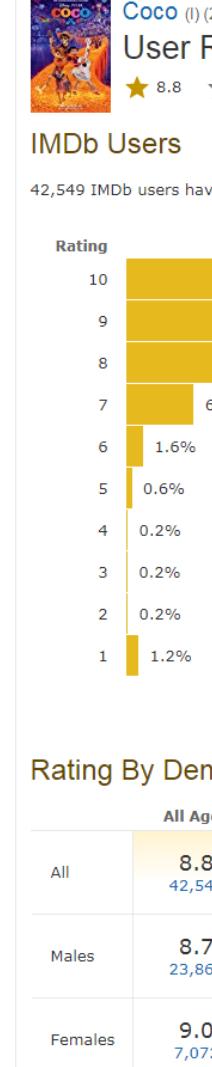


Get Show
In 37 thea

Aspiring musicia
of the Dead to f

Directors: Lee
Writers: Lee U
Stars: Anthony

81 Metascore
From metacritic



Coco (I) (2017)
User Rating
8.8 R

IMDb Users

42,549 IMDb users have given a rating

Coco (I) (2017)
User Review
Add a Review

184 Reviews

10/10

Gracias Pixar
mryohual 27 October 2017

I'm Mexican and all I can say is I remembered all my childhood beautifully, the music, the colors, the culture... I created an account just to say thank you for this beautiful movie.

154 out of 210 found this helpful. Was this review helpful?

10/10

I cried twice watching this movie
pramsalim 2 December 2017

I have always been a fan of Pixar movies. I grew up watching them in the bar for quality animated movies. They were made for kids only. They were simple stories with relatable characters. Then came Pixar with its relatable characters and plus, completely 3D animated. And I

Videos



Photos



People who liked this also liked...



Add to Watchlist

Next »

Wonder (I) (2017)

Certificate: PG Drama | Family

8.1/10

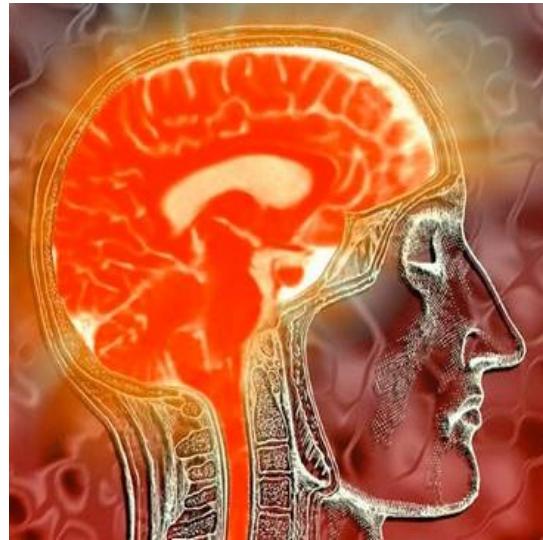
Based on the New York Times bestseller, WONDER tells the incredibly inspiring and heartwarming story of August Pullman, a boy with facial differences who enters fifth grade, attending a mainstream elementary school for the first time.

Director: Stephen Chbosky
Stars: Jacob Tremblay, Owen Wilson...

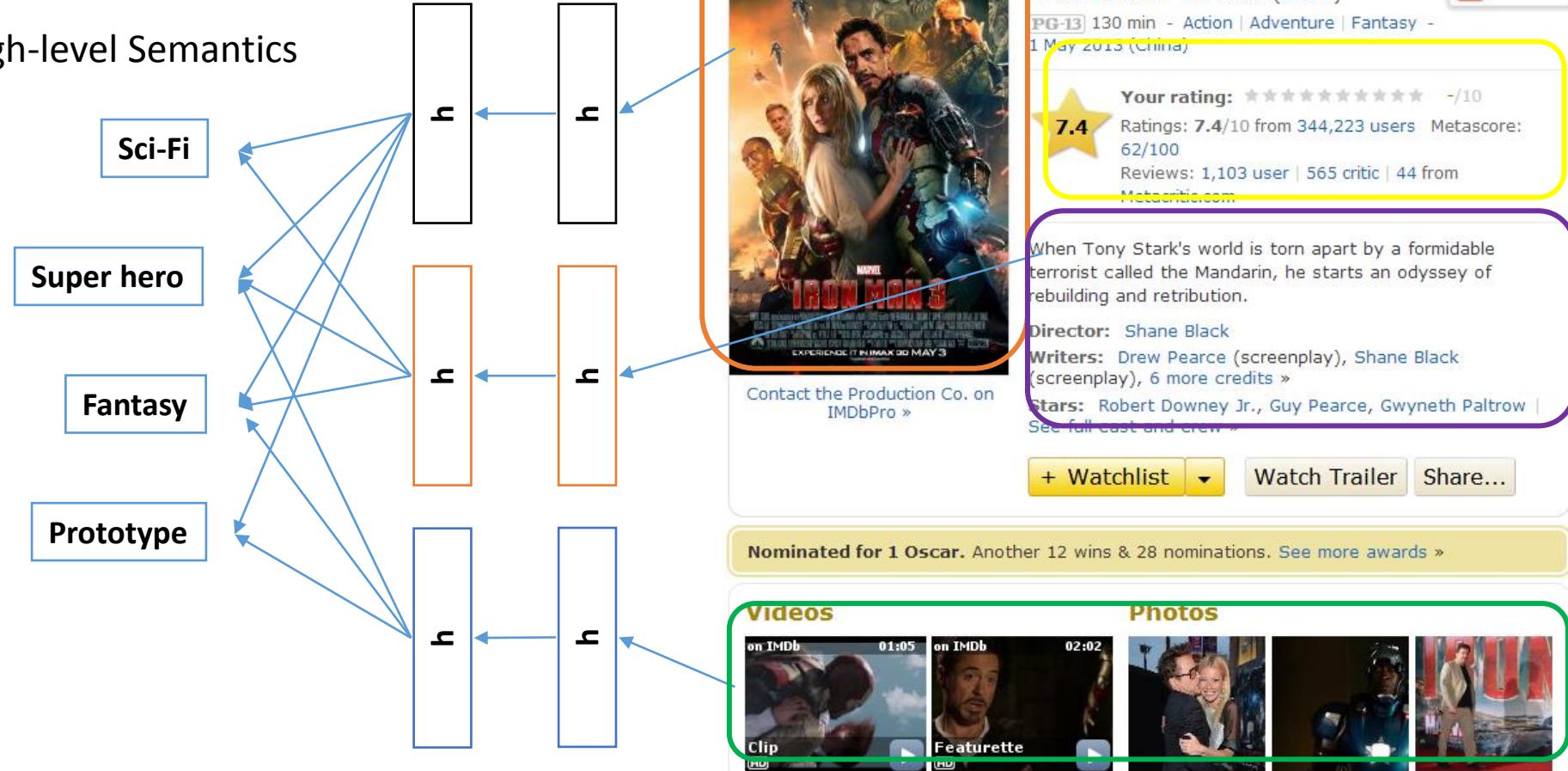
Section II:

- Leveraging Complementary Information
 - Cross-domain Recommender Systems
 - Social Recommender Systems
- Leveraging Comprehensive Information
 - Multimodal Recommender Systems
 - Multi-objective Recommender Systems

Human are Joint Thinking with Related Data



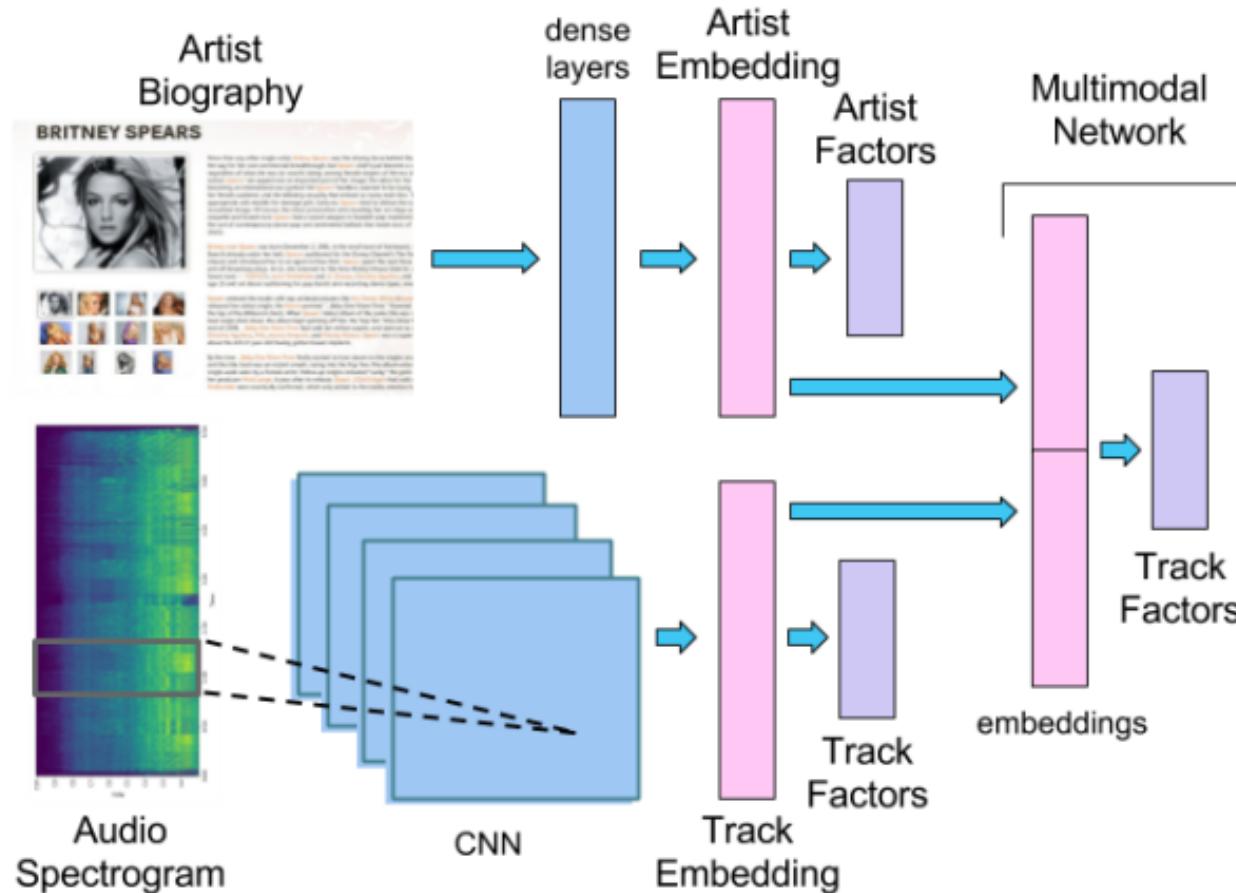
High-level Semantics



Multimodal Learning

- The information in real world usually comes as different modalities. Different modalities are characterized by very different statistical properties. For example,
 - Images are usually associated with tags and text;
 - Texts contain images to more clearly express the main idea of the article.
- **Multimodal learning** aims to learn a joint representation of different modalities.
- **Multimodal learning** is also capable to fill missing modality given the observed ones.

Multimodal Music Recommendation



Datasets

- Million Song Dataset (MSD)
 - <https://labrosa.ee.columbia.edu/millionsong/>
 - Echo Nest Taste Profile Subset provides play counts of 1 million users on more than 380,000 songs from the MSD
 - Biographies and social tags are collected from Last.fm for all the artists that have at least one song in the dataset.
- Final Dataset (MSD-A)
 - <https://zenodo.org/record/831348>
 - The dataset consists of 328,821 tracks from 24,043 artists. Each track has at least 15 seconds of audio, each biography is at least 50 characters long, and each artist has at least 1 tag associated with it.

Results of Artist and Song Recommendation

Table 1: Artist Recommendation Results

Approach	Input	Data model	Arch	MAP
A-TEXT	Bio	VSM	FF	0.0161
A-SEM	Sem Bio	VSM	FF	0.0201
A-W2V-GOO	Bio	w2v-pretrain	CNN	0.0119
A-W2V	Bio	w2v-trained	CNN	0.0145
A-TAGS	Tags	VSM	FF	0.0314
TAGS-ITEMKNN	Tags	-	itemKnn	0.0161
TEXT-RF	Bio	VSM	RF	0.0089
RANDOM	-	-	-	0.0014
UPPER-BOUND	-	-	-	0.5528

Mean average precision (MAP) at 500 for the predictions of artist recommendations in 1M users. VSM refers to Vector Space Model, FF to Feedforward, RF to Random Forest, CNN to Convolutional Neural Network, and itemKnn to itemAttributeKnn approach. Bio refers to biography texts and Sem Bio to semantically enriched texts.

Table 2: Song Recommendation Results

Approach	Artist Input	Track Input	Arch	MAP
AUDIO	-	audio spec	CNN	0.0015
SEM-VSM	Sem Bio	-	FF	0.0032
SEM-EMB	A-SEM	-	FF	0.0034
MM-LF-LIN	A-SEM	AUDIO emb	MLP	0.0036
MM-LF-H1	A-SEM	AUDIO emb	MLP	0.0035
MM	Sem Bio	audio spec	CNN	0.0014
TAGS-VSM	Tags	-	FF	0.0043
TAGS-EMB	A-TAGS	-	FF	0.0049
RANDOM	rnd emb	-	FF	0.0002
UPPER-BOUND	-	-	-	0.1649

Mean average precision (MAP) at 500 for the predictions of song recommendations in 1M users. AUDIO emb refers to the track embedding of AUDIO approach, SEM to artist embedding of SEM approach, TAGS to artist embedding of TAGS approach, spec to spectrogram, mm to multimodal, lf to late fusion, lin to linear, and h1 to one hidden layer.

Transferring Deep Visual Semantic Features to Learning to Rank



“Red Short
dress, Prom
Dress,
wedding
dress, dress,
...”



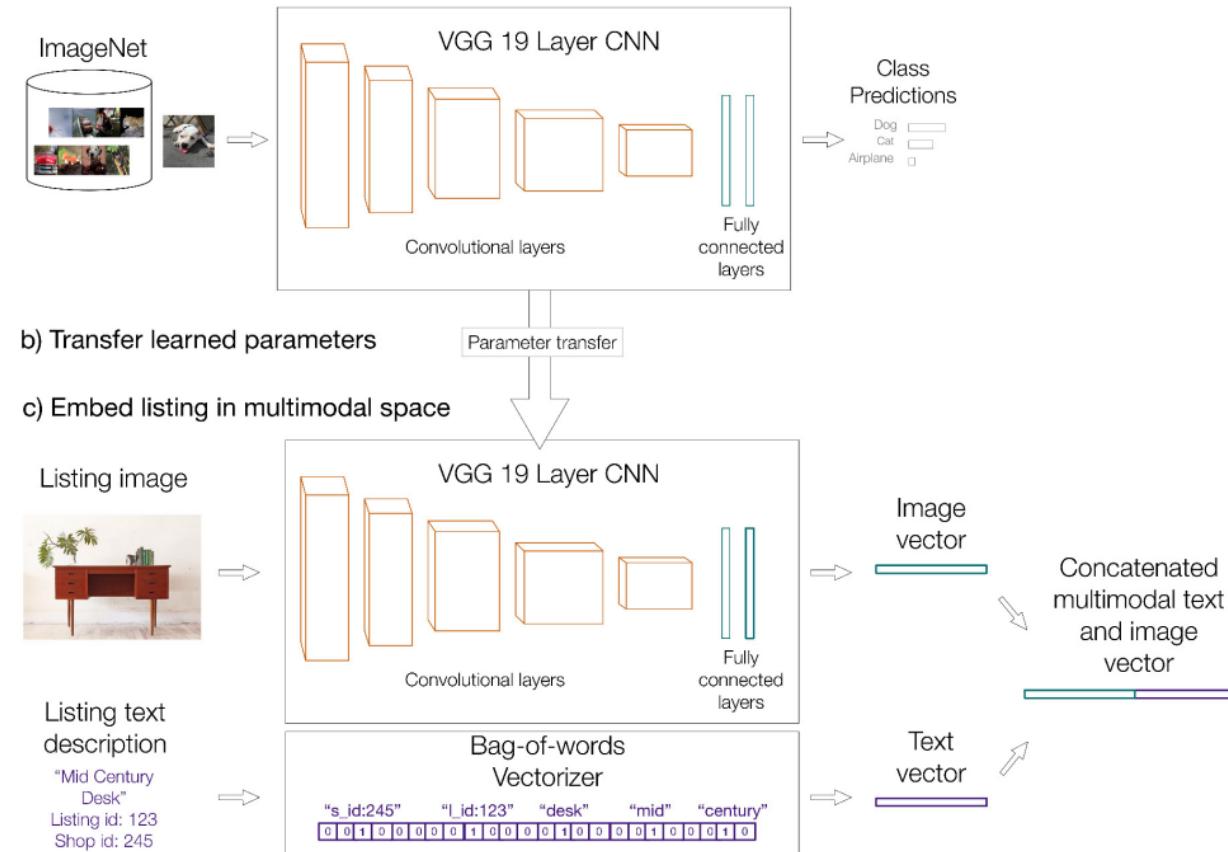
“Pocket Knife
wedding
shower ideas
wedding
dresses,
beach ...”



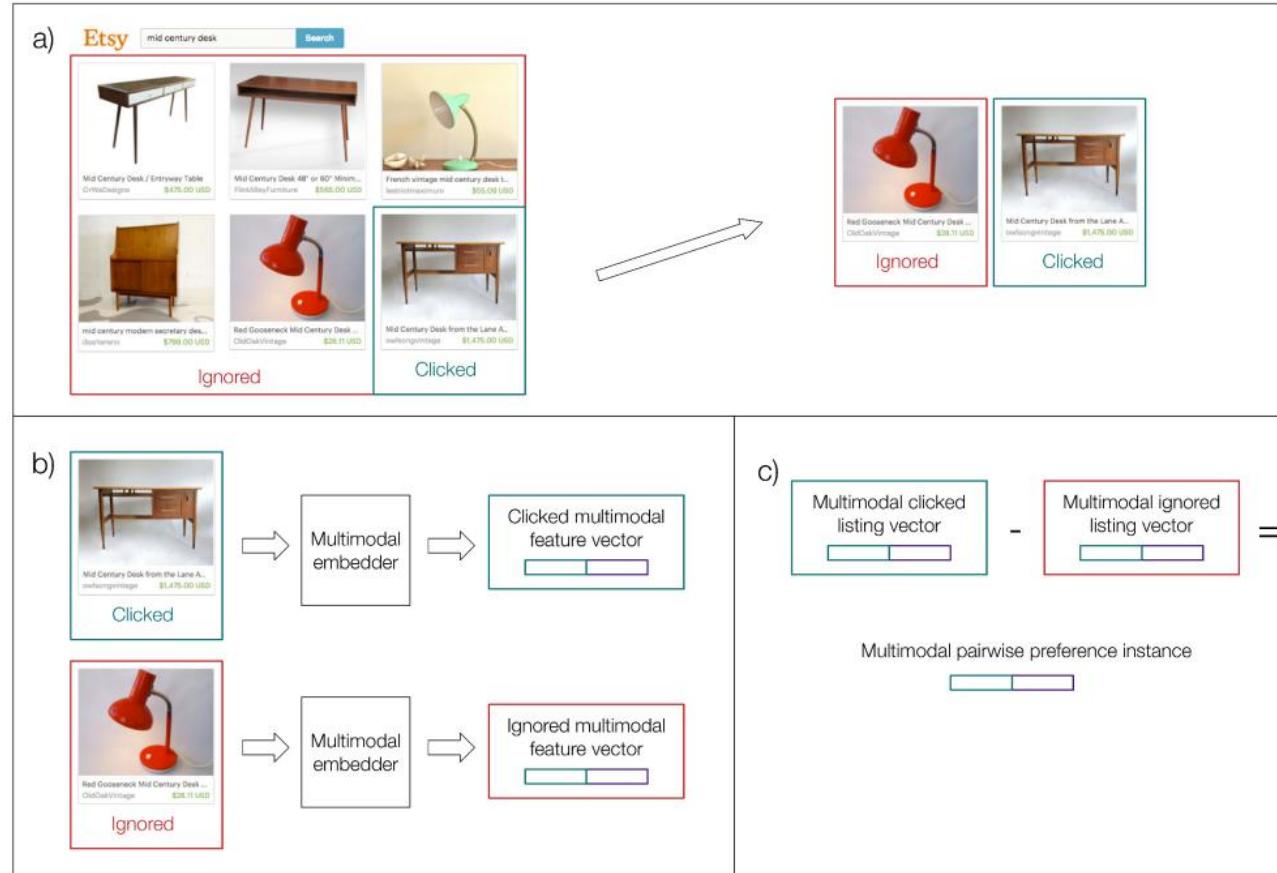
“Yellow
dress. Retro
dress
Wedding
dress. Flared
skirt...”

- Irrelevant search results for the query “wedding dress”
- Even though it’s apparent in the images that these are not wedding dresses

Transferring Parameters of A CNN to The Task of Multimodal Embedding



From search logs to multimodal pairwise classification instances



Datasets

- <https://www.etsy.com/>
- 2 week period in search logs, 1.4 million Etsy listings with images.
- Related dataset:
 - <http://vision.is.tohoku.ac.jp/~kyamagu/research/etsy-dataset/>

Image information can help disentangle different listings considered similar by a text model

a) Query: "**wolf**"



"animal_shirt **animals** **clothing**
hipster_shirt l l_xl m m_l pocket s
s_m shirt tshirt xl **wolf**"

b) Query: "**leather bag**"



"accessories **bag** bag_charm
charm charm_leather key_chain
leather **leather_bag**
leather_bag_charm long_tassel"

c) Query: "**wedding band**"



"**bag** bags bags_and_purses
leather messenger
messenger_bag vintage
vintage_bags vintage_messenger
leather_bag"



"**band** hammered **jewelry**
mens_mens_wedding silver
silver_ring sterling **wedding**
wedding_band **ring**"

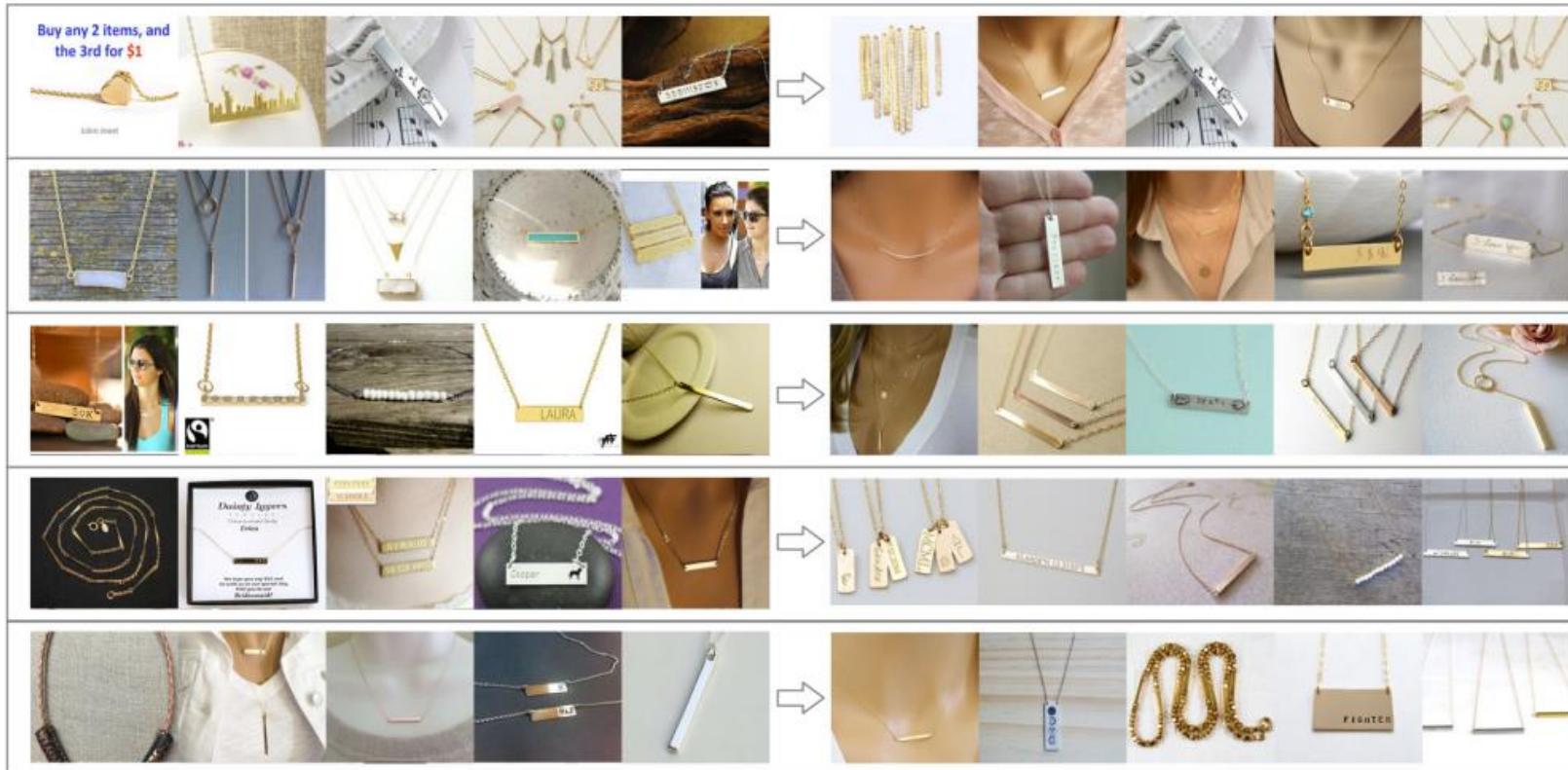
(a)

(b)

(c)

Visualizing ranking changing by incorporating image information

Original ranking for “bar necklace”

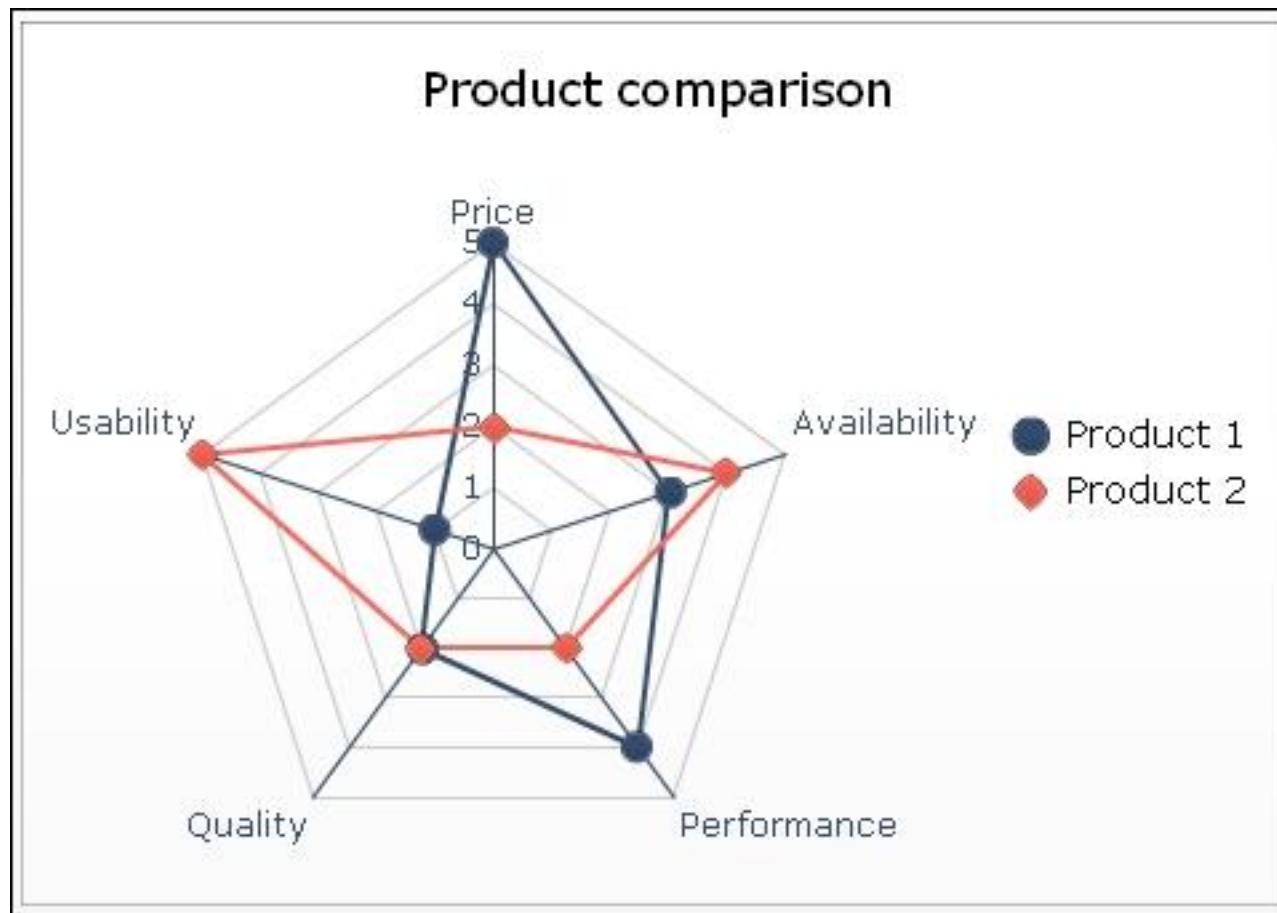


Multimodal ranking for “bar necklace”

Open issues and directions

- Information integration from multiple components
 - How to jointly represent heterogeneous information from multiple components?
 - How to deal with noisy data in each modality?
- Multi-view learning over heterogeneous data sources
 - Extract consistent information from review, description, item images, user videos
- GAN-based models to generate multiple types of samples
 - Text, images, videos

Learning Comprehensive Information



Section II:

- Leveraging Complementary Information
 - Cross-domain Recommender Systems
 - Social Recommender Systems
- Leveraging Comprehensive Information
 - Multimodal Recommender Systems
 - Multi-objective Recommender Systems

Multi-objective Recommender Systems

- Traditional RSs are built on single objective
- However, users' choices are determined by multiple aspects
- To learn users' profile more comprehensively, we need to build new RSs to optimize multiple objectives for each aspect

Long-tail Users/Items for Optimization

- Popularity Bias
 - Short-head users and items account for the majority of data, and models tend to fit these users and items. It overlooks the preference of users and items in the tail.
- Shilling Attack
 - Short-head items are well known by users. However, long-tail items have few data and they are more vulnerable to shilling attack.

RMRM : Joint Optimizing Credibility and Specialty

- Recurrent Mutual Regularization Model (RMRM) consists of two main components
 - C-HMF models user choices by emphasizing credibility
 - S-HMF models user choices by emphasizing specialty
- Each component leads to a different objective for optimization, so RMRM is a multi-objective recommenders systems

Classic Probabilistic MF & Heteroscedastic MF

- $P(\mathbf{U}_i) = N(\mathbf{U}_i | \mathbf{0}, \sigma_U^2 \mathbf{I})$
- $P(\mathbf{V}_j) = N(\mathbf{V}_j | \mathbf{0}, \sigma_V^2 \mathbf{I})$
- $P(Y_{ij} | \mathbf{U}_i, \mathbf{V}_j) = N(Y_{ij} | \mathbf{U}_i^\top \mathbf{V}_j, \sigma^2)$



- $P(\mathbf{U}_i) = N(\mathbf{U}_i | \boldsymbol{\mu}_{\mathbf{U}}, \sigma_U^2 \mathbf{I})$
- $P(\mathbf{V}_j) = N(\mathbf{V}_j | \boldsymbol{\mu}_{\mathbf{V}}, \sigma_V^2 \mathbf{I})$
- $P(Y_{ij} | \mathbf{U}_i, \mathbf{V}_j) = N(Y_{ij} | \mathbf{U}_i^\top \mathbf{V}_j, \sigma_{ij}^2)$

$$P(\mathbf{U}, \mathbf{V} | Y) \propto P(Y, \mathbf{U}, \mathbf{V}) = \prod_{ij \in \mathcal{O}} P(Y_{ij} | \mathbf{U}_i, \mathbf{V}_j) \prod_i P(\mathbf{U}_i) \prod_j P(\mathbf{V}_j)$$

- *Loss function:*

$$\underset{\mathbf{U}, \mathbf{V}}{\operatorname{argmin}} \left[\underbrace{\sum_{ij} (Y_{ij} - \mathbf{U}_i^\top \mathbf{V}_j)^2}_{\text{weighted loss}} + \underbrace{\lambda_U \sum_i \|\mathbf{U}_i\|^2 + \lambda_V \sum_j \|\mathbf{V}_j\|^2}_{\text{regularization}} \right]$$



- *Loss function:*

$$\begin{aligned} & \underset{\mathbf{U}, \mathbf{V}}{\operatorname{argmin}} \left[\underbrace{\sum_{ij} \mathbf{w}_{ij} (Y_{ij} - \mathbf{U}_i^\top \mathbf{V}_j)^2}_{\text{weighted loss}} + \underbrace{\lambda_U \sum_i \|\mathbf{U}_i - \boldsymbol{\mu}_{\mathbf{U}}\|^2 + \lambda_V \sum_j \|\mathbf{V}_j - \boldsymbol{\mu}_{\mathbf{V}}\|^2}_{\text{regularization}} \right] \end{aligned}$$

Popularity Bias

Shilling Attack

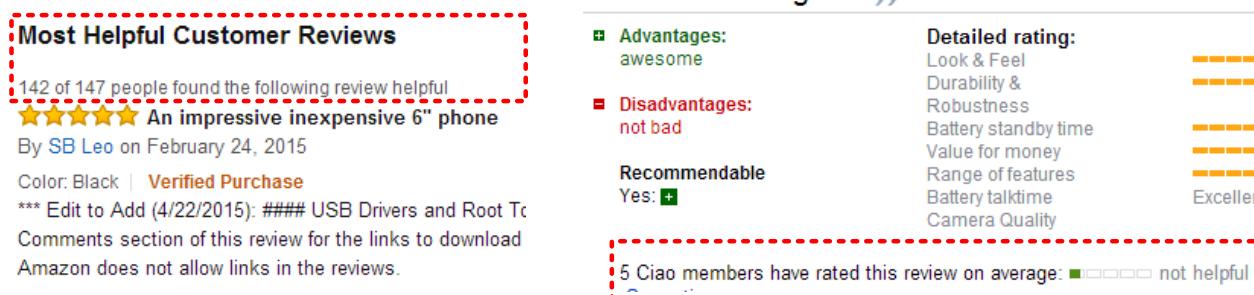
- model variance, i.e. weight on the loss : $w_{ij} = f(\sigma_{ij}^{-2})$

Credibility Enhancement

- C-HMF (Credibility-specific Heteroscedastic MF)
 - $\sigma_{ij}^2 = f^C(Y_{ij}) \propto \varphi_i^{-1}$ scores the *credibility* of each review
- Bayesian Reputation Modeling
 - *Reputation Score*: Given the helpfulness scores h_i of a user i , the reputation score on this user is defined by:

Shilling Attack

$$\varphi_i = \mathcal{R}(e_i | h_i) \stackrel{\text{def}}{=} \frac{r + \alpha}{r + s + \alpha + \beta}$$



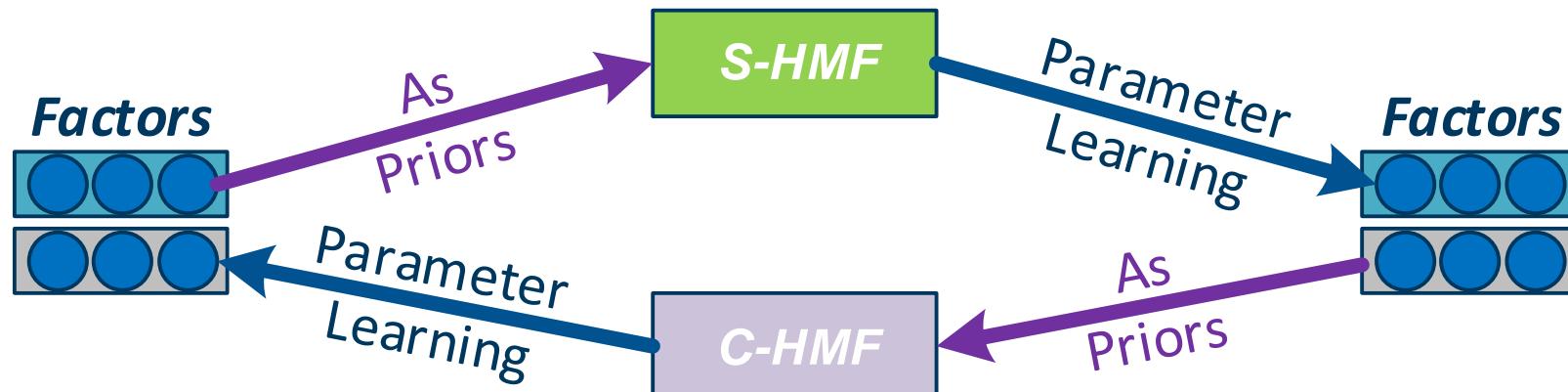
Specialty Enhancement

- S-HMF (Specialty-specific Heteroscedastic MF)
 - $\sigma_{ij}^2 = f^S(Y_{ij}) \propto \psi_j^{-1}$ scores the *specialty* of user choice, which tightly fits the choices over long-tail items
- Given all observed choices, the *specialty score* of a choice on an item j is measured by the *self-information*:
 - $\psi_j = -\log \bar{p}(j|\alpha)$

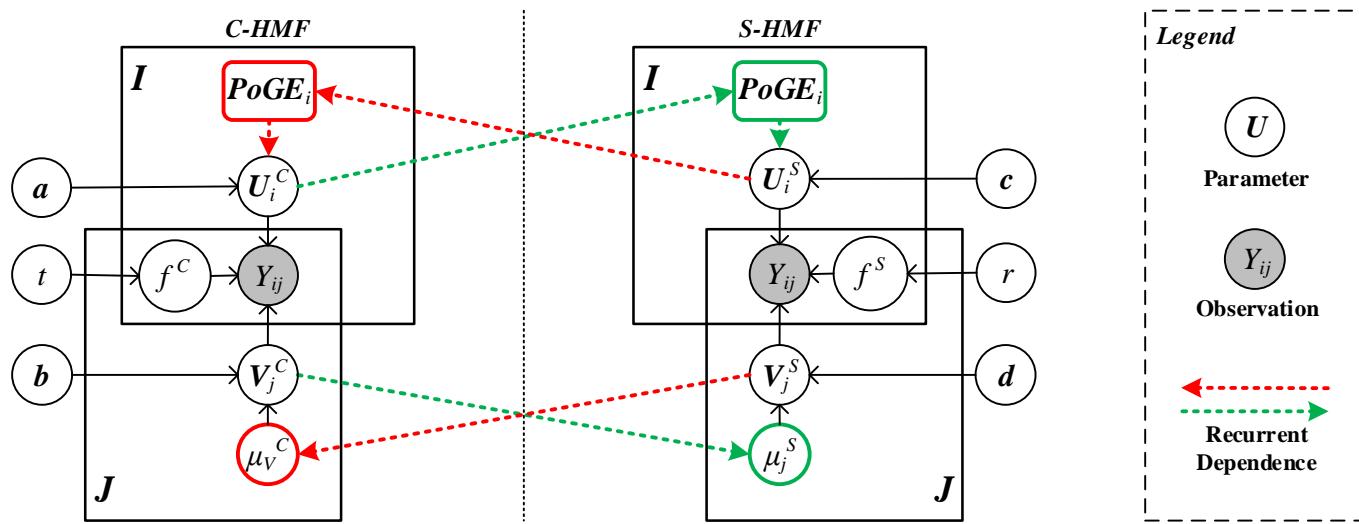
Popularity Bias

Recurrent Mutual Regularization

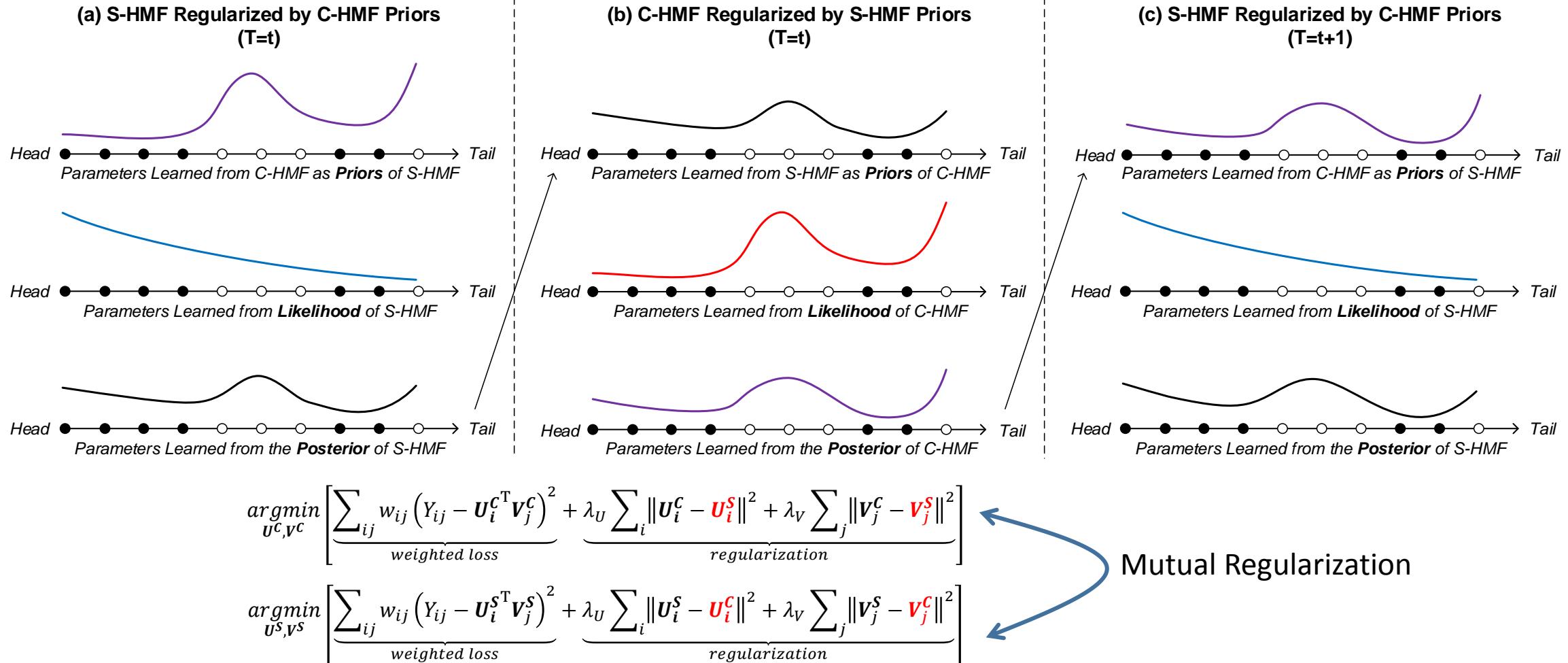
- A **recurrent mutual regularization process** couples S-HMF and C-HMF using the user and items factors learned from each other as the **empirical priors**



Graphical model of RMRM framework

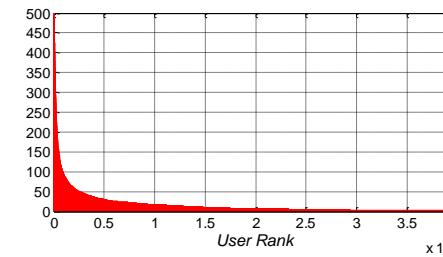
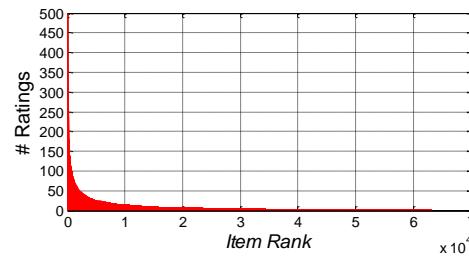


Demonstration of the recurrent mutual Regularization process

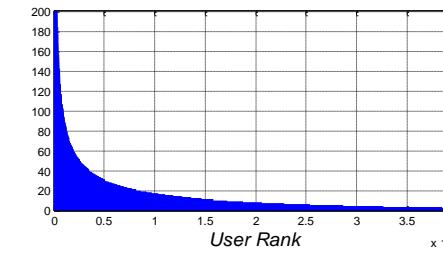
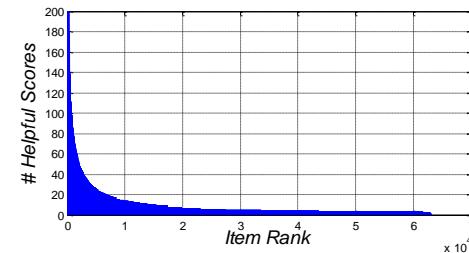


Dataset: the Epinions

# users: 39,902	# items: 63,027
# trust links: 43,8965	# trusters / users: 11
max # of trusters: 1,713	# users with zero truster: 14,202
# ratings: 734,441	density: 0.029%
# ratings / users: 18	# ratings / items: 11
max # ratings of user: 1,809	max # ratings of item: 2,112

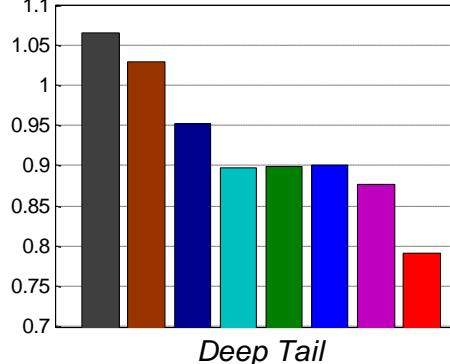
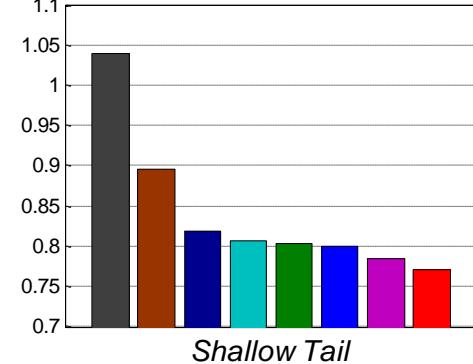
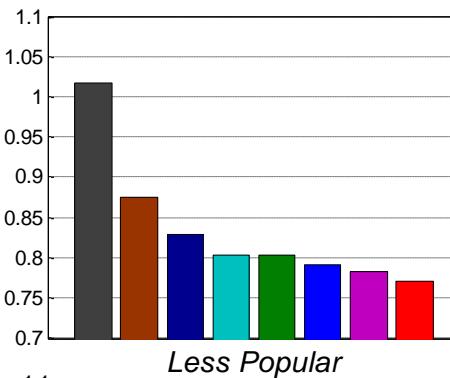
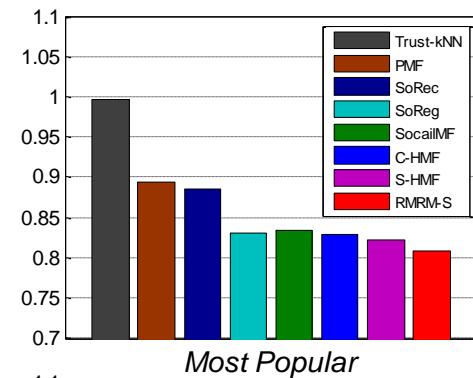


Long-tail distributions for the number of ratings of items and users (truncated from 0 to 500)

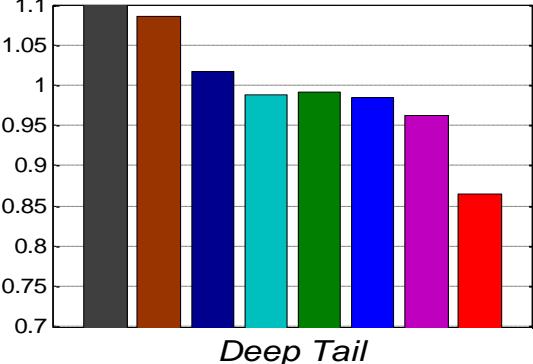
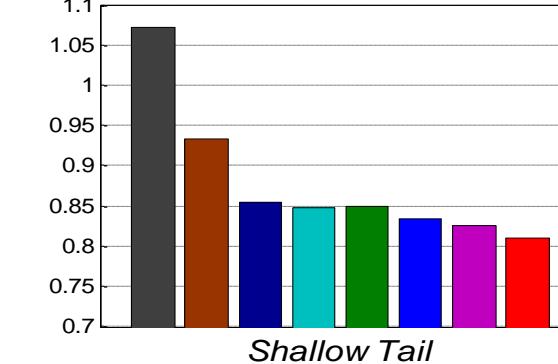
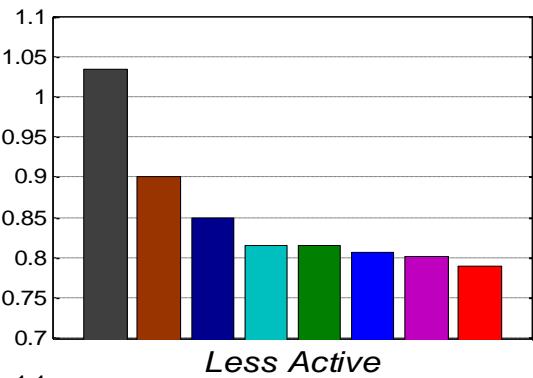
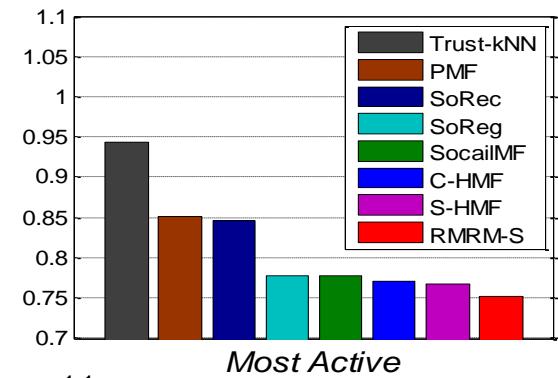


The distributions for the number of helpful scores w.r.t. items and users (truncated from 0 to 200)

Rating Prediction on Long-tail Distributed Items and Users



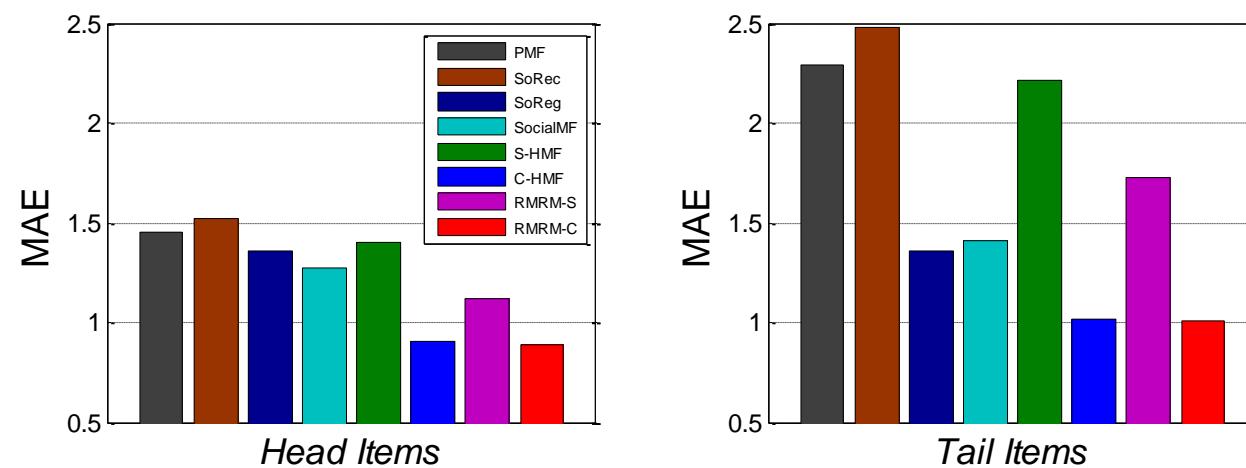
MAEs of rating prediction for the long-tail item distribution



MAEs of rating prediction for the long-tail user distribution

Shilling Attack Simulation

- To simulate such an environment
 - We created 1,000 virtual spam users to conduct the attack
 - We selected 100 items from the head (0%~20%) and the tail (20%~100%) as the attack targets.
- Nuke attack in the case of the average attack model



Open issues and directions

- How to integrate the impacts from multiple objectives?
 - Different users may pay attention on different objectives
 - The importance of objectives are often dependent on the context
- Modeling with game theory to find equilibria over multiple objectives
- Applying multi-objective optimization methods in RS
 - Multiple-criteria decision analysis
 - Multidisciplinary design optimization

Section III:

- Leveraging Contextual Information
 - Context Modeling in Recommender Systems
 - Session-based Recommender Systems
 - Group-based Recommender Systems
- Real Recommender Systems in Practice

What is context?

- There are many definitions of context across various disciplines and even within specific subfields of these disciplines.
- The **representational view** assumes that the contextual attributes are identifiable and known a priori and, hence, can be captured and used within the context-aware applications.
- The **interactional view** assumes that the user behavior is induced by an underlying context, but that the context itself is not necessarily observable.

In a short word

- Context is any factor (observable or not observable) leading to user behavior

Section III:

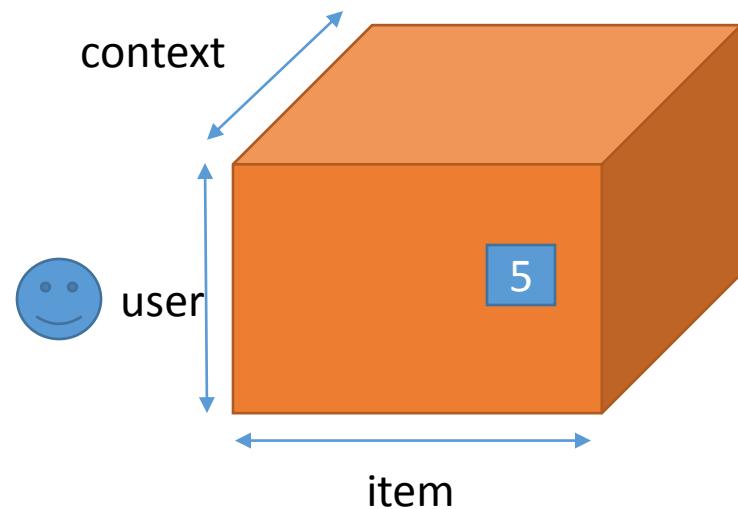
- Leveraging Contextual Information
 - Context Modeling in Recommender Systems
 - Session-based Recommender Systems
 - Group-based Recommender Systems
- Real Recommender Systems in Practice

Context-aware Recommender Systems

- Rating mapping without context
 - $User \times Item \rightarrow R$
- Rating mapping with context
 - $User \times Item \times C_1 \times C_2 \times \dots \rightarrow R$

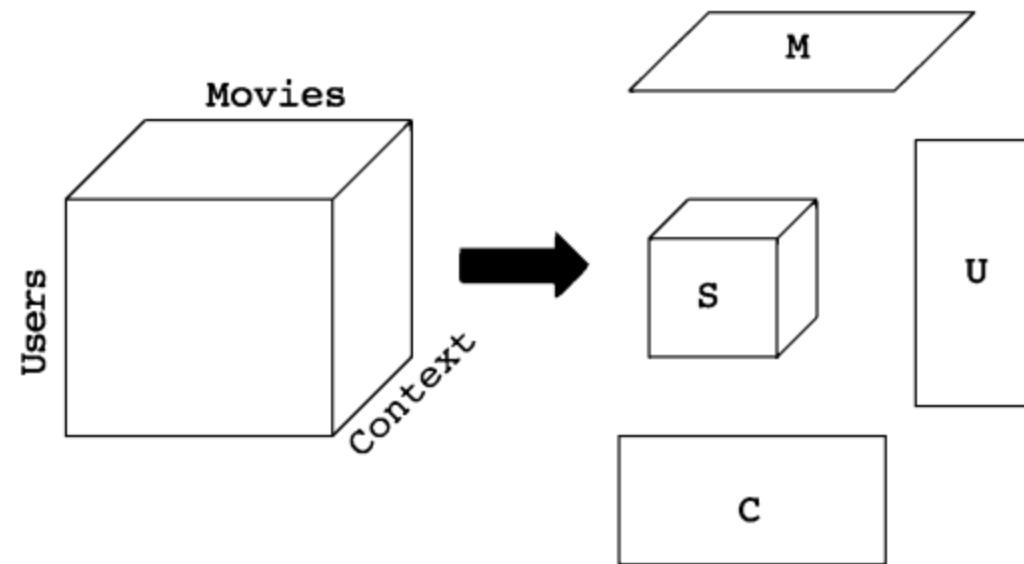
Represent context in higher dimensions

- Rating mapping
 - $R(u, i, c) = 5$



Tensor factorization model

- 3-dimensional tensor over <User, Movie, Context>



Fast context-aware recommendations with factorization machines

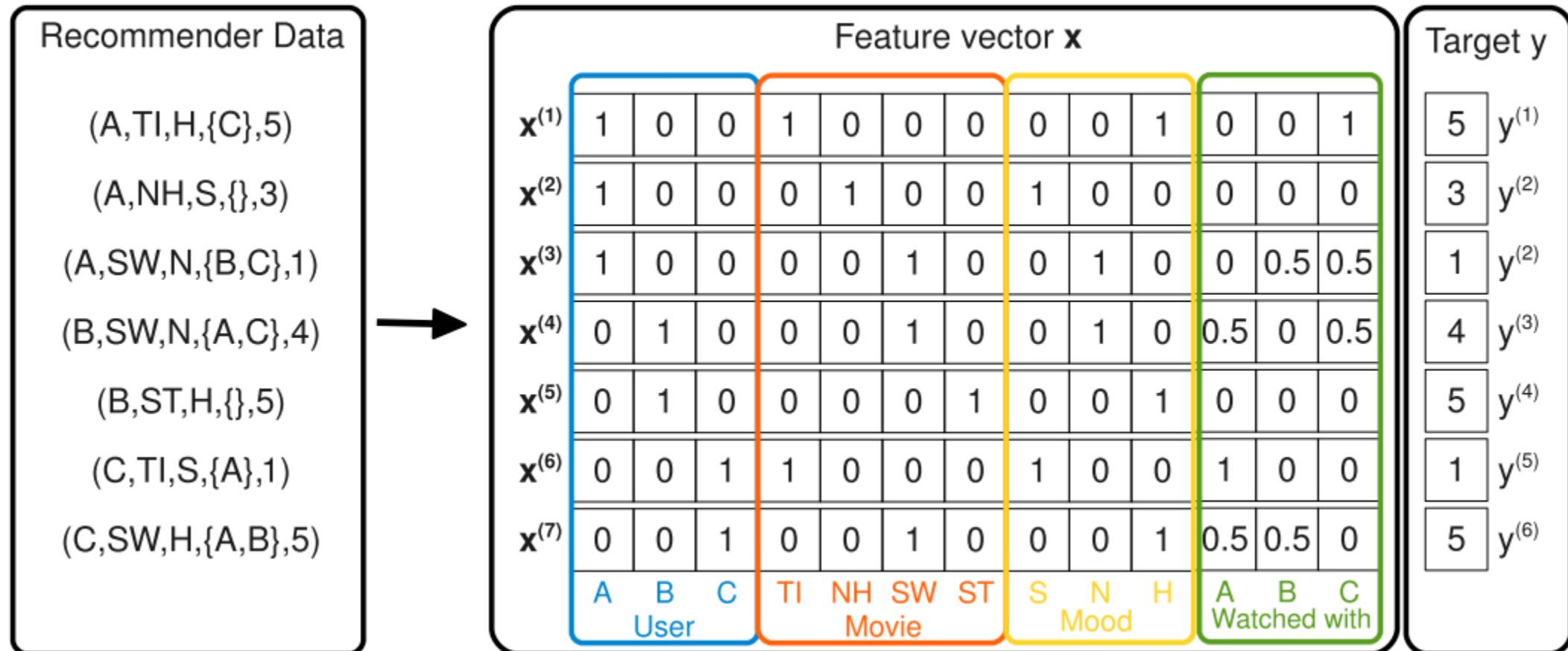
- The idea behind FMs is to model interactions between features using factorized parameters. The FM model has the ability to estimate all interactions between features even with extreme sparse data.
- FM models all interactions between pairs of variables with the target (2nd-order), including nested ones (1st-order), by using factorized interaction parameters

$$\hat{y}(\mathbf{x}) := w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j=i+1}^n \hat{w}_{i,j} x_i x_j$$

where $\hat{w}_{i,j}$ are the factorized interaction parameters between pairs:

$$\hat{w}_{i,j} := \langle \mathbf{v}_i, \mathbf{v}_j \rangle = \sum_{f=1}^k v_{i,f} \cdot v_{j,f}$$

Representing context data as features

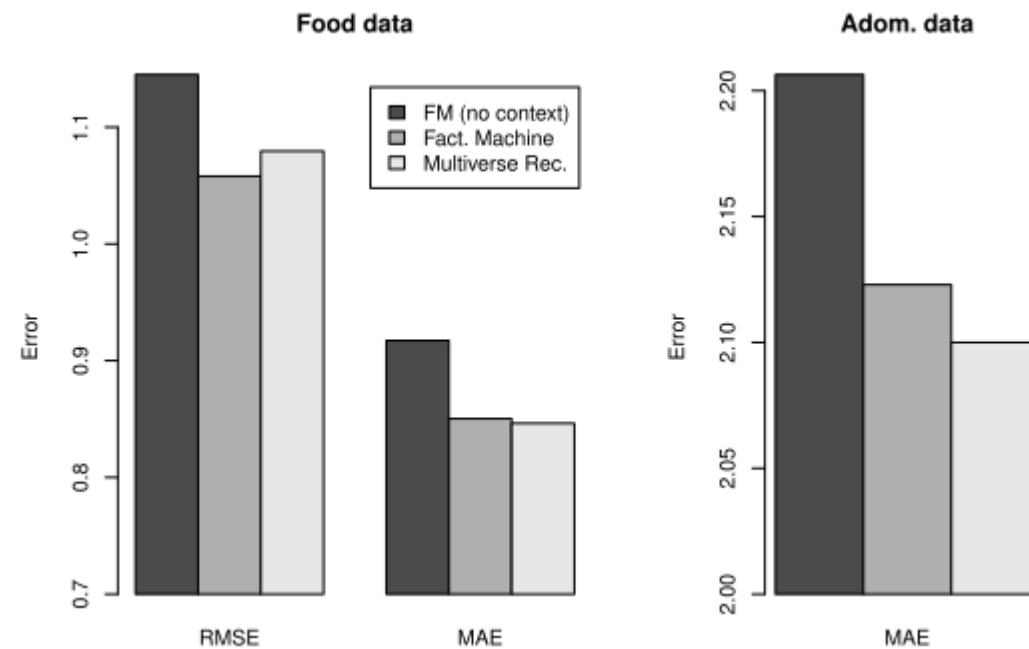


Here in the feature vector \mathbf{x} , the first three values indicate the user, the next four ones the movie, the next three ones the mood and the last three ones the other users a movie has been watched with.

Datasets

- Adom dataset
 - 1524 rating events (1 to 15 stars) for movies with five context variables about companion, the weekday and other time information
- Food dataset
 - 6360 ratings (1 to 5 stars) by 212 users for 20 menu items where one context variable captures if the situation in which the user rates is virtual or real (i.e. if (s)he imagines to be hungry or (s)he really is) and the second one how hungry the user is.

With/without context

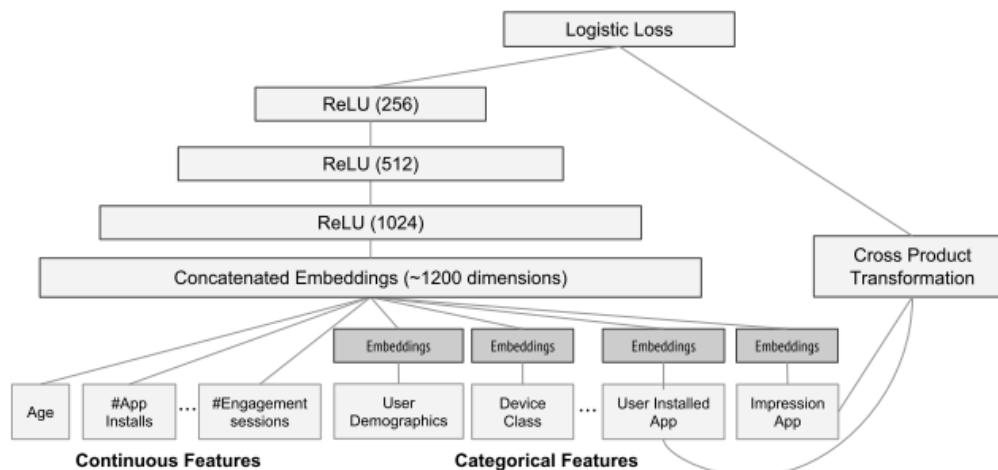
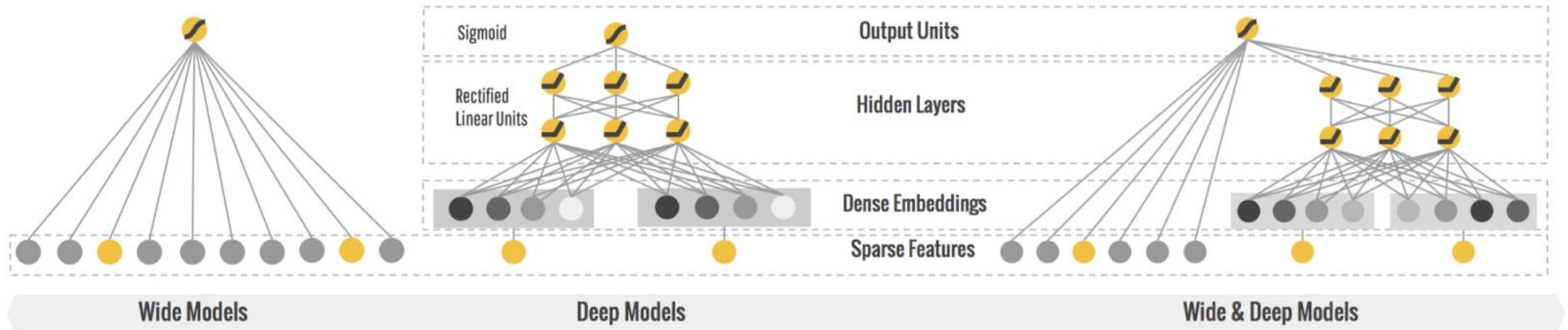


The context-aware methods Multiverse Recommendation and context-aware Factorization Machine benefit from incorporating the context-information into the rating prediction.

Open issues and direction

- Even though the problem of context-aware rating prediction is highly relevant in practice, there are only a few publicly available datasets.
- Finding efficient way to capture the coupling between all context features.
- Modeling high-dimensional context features with deep models.

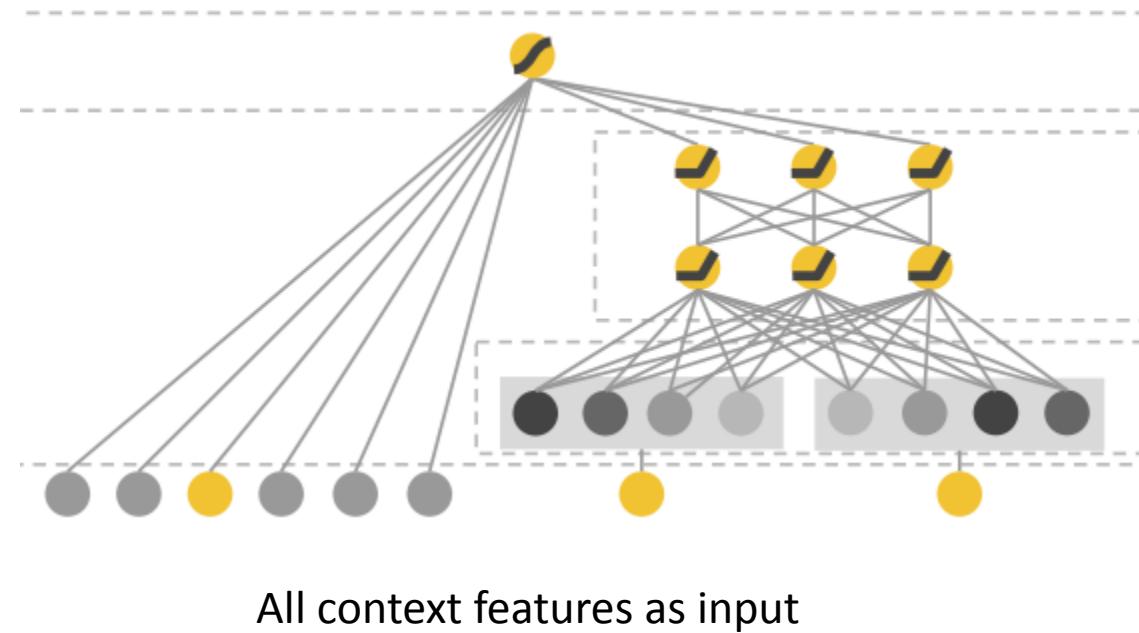
Wide & Deep Learning for RS



Wide & Deep model structure for apps recommendation

Wide & Deep Learning with Context Features

- Just feed all context features into the networks

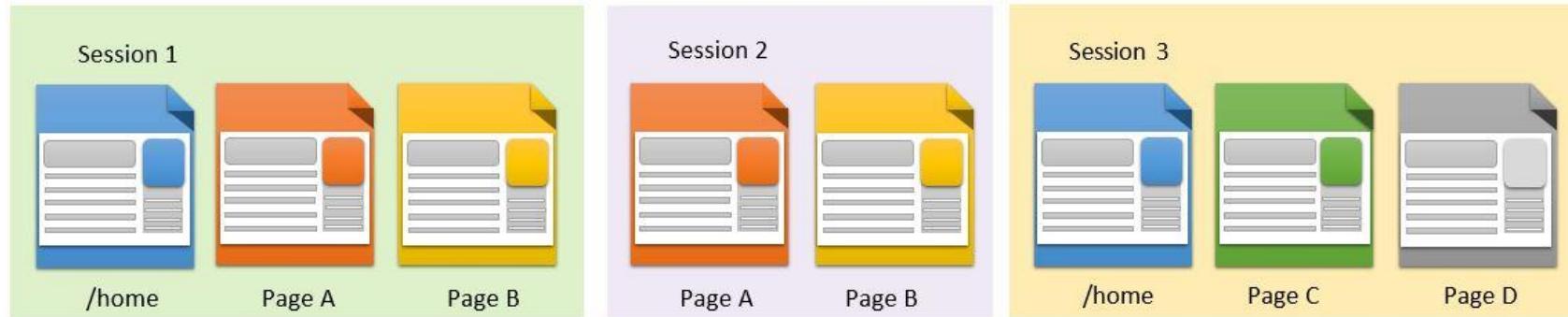


Section III:

- Leveraging Contextual Information
 - Context Modeling in Recommender Systems
 - **Session-based Recommender Systems**
 - Group-based Recommender Systems
- Real Recommender Systems in Practice

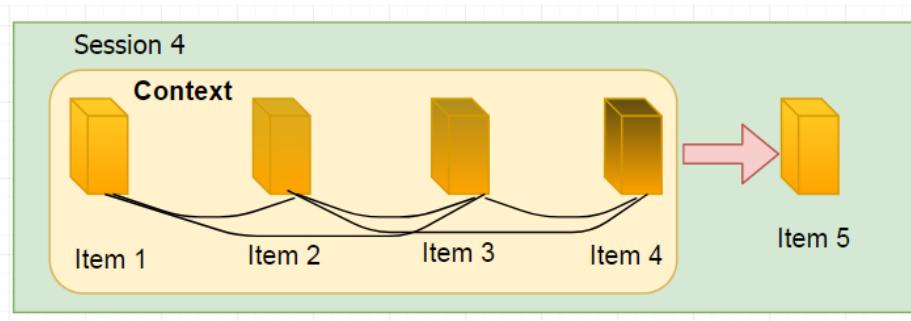
What is a session?

- A session is a sequence of items, actions or events with a bound.
- Strong dependency exists between the objects within a session.



Session context

- Session context consists of observed sequence that leads to the consequent events.
 - e.g., clicked pages in browsing history, chosen items in a transaction.

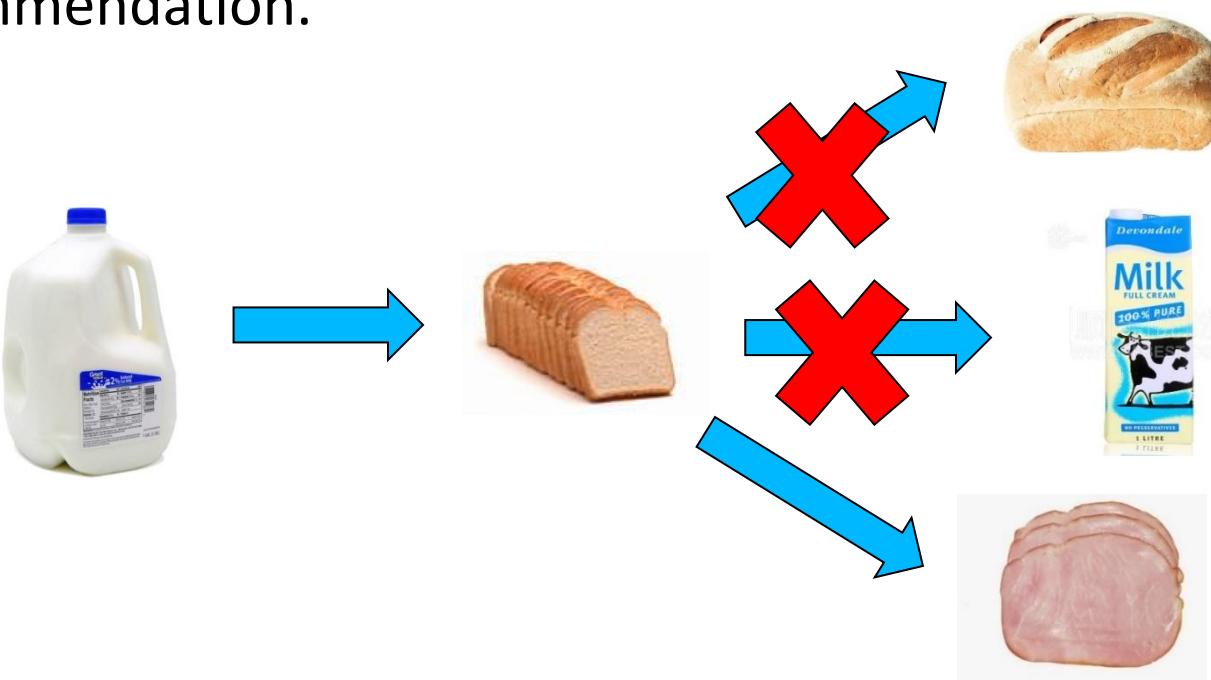


Modeling session

- Deficiencies of non-session based RS
 - Recommender systems built on historical profile are often **repeatedly recommended similar items**.
 - E.g. neighborhood-based methods, matrix factorization methods
 - In most real-world scenarios, we prefer to find items that are **related** to our recent context instead of only **similar** with items that have been bought.
- A system makes more sensible and relevant recommendations if the **session context** was taken into consideration.

Diversifying recommendations

- Users prefer **more diversified options** than those they have had.
 - It is unlikely that a consumer will purchase another loaf of bread if they have purchased one, whereas butter or ham may be a more appealing recommendation.

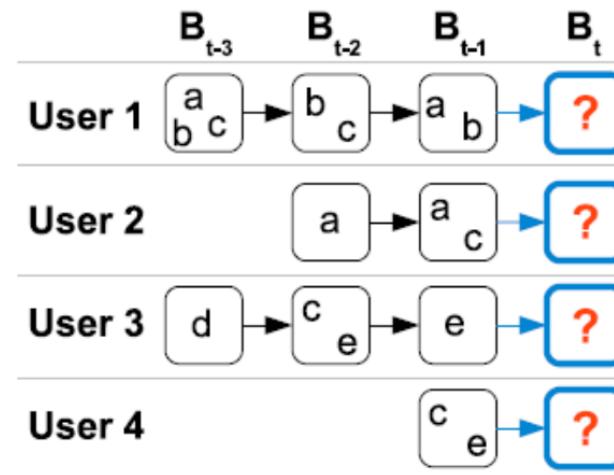


Session-based context modeling

- First-order dependency modeling
- Higher-order dependency modeling
- Loosely ordered dependency modeling

Markov chain-based matrix factorization for next-basket recommendation

- Sequential shopping basket data is given per user
- To recommend the items the user may buy in his next visit



Factorize transition tensor

- A generalization of Matrix Factorization (MF) and Markov Chain (MC) models
- The FM models the *pairwise* interaction in *<user u, item i, item l>*:

$$\hat{a}_{u,l,i} := \langle v_u^{U,I}, v_i^{I,U} \rangle + \langle v_i^{I,L}, v_l^{L,I} \rangle + \langle v_u^{U,L}, v_l^{L,U} \rangle$$

- For each interaction mode, the pair of factorization matrices are :

$$U - I : V^{U,I} \in \mathbb{R}^{|U|*k_{U,I}}, V^{I,U} \in \mathbb{R}^{|I|*k_{U,I}}$$

$$I - L : V^{I,L} \in \mathbb{R}^{|I|*k_{I,L}}, V^{L,I} \in \mathbb{R}^{|I|*k_{I,L}}$$

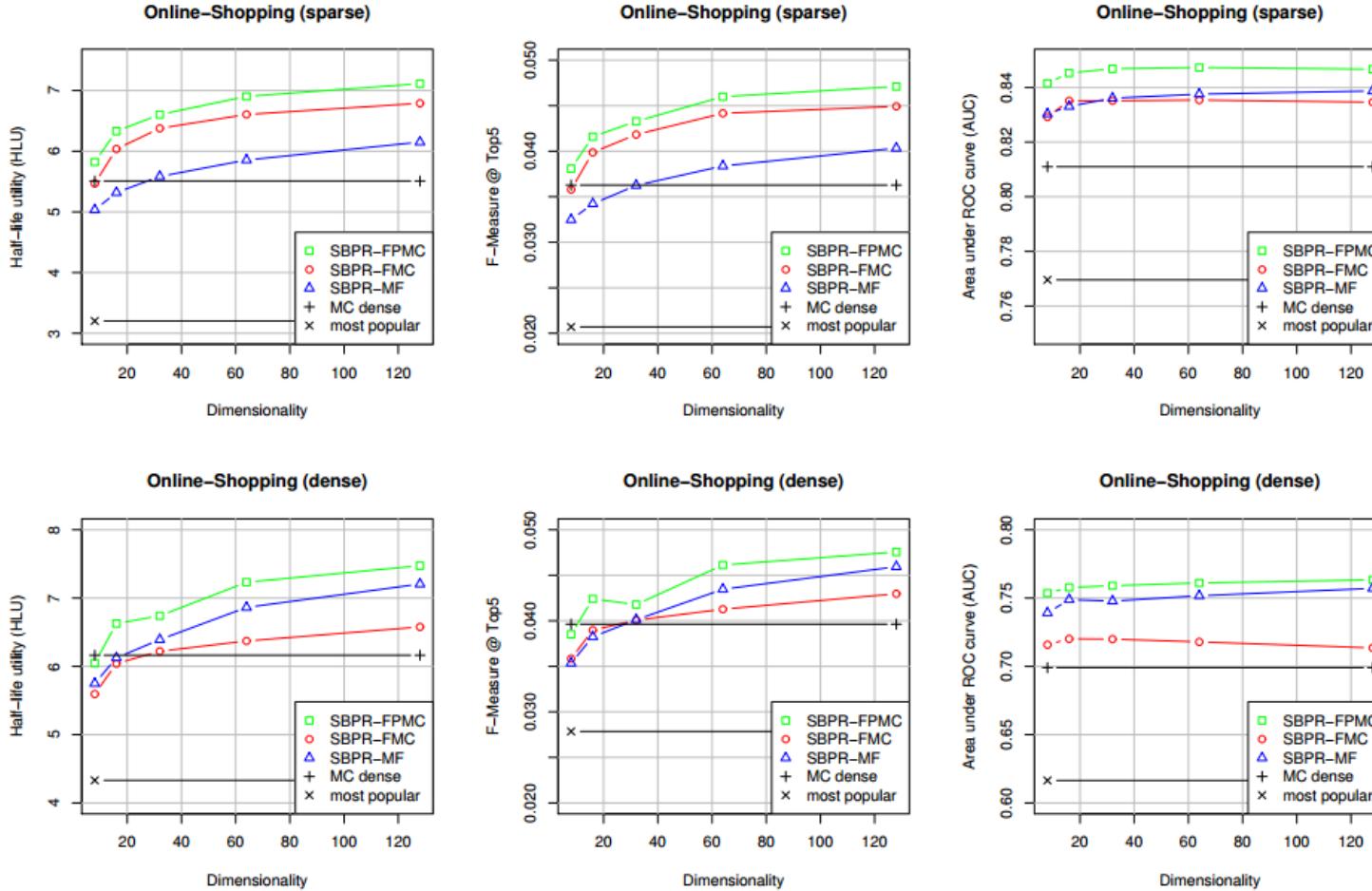
$$U - L : V^{U,L} \in \mathbb{R}^{|U|*k_{U,L}}, V^{L,U} \in \mathbb{R}^{|I|*k_{U,L}}$$

Experiment datasets

- The evaluation is performed on a anonymized purchase data of online drug store. <http://www.rosmannversand.de>
- The dataset is 10-core subset, i.e. every user bought at least 10 items and vice versa each item was bought by 10 users.

dataset	users $ U $	items $ I $	baskets	avg. basket size	avg. baskets per user	triples
Drug store 10-core (sparse)	71,602	7,180	233,476	11.3	3.2	2,635,125
Drug store (dense)	10,000	1,002	90,655	9.2	9.0	831,442

Experimental results



- Factorized personalized Markov chains (FPMC)
- Factorized Markov chain (FMC)
- Matrix factorization (MF)
- A standard dense Markov chain (MC dense)
- baseline ‘most-popular’

Session-based context modeling

- First-order dependency modeling
- Higher-order dependency modeling
- Loosely ordered dependency modeling

Gru4Rec network architecture

- By modeling the whole session, more accurate recommendations can be provided.
- RNN-based approach to model the session and provide session-based recommendations.

Gated Recurrent Unit

Hidden state is the mix of the previous hidden state and the current hidden state candidate (controlled by the update gate):

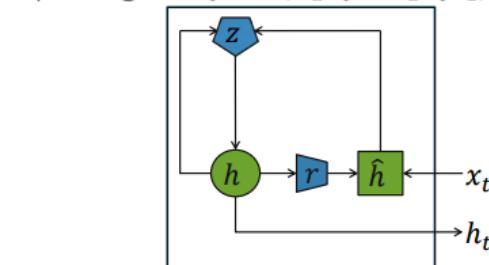
$$h_t = (1 - z_t)h_{t-1} + z_t \hat{h}_t$$

The reset gate controls the contribution of the previous hidden state to the hidden state candidate:

$$\hat{h}_t = \tanh(Wx_t + U(r_t \circ h_{t-1}))$$

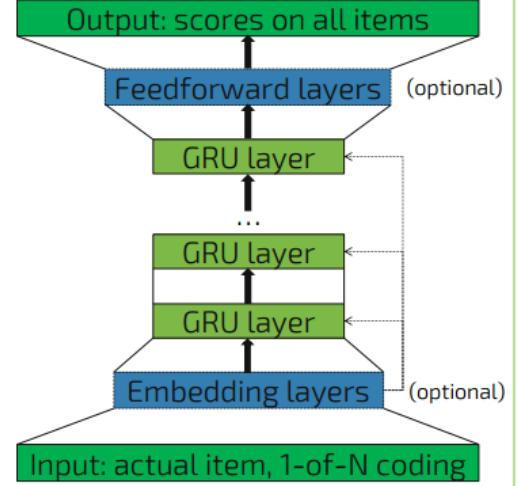
$$\text{Reset gate: } r_t = \sigma(W_r x_t + U_r h_{t-1})$$

$$\text{Update gate: } z_t = \sigma(W_z x_t + U_z h_{t-1})$$



Architecture

- Input: item of the actual event
- Output: likelihood for every item for being the next one in the event stream



Motivation and key idea in this paper

- Applying GRU-RNN to model session.
- Treating the clicks on items as a sequence.
- Modeling the transition between items with GRU.

Dataset

- RecSys Challenge 2015:
 - <http://2015.recsyschallenge.com/>.
 - This dataset contains click-streams of an ecommerce site.

Experiments

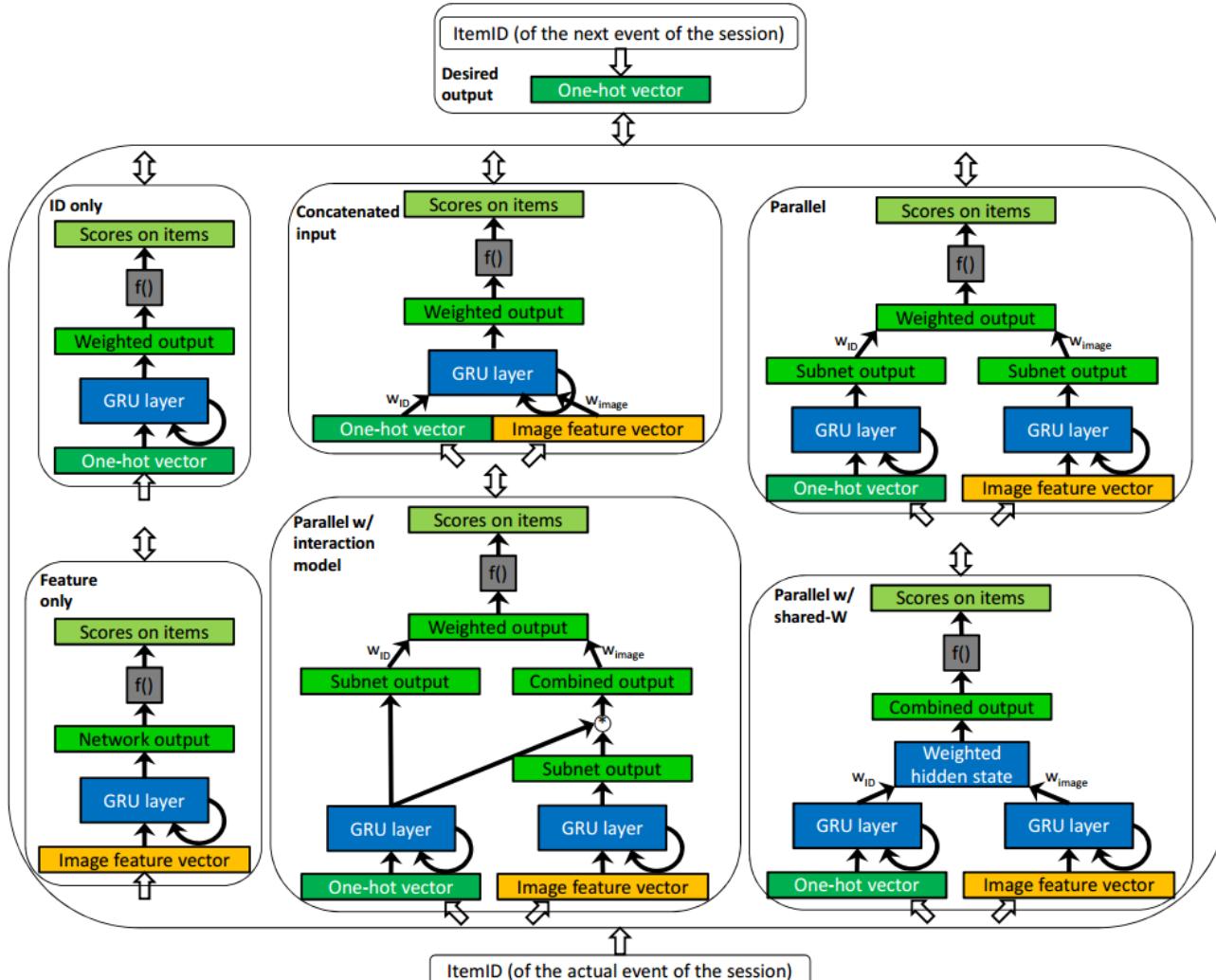
Table 1: Recall@20 and MRR@20 using the baseline methods

Baseline	RSC15		VIDEO	
	Recall@20	MRR@20	Recall@20	MRR@20
POP	0.0050	0.0012	0.0499	0.0117
S-POP	0.2672	0.1775	0.1301	0.0863
Item-KNN	0.5065	0.2048	0.5508	0.3381
BPR-MF	0.2574	0.0618	0.0692	0.0374

Table 3: Recall@20 and MRR@20 for different types of a single layer of GRU, compared to the best baseline (item-KNN). Best results per dataset are highlighted.

Loss / #Units	RSC15		VIDEO	
	Recall@20	MRR@20	Recall@20	MRR@20
TOP1 100	0.5853 (+15.55%)	0.2305 (+12.58%)	0.6141 (+11.50%)	0.3511 (+3.84%)
BPR 100	0.6069 (+19.82%)	0.2407 (+17.54%)	0.5999 (+8.92%)	0.3260 (-3.56%)
Cross-entropy 100	0.6074 (+19.91%)	0.2430 (+18.65%)	0.6372 (+15.69%)	0.3720 (+10.04%)
TOP1 1000	0.6206 (+22.53%)	0.2693 (+31.49%)	0.6624 (+20.27%)	0.3891 (+15.08%)
BPR 1000	0.6322 (+24.82%)	0.2467 (+20.47%)	0.6311 (+14.58%)	0.3136 (-7.23%)
Cross-entropy 1000	0.5777 (+14.06%)	0.2153 (+5.16%)	-	-

Parallel Recurrent Neural Network Architectures for Feature-rich Session-based Recommendations



- **Top row** (left to right): ID only; Concatenated input; Parallel.
- **Bottom row** (left to right): Feature only; Parallel with interaction model; Parallel with shared weight matrix.

Motivation and key idea of this paper

- Incorporate **item features** (e.g., text, image) into RNN-based session models.
- Introduce a number of **parallel RNN** (p-RNN) architectures to model sessions and item features at the same time.
- Propose alternative **training strategies** better than standard training.

Datasets

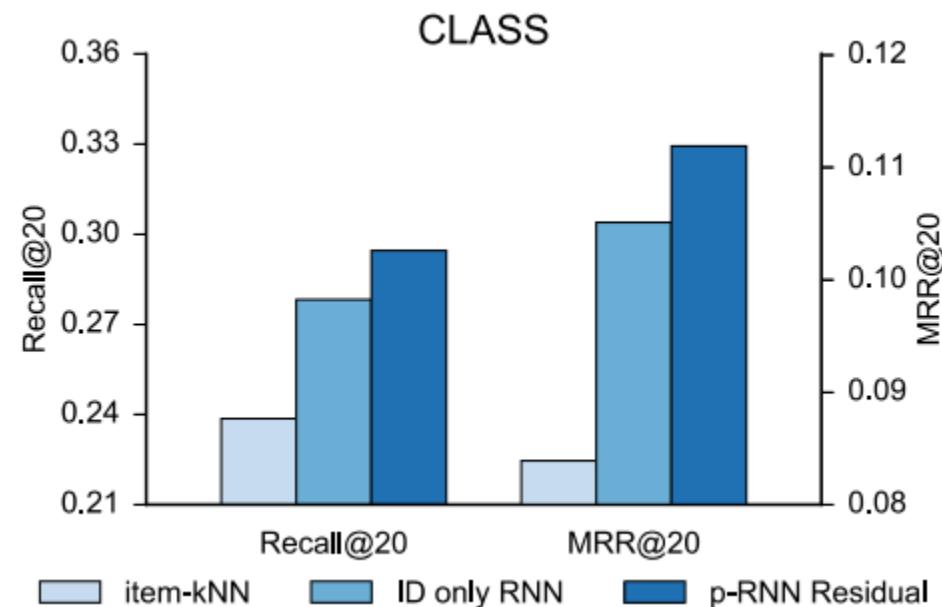
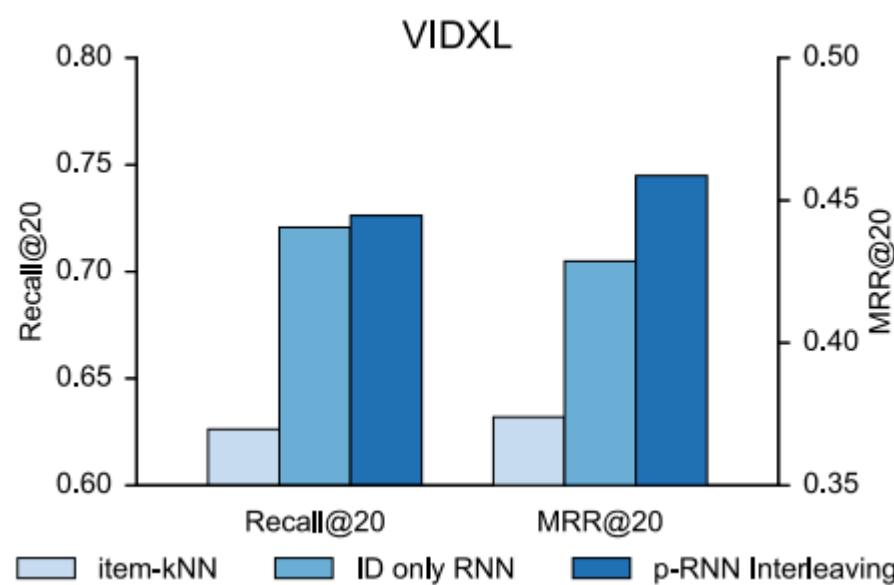
- Two datasets
 - Coined VIDXL : it was collected over a 2-month period from a Youtube-like video site, and contains video watching events having at least a predefined length
 - Class: it consists of product view events of an online website

Table 1: Properties of the datasets.

Data	Train set		Test set		Items
	Sessions	Events	Sessions	Events	
VIDXL	17,419,964	69,312,698	216,725	921,202	712,824
CLASS	1,173,094	9,011,321	35,741	254,857	339,055

Experiments

- Performance of p-RNN, ID only RNN, and item-KNN.
- p-RNN with features incorporated clearly outperforms the other two approaches.



Session-based context modeling

- First-order dependency modeling
- Higher-order dependency modeling
- Loosely ordered dependency modeling

Loosely ordered sequence in session

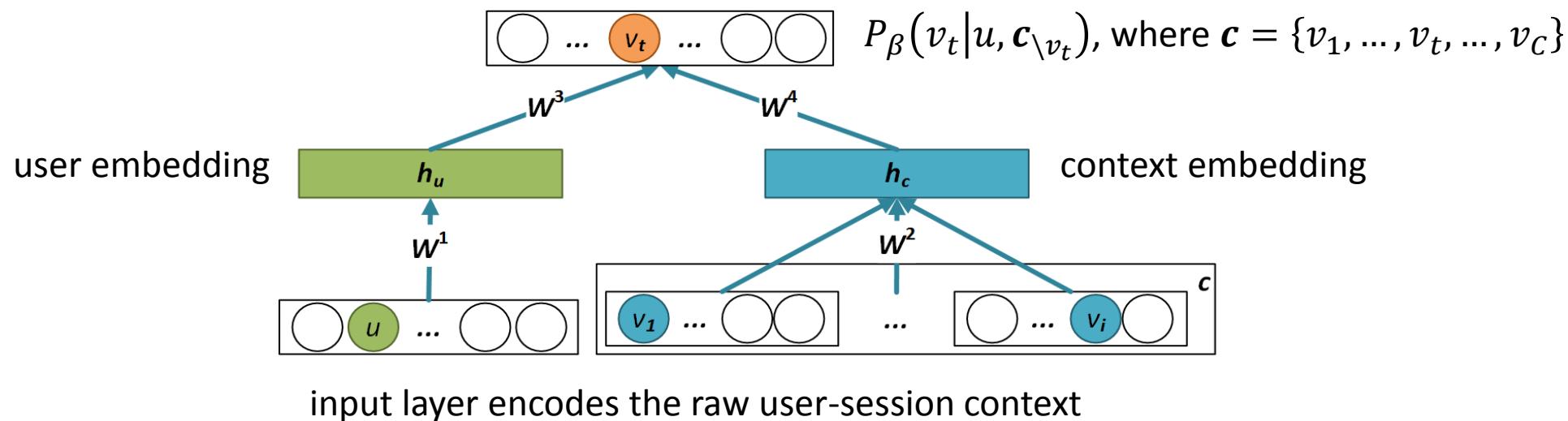
- The choices of items in a session may **not follow a rigidly ordered sequence**
- For example, toast and milk, which is first put into a shopping cart is not sensitive to the next choice.



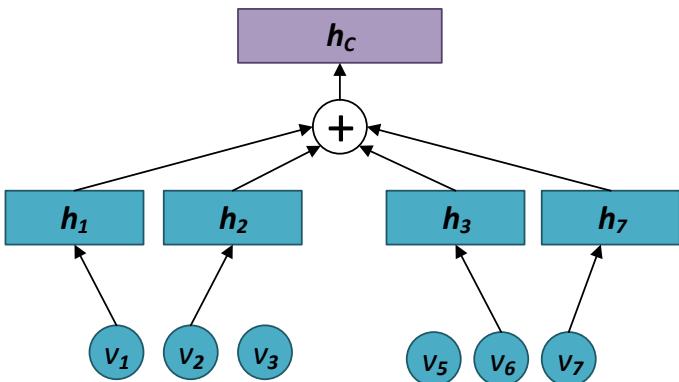
Wide-in-wide-out Shallow Networks

- SWIWO Architecture (Inspired by CBOW)
 - Three-layer shallow wide-in-wide-out networks

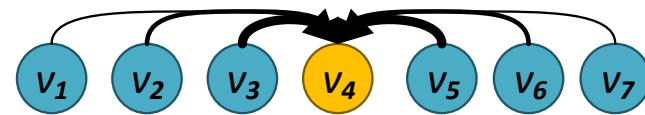
softmax layer to model the probability of choice



Weight assignment on context items



$$\mathbf{h}_c = \sum_{v \in \mathbf{c}} w_v \mathbf{h}_v$$



The context items previous and next to the target item v_t , i.e. v_{t-1} and v_{t+1} , have the largest weight, and those context items farther from v_t are assigned smaller weights.

$$w_v \propto \exp[-\lambda(|v - t| - 1)]$$

Dataset

- IJCAI-15 Dataset: <https://tianchi.aliyun.com/datalab/dataSet.htm?id=5>
 - This real-world dataset was collected from Tmall.com which is the largest online B2C platform in China, and it contains anonymized users' shopping logs for the six months before and on the “Double 11” day (November 11th).

Statistic of IJCAI-15 dataset for evaluation
#users: 50K
#items: 52K
avg. session length: 2.99
#training sessions: & 0.20M
#training examples: & 0.59M
#testing cases (<i>LAST</i>): 4.5K
#testing cases (<i>LOO</i>): 11.9K

Accuracy Evaluation

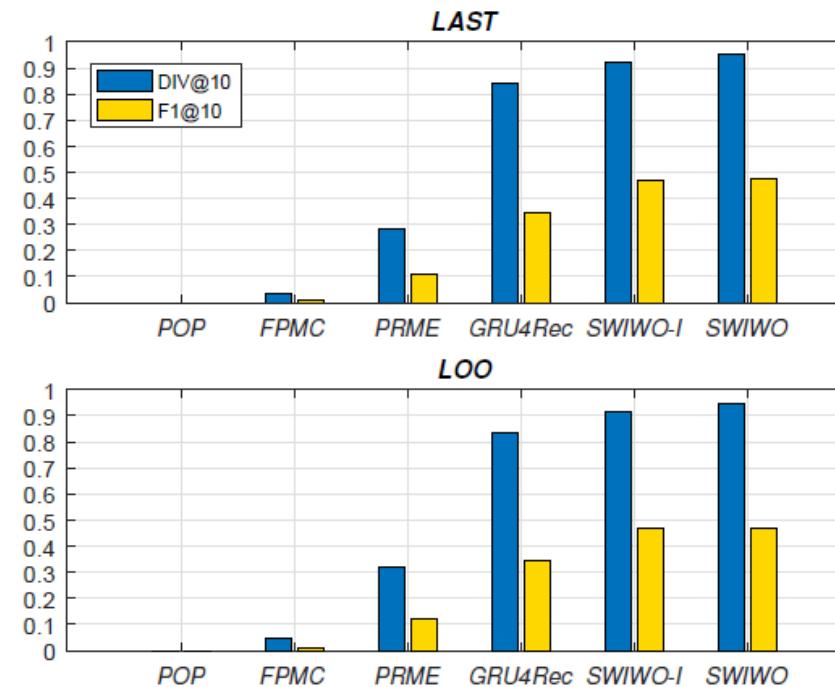
- The result of $REC@10$, $REC@20$ and MRR over the testing sets ***Last*** and ***LOO***.

LAST			
Model	REC@10	REC@20	MRR
<i>POP</i>	0.0185	0.0317	0.0104
<i>FPMC</i>	0.0023	0.0068	0.0021
<i>PRME</i>	0.0670	0.0821	0.0363
<i>GRU4Rec</i>	0.2283	0.2464	0.1586
<i>SWIWO-I</i>	0.3223	0.3797	0.1918
<i>SWIWO</i>	0.3131	0.3689	0.1896

LOO			
Model	REC@10	REC@20	MRR
<i>POP</i>	0.0234	0.0420	0.0123
<i>FPMC</i>	0.0064	0.0117	0.0044
<i>PRME</i>	0.0757	0.0976	0.0431
<i>GRU4Rec</i>	0.2242	0.2425	0.1574
<i>SWIWO-I</i>	0.3177	0.3810	0.1903
<i>SWIWO</i>	0.3082	0.3703	0.1885

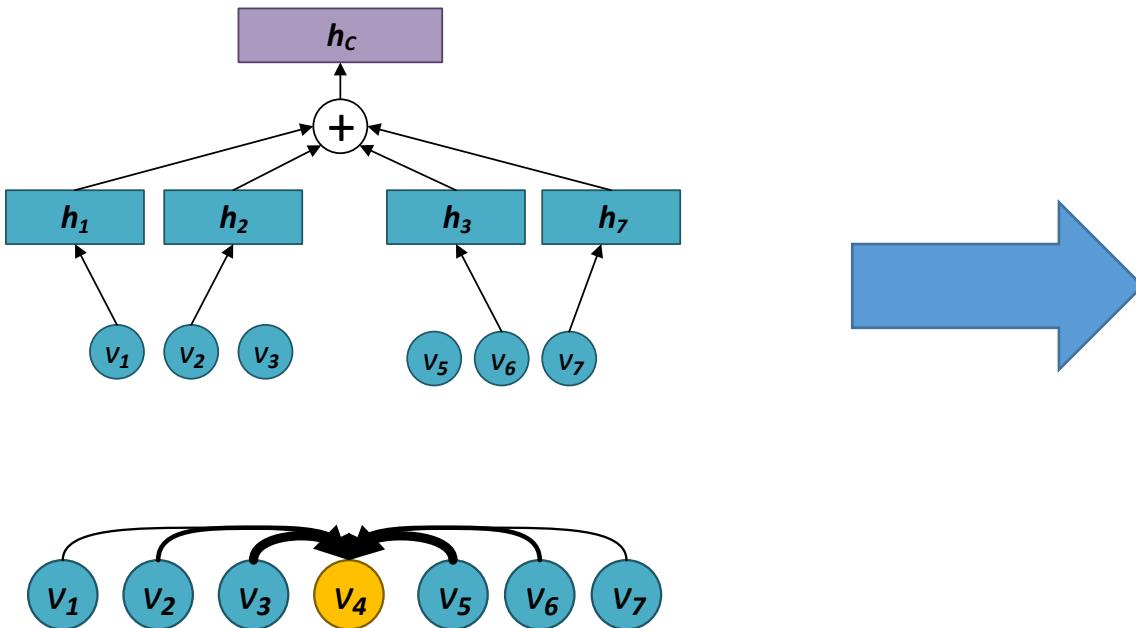
Diversity Evaluation

- SWIWO considers the whole session context so they more easily provide diverse recommendation results.



Extension: Weight transaction embedding with attention mechanism

Context items **contribute differently** to the next choice



Datasets

- IJCAI-15 Dataset
- Tafang Dataset
 - This real-world dataset is a grocery shopping-supermarket dataset collected from a supermarket from November 2001 to February 2002.

Table 1: Statistics of experimental datasets

Statistics	IJCAI-15	Tafang
#Transactions	144,936	19,538
#Items	27,863	5,263
Avg. Transaction Length	2.91	7.41
#Training Transactions	141,840	18,840
#Training Instances	412,679	141,768
#Testing Transactions	3,096	698
#Testing Instances	9,030	3,150

Experiments

- ATEM achieves best performance compared to baselines.
- Attention mechanism **contributes greatly** by comparing ATEM and TEM, a similar model **without** attention mechanism.

Table 2: Accuracy comparisons on IJCAI-15

Model	REC@10	REC@50	MRR
<i>PBRS</i>	0.0780	0.0998	0.0245
<i>FPMC</i>	0.0211	0.0602	0.0232
<i>PRME</i>	0.0555	0.0612	0.0405
<i>GRU4Rec</i>	0.2283	0.3021	0.1586
ATEM	0.3542	0.5134	0.2041
<i>TEM</i>	0.3177	0.3796	0.1918

Table 3: Accuracy comparisons on Tafang

Model	REC@10	REC@50	MRR
<i>PBRS</i>	0.0307	0.0307	0.0133
<i>FPMC</i>	0.0191	0.0263	0.0190
<i>PRME</i>	0.0212	0.0305	0.0102
<i>GRU4Rec</i>	0.0628	0.0907	0.0271
ATEM	0.1089	0.2016	0.0347
<i>TEM</i>	0.0789	0.1716	0.0231

Experiments

- Test the robustness to the **item order** within session
- ATEM is almost not affected when randomly disordering items.

Table 2: Accuracy comparisons on IJCAI-15

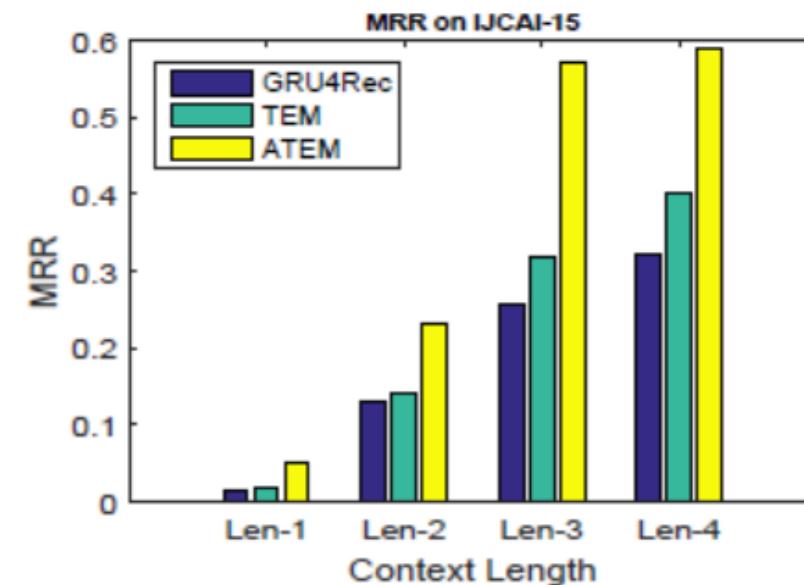
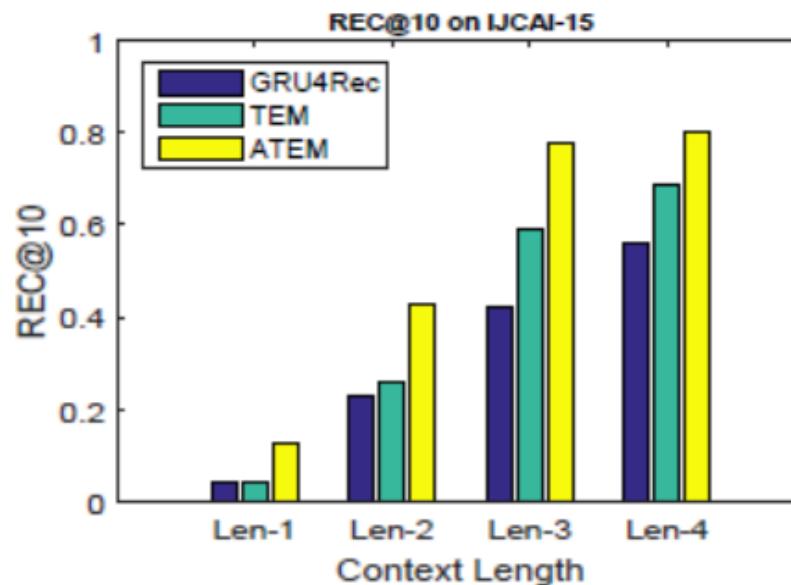
Model	REC@10	REC@50	MRR
<i>PBRS</i>	0.0780	0.0998	0.0245
<i>FPMC</i>	0.0211	0.0602	0.0232
<i>PRME</i>	0.0555	0.0612	0.0405
<i>GRU4Rec</i>	0.2283	0.3021	0.1586
ATEM	0.3542	0.5134	0.2041
<i>TEM</i>	0.3177	0.3796	0.1918

Table 4: Accuracy on disordered IJCAI-15

Model	REC@10	REC@50	MRR
<i>PBRS</i>	0.0500	0.0559	0.0185
<i>FPMC</i>	0.0151	0.0412	0.0183
<i>PRME</i>	0.0346	0.0389	0.0351
<i>GRU4Rec</i>	0.1636	0.2121	0.1022
ATEM	0.3423	0.4981	0.1960
<i>TEM</i>	0.2660	0.3012	0.1431

Experiments

- Test the effect of context length
 - ATEM outperforms other methods on longer context, which proves attention mechanism **effectively assigning large weights to related items in context.**

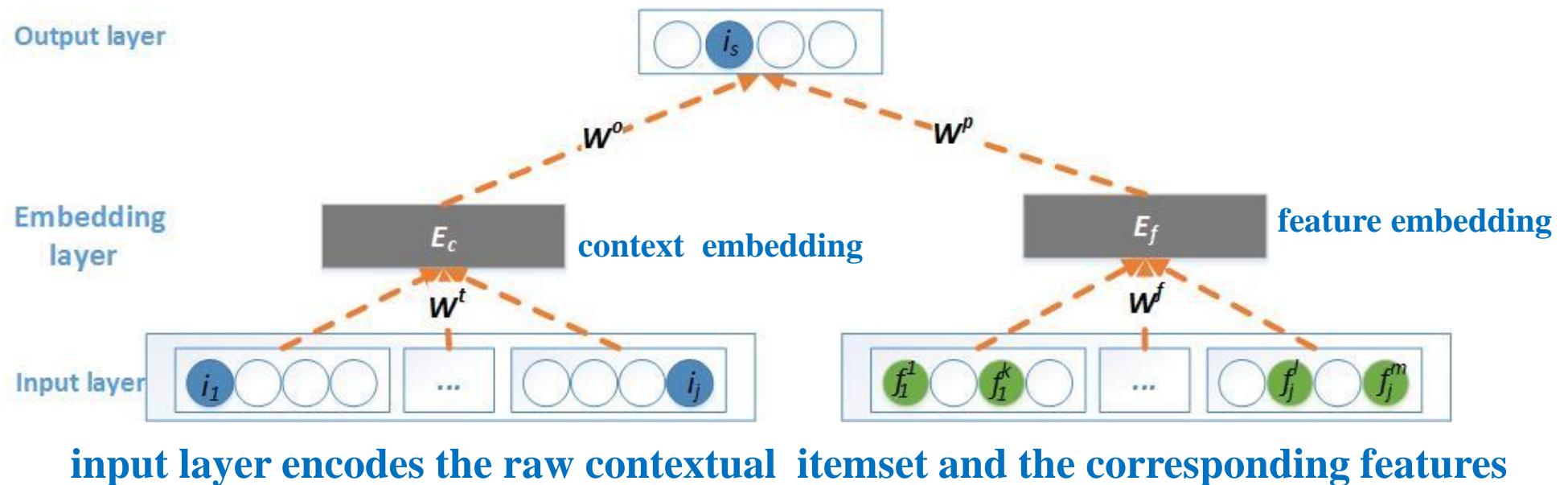


Another extension: Embedding attributes for cold start recommendations

- Previous models cannot recommend items which rarely occurred or totally new items.
- We incorporate the item features into the embedding model to handle such cold-start item recommendation issue.

NTEM Architecture

- Three-layer shallow wide-in-wide-out networks



Accuracy Evaluation

- The result of $REC@10$, $REC@50$ and MRR over the testing sets of two real-world datasets.

		IJCAI-15			Tafang		
Scenario	Model	REC@10	REC@50	MRR	REC@10	REC@50	MRR
drop 0	FPMC	0.0016	0.0025	0.0031	0.0189	0.0216	0.0089
	PRME	0.0555	0.0612	0.0405	0.0212	0.0305	0.0102
	GRU4Rec	0.1182	0.1566	0.0965	0.0428	0.0887	0.0221
	NTEM	0.2026	0.3224	0.1125	0.0689	0.1716	0.0231
drop 40%	FPMC	0.0012	0.0021	0.0026	0.0008	0.0010	0.0058
	PRME	0.0327	0.0411	0.0312	0.0102	0.0205	0.0095
	GRU4Rec	0.1108	0.1356	0.0868	0.0330	0.0659	0.0196
	NTEM	0.1928	0.2794	0.1117	0.0575	0.1049	0.0377
drop 80%	FPMC	0.0009	0.0017	0.0021	0.0005	0.0008	0.0020
	PRME	0.0212	0.0287	0.0215	0.0084	0.0125	0.0056
	GRU4Rec	0.0493	0.0611	0.0398	0.0110	0.0244	0.0054
	NTEM	0.1098	0.1450	0.0686	0.0254	0.0494	0.0072
drop 95%	FPMC	0.0003	0.0008	0.0012	0.0002	0.0004	0.0008
	PRME	0.0089	0.0113	0.0105	0.0071	0.0096	0.0043
	GRU4Rec	0.0233	0.0337	0.0173	0.0101	0.0172	0.0042
	NTEM	0.0318	0.0639	0.0173	0.0215	0.0305	0.0068

We build datasets with different cold start levels to test our model's capability on cold start recommendations.

Novelty Evaluation

- We aim to recommend some novel items with the consideration of transactional context and the incorporation of item features.
- Now, let's consider the following metrics.

Global novelty M²ITF: the opposite of item popularity w.r.t the whole population.

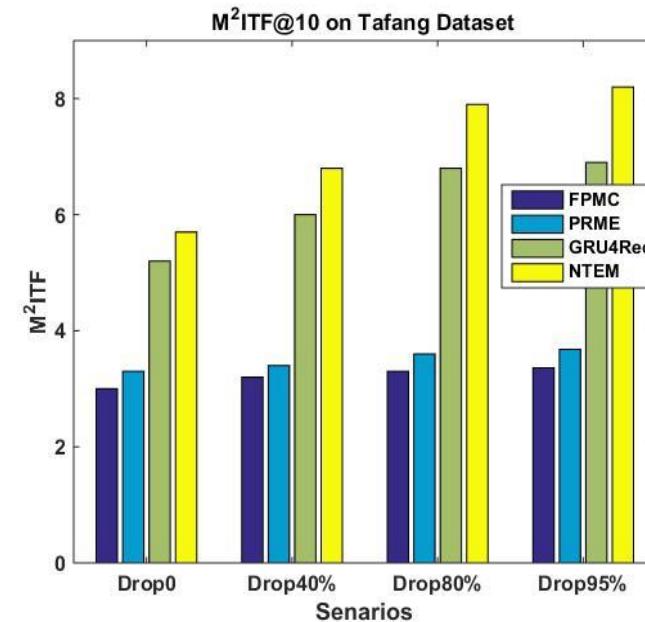
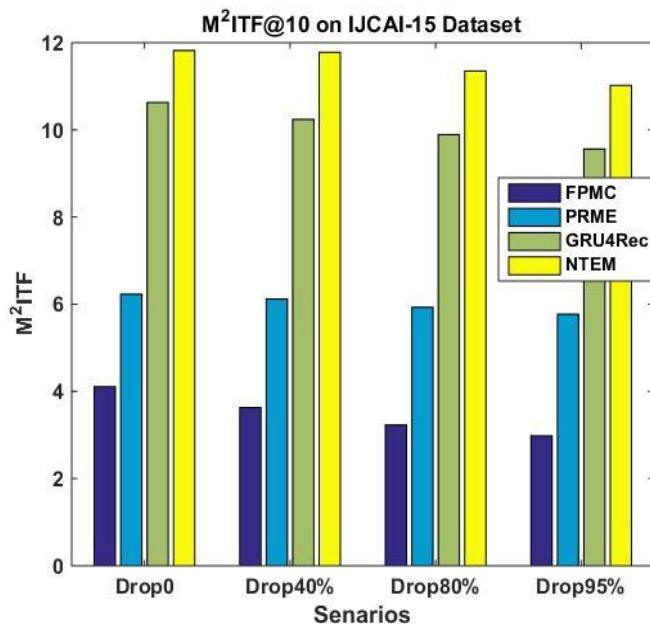
$$MITF = -\frac{1}{|R|} \sum_{i \in R} \log_2 \frac{|T_i|}{|T|} \quad M^2ITF = \frac{1}{N} \sum MITF$$

Local novelty MCAN: the difference of recommended list R w.r.t the corresponding context c.

$$CAN = 1 - \frac{|R \cap c|}{|R|} \quad MCAN = \frac{1}{N} \sum CAN$$

Novelty Evaluation

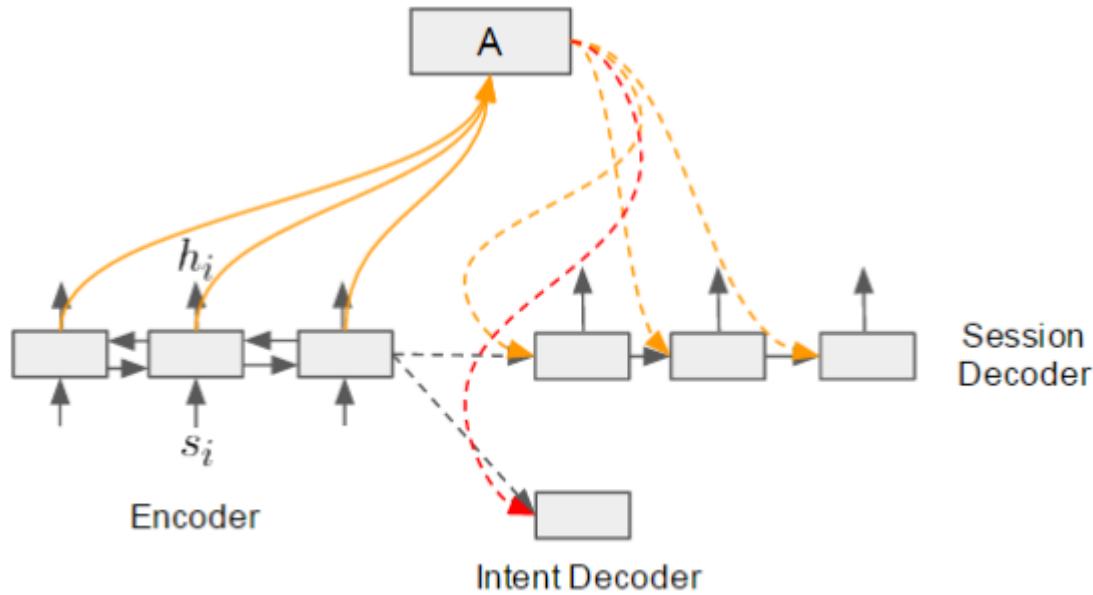
- NTEM incorporates the item features, so it is easier to discover and recommend those unpopular but relevant items.



Encoder-Decoder for Session Modeling

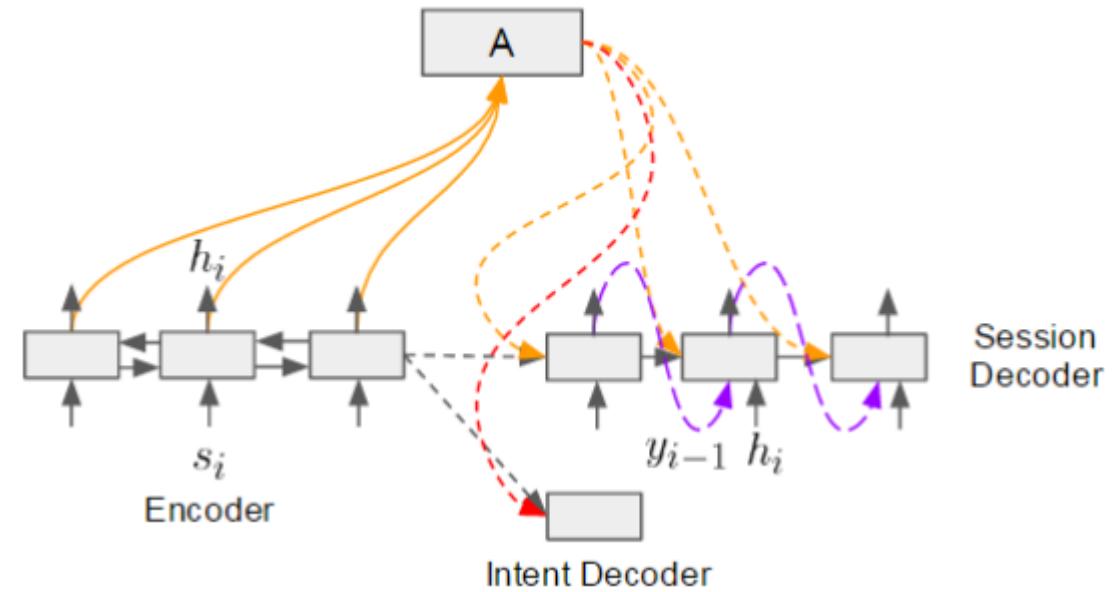
- Motivation and Key Idea
 - Using an encoder-decoder neural architecture to model user session
 - Incorporating attention mechanism to learn the more expressive portions of items in a sequence.

Encoder-Decoder Architecture:



model for session and intent modeling with attention

A bidirectional RNN is used for the encoder to load the item sequence. Decoder is a unidirectional RNN.



model for session and intent modeling with attention + alignments

Explicit information transfer with alignment, passing both the emitted label y_{i-1} and the internal hidden state h_i at time t to the decoder.

Experiments

- Results of different approaches and their variations, encoder-decoder with attention achieves the best performance.

Model	Recall@20	MRR@20
Item-KNN	0.327	0.139
BPR-MF	0.310	0.135
GRU4Rec	0.3481	0.189
GRU4Rec (cross-entropy)	0.3506	0.207
EDRec	0.3775	0.214
EDRec w/ alignment	0.3905	0.249
EDRec w/ alignment and attention	0.3914	0.231

Open issues and future directions

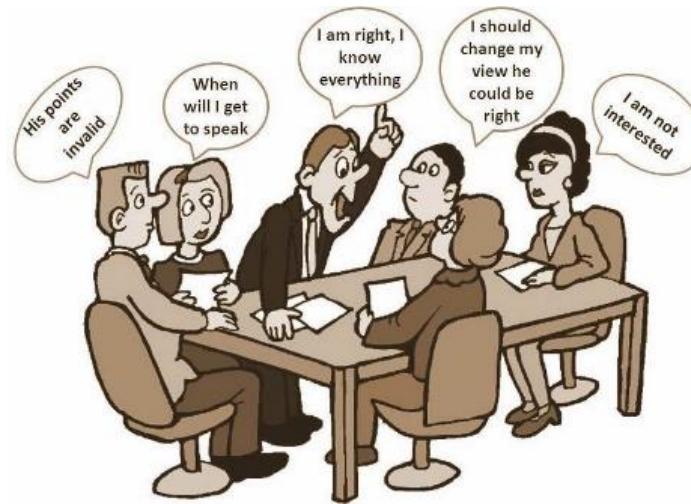
- Open issues
 - How to deal with long-term dependency, e.g., long sessions
 - How to find out really relevant items from a long session
 - How to degrade the influence of irrelevant items in a session
- Future directions
 - Incorporating cross-session dependency
 - Involving more side information, e.g., attributes, text, images
 - Involving more contextual information, e.g., weather, locations

Section III:

- Leveraging Contextual Information
 - Context Modeling in Recommender Systems
 - Session-based Recommender Systems
 - Group-based Recommender Systems
- Real Recommender Systems in Practice

Group choices are joint decision

- Group activities are observed throughout life
 - E.g., watching a family movie, planning family travel
- Each member of a group may have **different opinions** on the same items, so the main challenge in GRSs is to satisfy most group members with **diverse preferences**.
- This cannot be achieved through an individual-based recommendation method.



Profile Aggregation

- Group Preference Aggregation (GPA) (Pre-aggregation)
 - Aggregating all members' ratings into a group profile
 - Groups are regarded as virtual individual users.
 - **Disadvantage:** the preference is biased to active users with more data
- Individual Preference Aggregation (IPA) (Post-aggregation)
 - Predicting the individual ratings over candidate items
 - Aggregating the predicted ratings of members via predefined strategies.
 - **Disadvantage:** IPA fails to consider the group behavior

Overview of Aggregation Strategies for Group Recommendation

- Many strategies exist for aggregating individual ratings into a group rating (e.g. used in elections and when selecting a party leader)

Strategy	How it works	Example
Plurality voting	Uses ‘first past the post’: repetitively, the item with the most votes is chosen.	A is chosen first, as it has the highest rating for the majority of the group, followed by E (which has the highest rating for the majority when excluding A).
Average	Averages individual ratings	B’s group rating is 6, namely $(4 + 9 + 5)/3$.
Multiplicative	Multiplies individual ratings	B’s group rating is 180, namely $4 \cdot 9 \cdot 5$.
Borda count	Counts points from items’ rankings in the individuals’ preference lists, with bottom item getting 0 points, next one up getting one point, etc.	A’s group rating is 17, namely 0 (last for Jane) + 9 (first for Mary) + 8 (shared top 3 for Peter)
Copeland rule	Counts how often an item beats other items (using majority vote ^a) minus how often it loses	F’s group rating is 5, as F beats 7 items (B,C,D,G,H,I,J) and loses from 2 (A,E).
Approval voting	Counts the individuals with ratings for the item above a approval threshold (e.g. 6)	B’s group rating is 1 and F’s is 3.
Least misery	Takes the minimum of individual ratings	B’s group rating is 4, namely the smallest of 4,9,5.
Most pleasure	Takes the maximum of individual ratings	B’s group rating is 9, namely the largest of 4,9,5.
Average without misery	Averages individual ratings, after excluding items with individual ratings below a certain threshold (say 4).	J’s group rating is 7.3 (the average of 8,8,6), while A is excluded because Jane hates it.
Fairness	Items are ranked as if individuals are choosing them in turn.	Item E may be chosen first (highest for Peter), followed by F (highest for Jane) and A (highest for Mary).
Most respected person (or Dictatorship)	Uses the rating of the most respected individual.	If Jane is the most respected person, then A’s group rating is 1. If Mary is most respected, then it is 10.

Most Frequently Used Aggregation Strategies

- *Average* and *Least misery* are the two most prevalent strategies.
- ***Least misery strategy*** assumes a group tends to be as happy as its **least happy member**.
- ***Average strategy*** recommends items with the highest average ratings over all members.

Fixed group and Flexible group

- Fixed group-based recommendation
 - A family
 - A working group
- Flexible group-based recommendation
 - Friends meetup
 - Conference attenders

Ranking oriented feature-based matrix factorization

- Feature-based matrix factorization

$$y = f \left(\mu + \left(\sum_j b_j^{(g)} \gamma_j + \sum_j b_j^{(u)} \alpha_j + \sum_j b_j^{(i)} \beta_j \right) + \left(\sum_j p_j \alpha_j \right)^T \left(\sum_j q_j \beta_j \right) \right)$$

- where α , β , γ denote user, item, and global features respectively

Household recommendation

- **Individual Preference Aggregation:** First get the predicted users' preference for items and then combine the household members' preference to get household preference.

$$\hat{r}_{h,i} = \sum_{u \in H(h)} w_u \cdot \hat{r}_{u,i}$$

where w is set to 1 for each member (Average strategy)

- **Group Preference Aggregation:** Learning preference of household directly. First get the ratings of household according to its member's rating and then adopt MF on household-item ratings.

$$r_{h,i} = \frac{\sum_{u \in H(h)} r_{u,i}}{|H(h)|}$$

The rating of a household equals to the average rating of its members (Average strategy)

Dataset for group recommendation

- CAMRa2011 dataset containing the movie watching records of households and the ratings on each watched movie given by some group members.
- The dataset for track 1 of CAMRa2011 has 290 households with a total of 602 users who gave ratings (on a scale 1~100) over 7,740 movies.

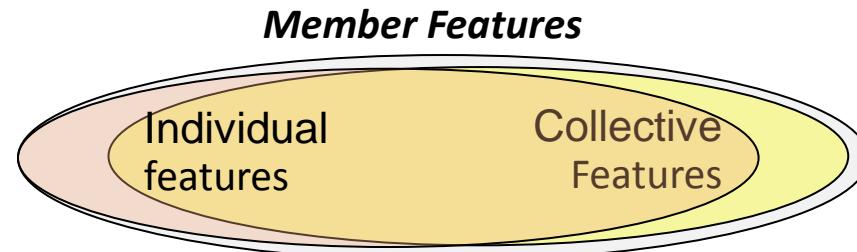
Experimental results

- Comparisons of recommendation performance:

Models	MAP	AUC	P@5	P@10
BMF	0.1390	0.8374	0.1344	0.1051
BMF-3N	0.2268	0.9926	0.2039	0.1680
BMF-3N-HIR	0.2315	0.9910	0.2124	0.1718
BMF-3N-IMFB	0.2383	0.9940	0.2150	0.1727
BMF-3N-I100NN	0.2614	0.9922	0.2402	0.1968
BMF-3N-ALL	0.2639	0.9924	0.2435	0.1970
RMF-S1	0.2053	0.9931	0.1931	0.1608
RMF-S2	0.2275	0.9939	0.2065	0.1741
RMF-S2-HIR	0.2387	0.9943	0.2167	0.1814
RMF-S2-IMFB	0.2477	0.9943	0.2322	0.1893
RMF-S2-I100NN	0.2847	0.9936	0.2550	0.2021
RMF-S2-ALL	0.3096	0.9956	0.2872	0.2190

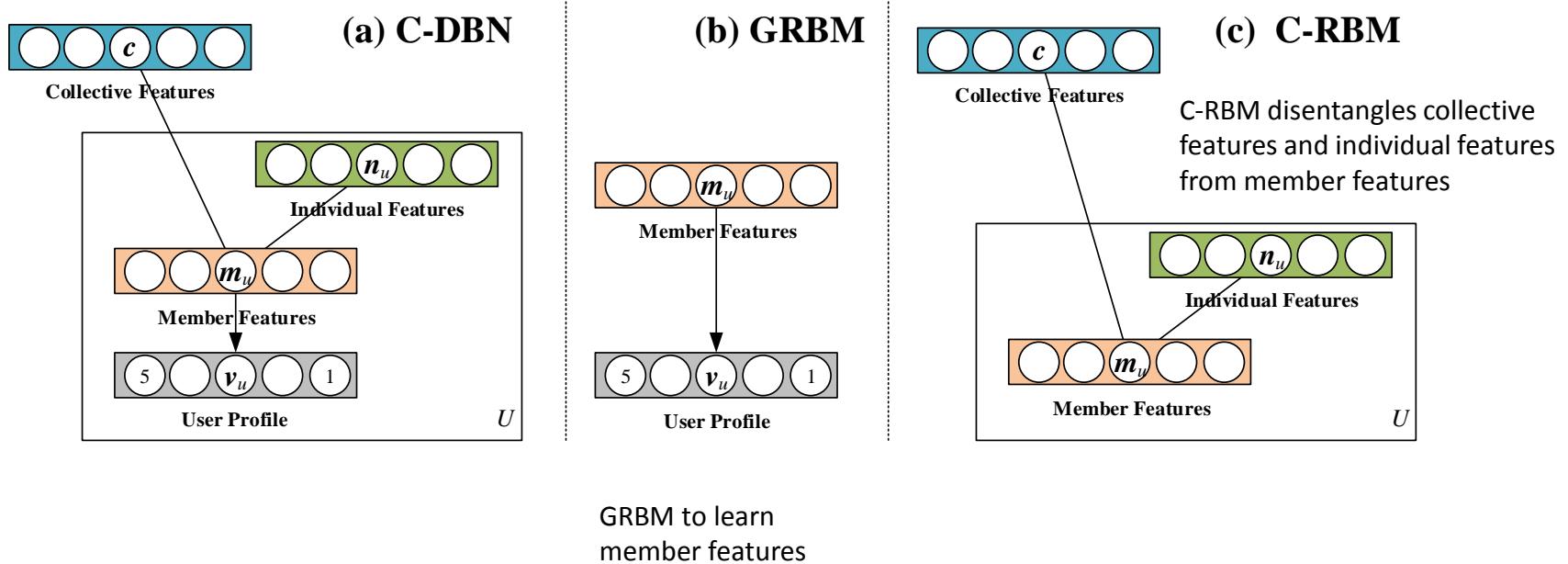
Modeling Features in Group-based Decision

- **Member Features**: these model the individual preference of a user when she/he makes choices as a group member, which can be regarded as **a mixture of Collective Features and Individual Features**.
- **Collective Features**: these represent compromised preferences of a group, which are **shared among all members** and can be disentangled from the *Member Features*.
- **Individual Features**: these represent **independent individual-specific preference**, which can be disentangled from the *Member Features* w.r.t. this user.



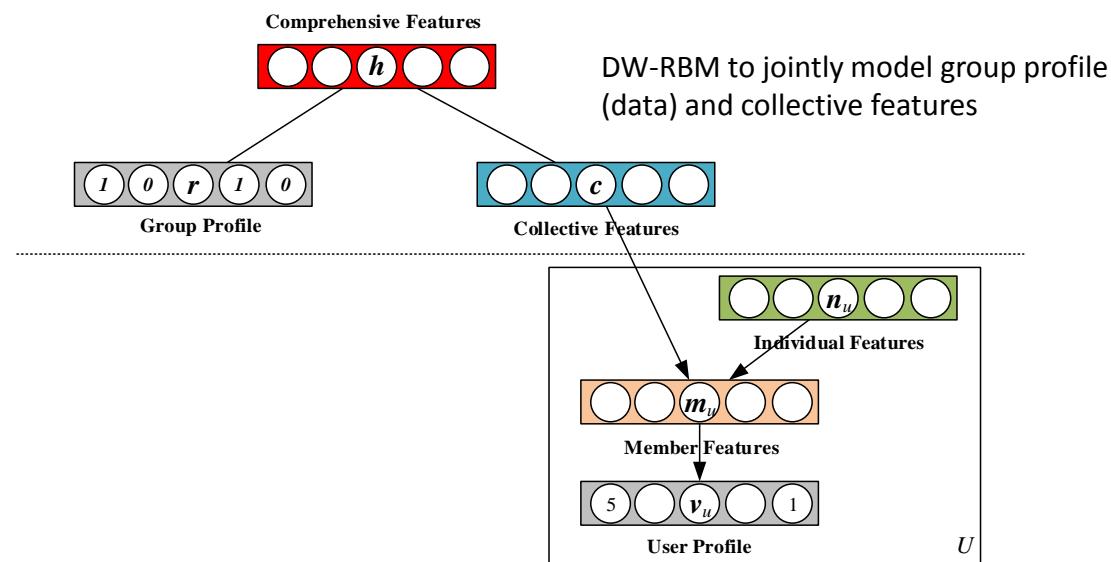
Disentangling Collective and Individual Features

- Each group choice can be regarded as a **joint decision** by all members



Comprehensive Representation of Group Preferences

- A dual-wing RBM is placed on the top of DBN, which jointly models the group choices and collective features to learn the **comprehensive** features of group preference



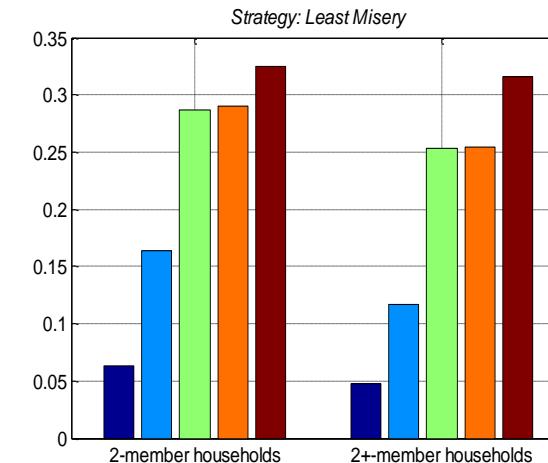
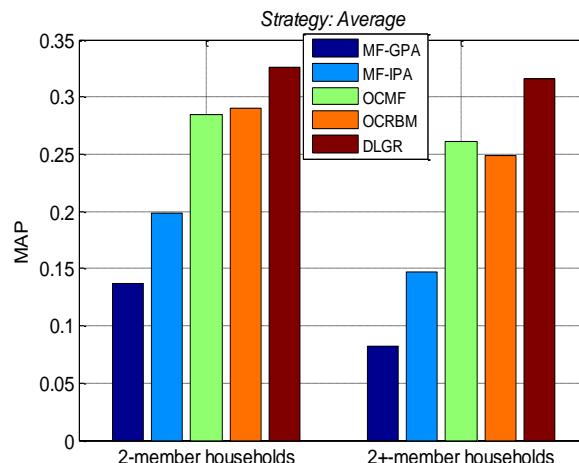
Results

MAP and mean AUC of all comparative models with different strategies

Model/Strategy	MAP			AUC		
	No Strategy	Average	Least Misery	No Strategy	Average	Least Misery
<i>kNN (k=5)</i>	0.1595	N/A	N/A	0.9367	N/A	N/A
<i>MF-GPA</i>	N/A	0.1341	0.0628	N/A	0.9535	0.9297
<i>MF-IPA</i>	N/A	0.1952	0.1617	N/A	0.9635	0.9503
<i>OCMF</i>	0.2811	0.2858	0.2801	0.9811	0.9813	0.9810
<i>OCRBM</i>	0.2823	0.2922	0.2951	0.9761	0.9778	0.9782
<i>DLGR</i>	0.3236	0.3252	0.3258	0.9880	0.9892	0.9897

Group with different number of members

- A group with **more members implies more different preferences**, so it is harder to find recommendations satisfying all members.
- Each household may contain 2~4 members in this dataset. We additionally evaluated the MAP w.r.t. 2-member households and the 2+-member (>2) households under *Average* and *Least Misery* strategies.

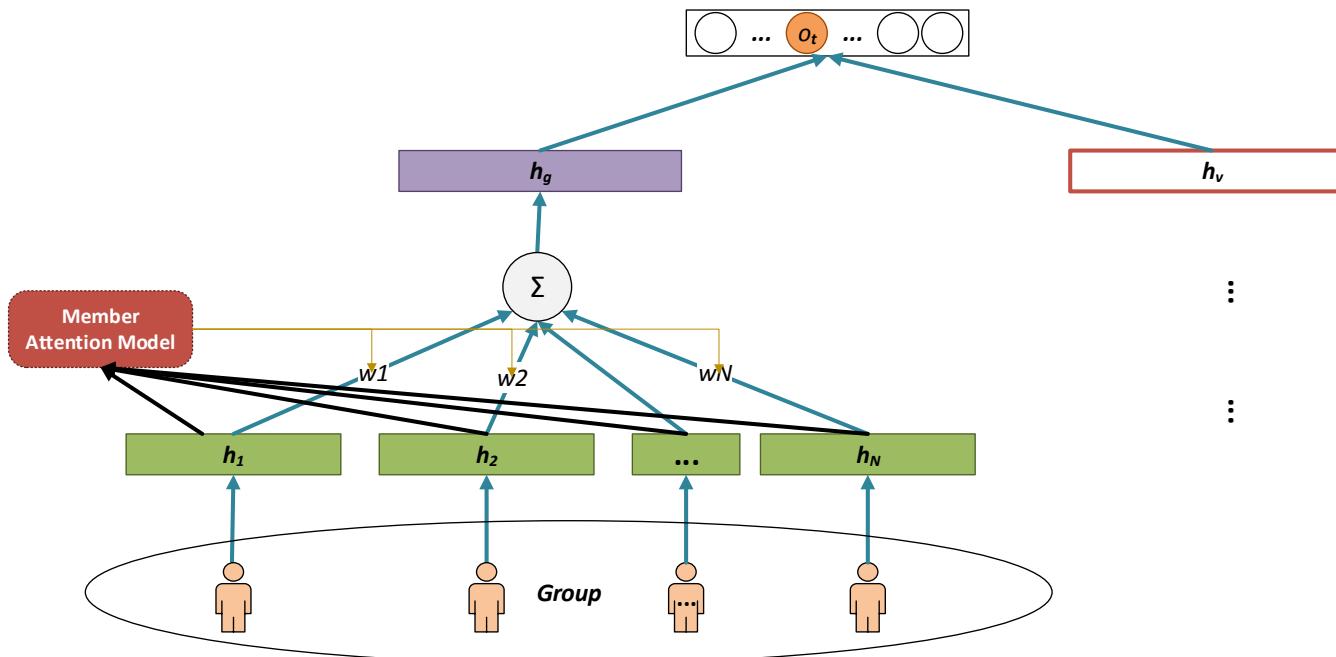


Open issues and directions

- Lack of group feedback data
 - There are very few real-world public datasets
 - Most datasets are synthetic from personal feedback, which does not contain the features of group decision
- Learning to represent group context given a group of any users
- Dynamic group recommendation with contextual information

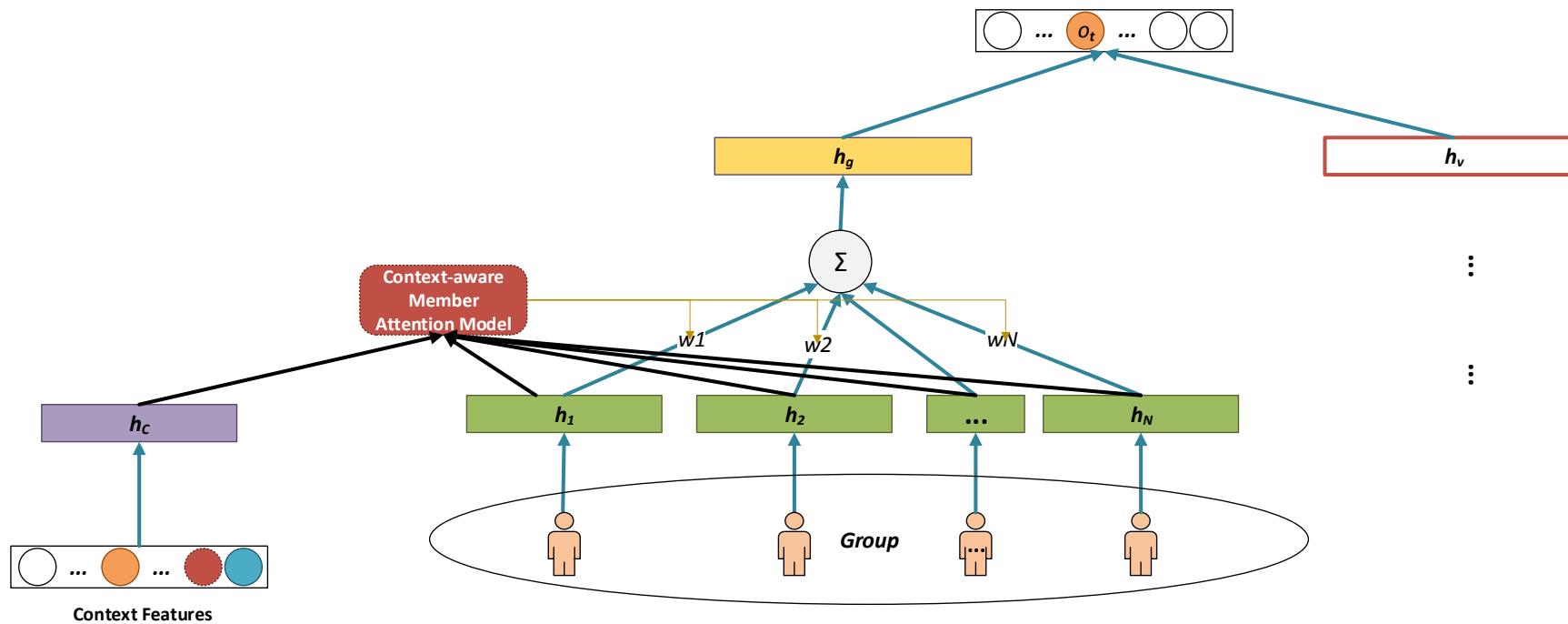
Using attention mechanism

- Most aggregation strategies are about how to weight group members
- Member attention model to learn how to assign weights on members



Context-aware group recommendation

- Each group member plays different role in different context
 - Assign different weights in different context



Section III:

- Leveraging Contextual Information
 - Context Modeling in Recommender Systems
 - Session-based Recommender Systems
 - Group-based Recommender Systems
- Real-world Recommender Systems

Evolution of Netflix



2006



2016



Recommendation in Netflix

<https://medium.com/netflix-techblog>

Recommending for the World

#AlgorithmsEverywhere

by [Yves Raimond](#) and [Justin Basilico](#)



<https://medium.com/netflix-techblog/recommending-for-the-world-8da8cbcf051b>

Netflix Prize

- In October 2006, Netflix as a service peddling discs of movie and TV show, announced “The Netflix Prize”
- The Netflix Prize was an open competition for the best collaborative filtering algorithm to predict user ratings for films
- The mission: Make the company's recommendation engine 10% more accurate

Netflix Prize in 2012

Netflix Prize

COMPLETED

What we were interested in:

- High quality *recommendations*

Proxy question:

- Accuracy in predicted rating
- Improve by 10% = \$1million!

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{j=1}^n (y_j - \hat{y}_j)^2}$$

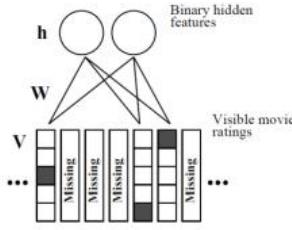


SVD

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} | & | \\ u_1 & u_2 \\ | & | \end{bmatrix} \times \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \times \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

Results

- Top 2 algorithms still in production



RBM

MF wins Netflix Prize (2012)

SVD for Rating Prediction

- User factor vectors $p_u \in \Re^f$ and item-factors vector $q_v \in \Re^f$
- Baseline $b_{uv} = \mu + b_u + b_v$ (user & item deviation from average)
- Predict rating as $\hat{r}_{uv} = b_{uv} + p_u^T q_v$
- **SVD++** (Koren et. Al) asymmetric variation w. implicit feedback

$$\hat{r}_{uv} = b_{uv} + q_v^T \left(|R(u)|^{-\frac{1}{2}} \sum_{j \in R(u)} (r_{uj} - b_{uj}) x_j + |N(u)|^{-\frac{1}{2}} \sum_{j \in N(u)} y_j \right)$$

- Where
 - $q_v, x_v, y_v \in \Re^f$ are three item factor vectors
 - Users are not parametrized, but rather represented by:
 - $R(u)$: items rated by user u
 - $N(u)$: items for which the user has given implicit preference (e.g. rated vs. not rated)



RBM wins Netflix Prize (2012)

RBM for the Netflix Prize

Binary hidden features

Visible movie ratings

...

...

NE

Restricted Boltzmann Machines
for Collaborative Filtering

Ruslan Salakhutdinov
Andriy Mnih
Geoffrey Hinton
University of Toronto, 6 King's College Rd., Toronto, Ontario M5S 3G4, Canada

RSALAKHU@CS.TORONTO.EDU
AMNIH@CS.TORONTO.EDU
HINTON@CS.TORONTO.EDU

Figure 1. A restricted Boltzmann machine with binary hidden units and softmax visible units. For each user, the RBM only includes softmax units for the movies that user has rated. In addition to the symmetric weights between each hidden unit and each of the $K = 5$ values of a softmax unit, there are 5 biases for each softmax unit and one for each hidden unit. When modeling user ratings with an RBM that has Gaussian hidden units, the top layer is composed of linear units with Gaussian noise.

31

Most watch on Netflix comes from Recommendation (2013)

TECH 08/01/2013 08:00 am ET | Updated Aug 01, 2013

Netflix Launches Profiles, Finally Realizing How People Really Watch Movies On It

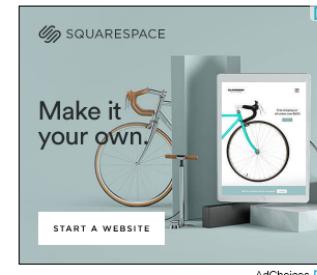


By Timothy Stenovec



For years, people who share Netflix accounts have befuddled the streaming service's recommendation engine, the tool that in theory is supposed to use what you've watched before to suggest movies, documentaries and TV shows you'd like. But your kids may stream Disney movies and Sesame Street, and you may binge on episodes of "House of Cards" and "Breaking Bad," leading Netflix to suggest movies and TV shows that may not appeal to anyone in your household.

In an attempt to fix this, Netflix today begins rolling out profiles, a free feature that allows any of the company's 37 million subscribers to create up to five different profiles on one account. Each profile will be treated like its own account, so recommendations will be more aligned with a single person's interests.



TRENDING



Donald Trump Stayed On The Golf Course As Hawaii Panicked



HITROL Projector Lights Up Trump's D.C. Hotel With 'Shithole' And Poop Emojis



Bill Murray Slays As The 'Bannon Cannon' On 'Saturday Night Live'



Liam Neeson Calls The #MeToo Movement A 'Bit Of A Witch Hunt'

Figuring out what people want to watch is key to Netflix's success. In an increasingly competitive streaming environment, where Hulu Plus and Amazon Prime Instant Video ink their own deals for exclusive and original content, Netflix needs not only to continue to attract new subscribers, but also keep existing ones happy. One way the company can do that — and keep people from ditching its service for a competitor — is by suggesting content that subscribers will like.

Introducing profiles is a move to combat "churn," the number of people who sign up and then quit paying the \$7.99 monthly fee if they feel like it's not valuable, said Mike McGuire, a vice president at Gartner, the technology research firm.

"When you're in the subscription business, churn is your worst enemy," said McGuire. "If there's not something else they're surfacing that meets your interest beyond what you initially dialed in for, then you're out."

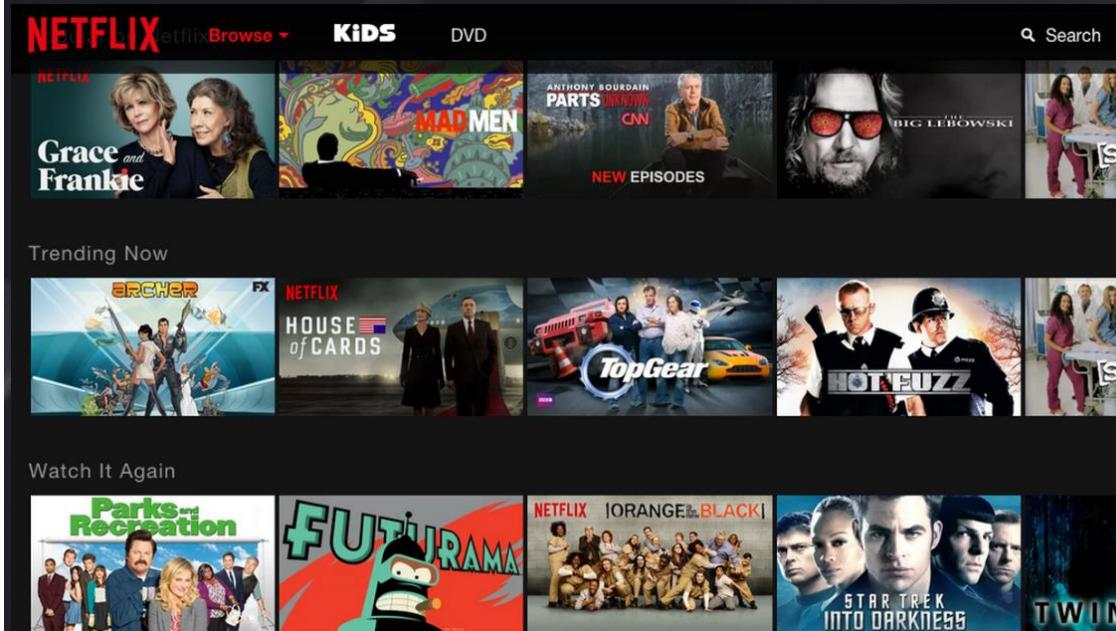
About 75 percent to 80 percent of what people watch on Netflix comes from what Netflix recommends, not from what people search for, said Yellin.

Netflix has applied deep learning for recommendation

Deep Learning with Tony Jebara, Director of ML Research at Netflix

By Sophie Curtis, Marketing Director - RE-WORK

May 26, 2016



Tell us more about your work as Director of Machine Learning Research at Netflix.

At Netflix we are inventing the future of Internet television and helping members across the world find videos to watch and enjoy. We help them make a selection from a catalog of thousands of titles. But we need to tailor recommendations to each user and each session within seconds and within a menu of 10 to 20 visible options. Achieving this relies on our recommendation system which is really an ecosystem of many machine learning algorithms that operate together. We are constantly working on improving these algorithms. We are also leveraging machine learning across all parts of Netflix: from deciding which new titles to add to our catalog to finding ways to more efficiently stream videos across the Internet.

What do you feel are the leading factors enabling recent advancements and uptake of deep learning?

The recent adoption of deep learning has been enabled by the confluence of several factors. Of course, bigger data-sets and more computational power have been essential. But, they triggered something more important: the freedom to try bigger and deeper models. Recent theoretical research by Choromanska et al shows that models with more layers and more neurons per layer become resilient to the local optima that plague the optimization landscape. Bigger models find more reliable solutions during learning while small ones get stuck in bad solutions. So, it's been a chain reaction: bigger computation led to bigger data, then to bigger models, then to better optima, and finally to better performance.

What are your thoughts on the recent surge of media interest surrounding deep learning?

The media interest certainly adds to the excitement. The uptake in press and articles has often revolved on deep learning shattering AI milestones in areas such as game-playing, computer vision and so on. But more practical progress is happening in business and commercial fronts where deep learning and machine learning are permeating almost every component of the workplace. So, beyond exciting milestones in the media (such as beating the grandmaster of Go), we are seeing a sustained ground swell in deep learning at companies all over the world.

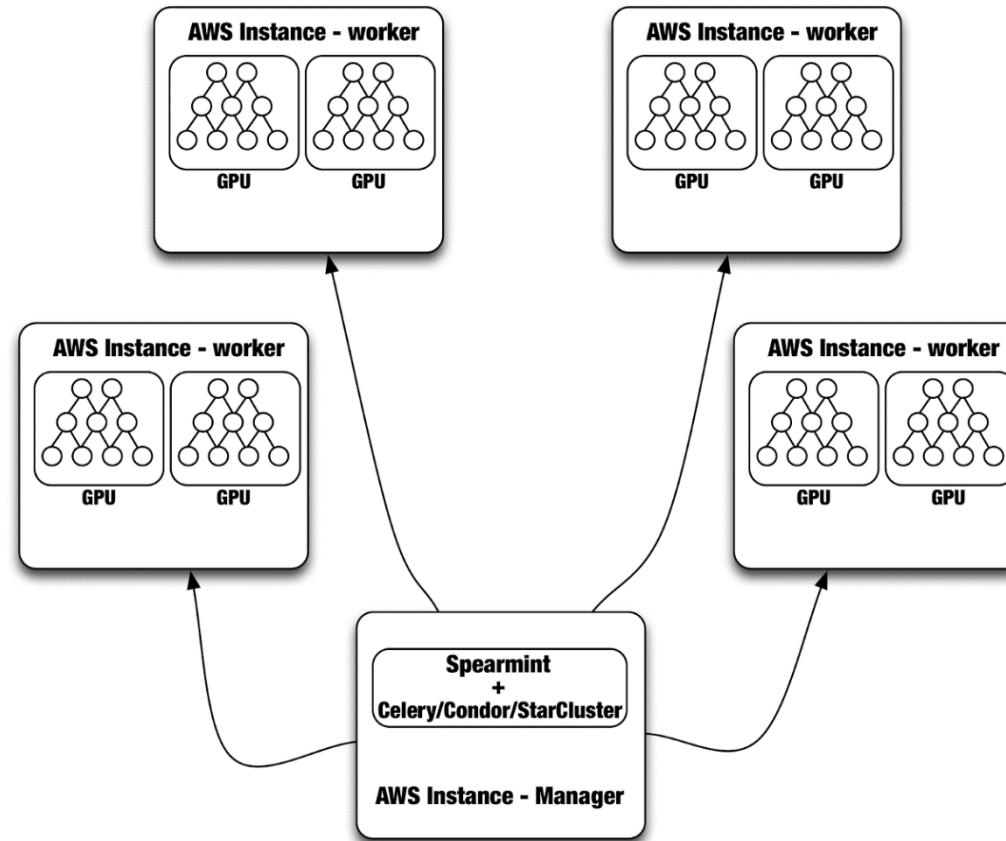
How can larger corporations working on deep learning ensure that their work benefits others within this field?

Corporations are increasingly part of the conversation and now have a much stronger presence at leading conferences in machine learning and deep learning. They are not only providing funding and exhibit booths at the events but are also contributing papers and organizing workshops. Some are even releasing open-source software and systems (such as Google's TensorFlow) which broaden the reach of deep learning to anyone in the world who wants to get involved.

What present or potential future applications excite you most?

I'm excited to see how deep learning can help in recommendation, personalization and search. So far, we've seen deep learning solve tasks that humans are already good at, such as vision, speech recognition, gaming or natural language processing. But humans are notoriously bad at recommending content or items to their friends. We think aspirationally rather than realistically; we recommend a high-brow documentary that sounds intellectual rather than what our friends really would rather watch. I'm excited to see how deep learning can anticipate these bias and preferences and help us each optimize our entertainment, our disposable time and our everyday life in general.

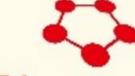
Distributed Neural Networks with GPUs in the AWS Cloud



Vectorflow: a neural network library optimized for sparse data

 Scope 

Lines of code*: *: git ls-files | xargs cat | wc -l

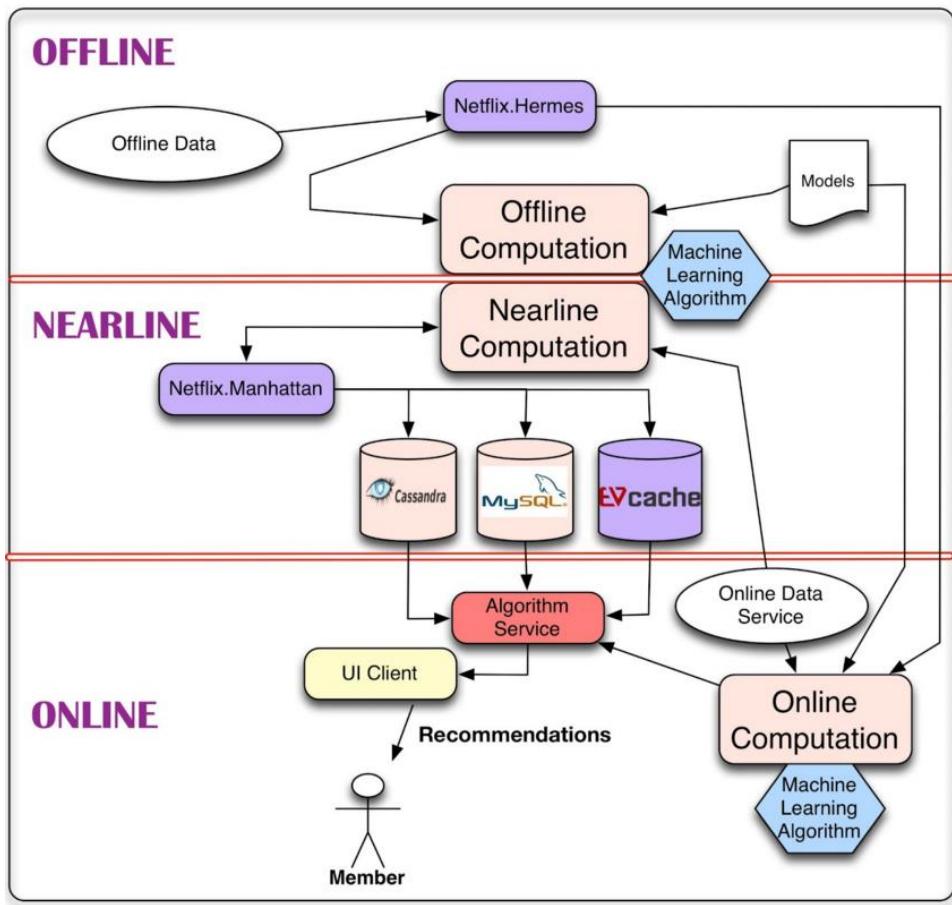
 TensorFlow	 PYTORCH	 mxnet	 Chainer	 vectorflow
1,888k	252k	2,322k	110k	6k

vectorflow

- 0.05 dev (I spend 5% of my time on it)
- offers a minimal DAG with backprop for feed-forward nets
- sparse data as first class citizen
- arbitrary loss function
- extremely fast on CPU
 - 0 memory allocation
 - lock-free inter-core parallelism
 - LLVM intrinsics for dense ops SIMD vectorization

<https://github.com/Netflix/vectorflow>

System diagram for personalized recommendation (2013)



- Offline jobs: **model training and batch computation of intermediate or final results.**
- **Nearline computation** is an intermediate compromise between these two modes in which we can perform online-like computations, but do not require them to be served in real-time.
- **Online computation** responds better to recent events and user interaction, and responds to requests in real-time.

Incomplete list of methods in machine learning for personalization

- Linear regression
- Logistic regression
- Elastic nets
- Singular Value Decomposition
- Restricted Boltzmann Machines
- Markov Chains
- Latent Dirichlet Allocation
- Association Rules
- Gradient Boosted Decision Trees
- Random Forests
- Clustering techniques from the simple k-means to novel graphical approaches
- Matrix factorization

Algorithms selection and testing in Netflix

- Once we test something we want to understand why it failed or succeeded.
- So, how does this work in practice?
- It is a slight variation over the traditional scientific process called A/B testing (or bucket testing):

The process of A/B testing

1. Start with a hypothesis

Algorithm/feature/design X will increase member engagement with our service and *ultimately member retention*

2. Design a test

Develop a solution or prototype. Ideal execution can be 2X as effective as a prototype, but not 10X.

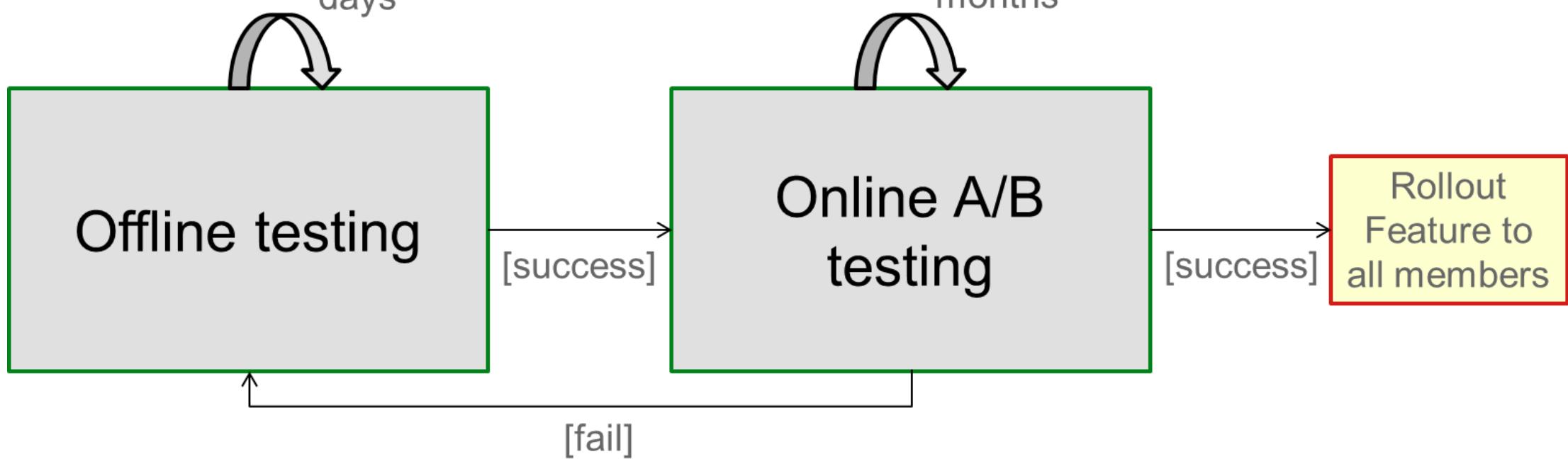
3. Execute the test

4. Let data speak for itself

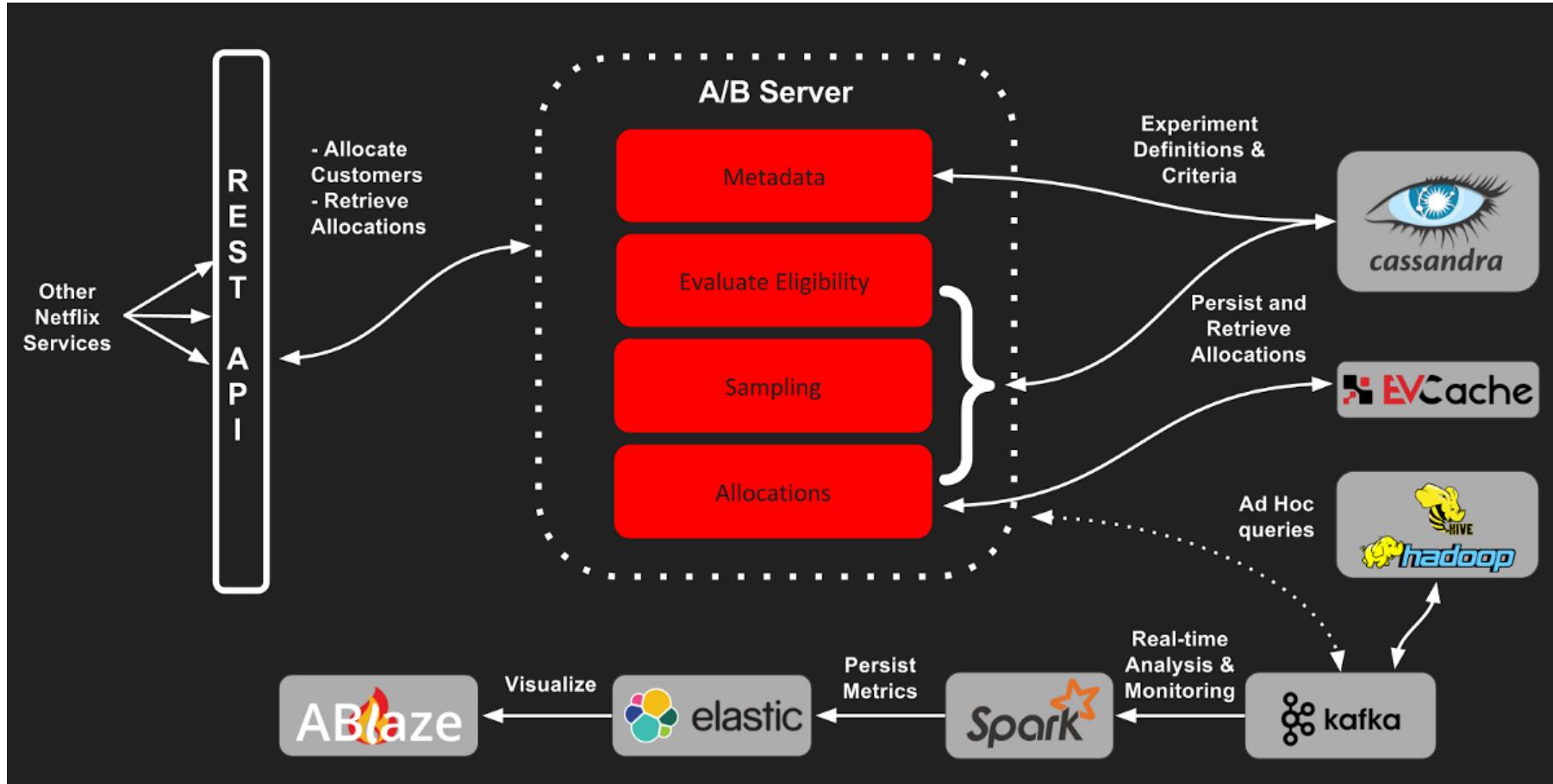
- When executing A/B tests, Netflix track many different metrics.
- Tests usually have thousands of members and anywhere from 2 to 20 cells exploring variations of a base idea.
- The key advantage of A/B tests is that they allow decisions to be data-driven.

Offline/online Testing

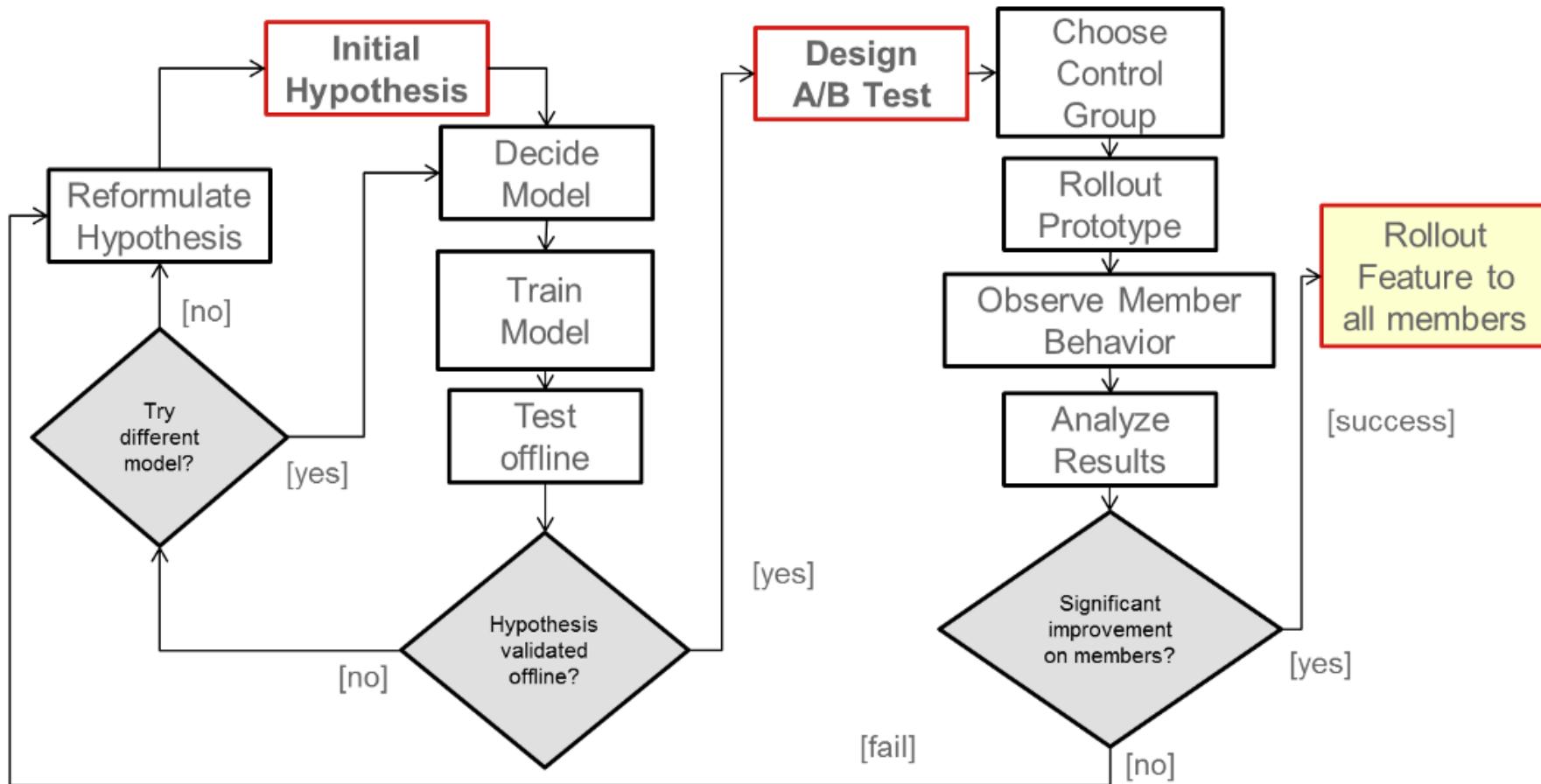
The offline testing cycle is a step to test and optimize algorithms prior to performing online A/B testing



The experimentation platform for A/B testing



The process of rolling out feature



Top10 Marathon

- 10-week effort to quickly test dozens of algorithmic ideas related to improving Top10 row



Conclusion