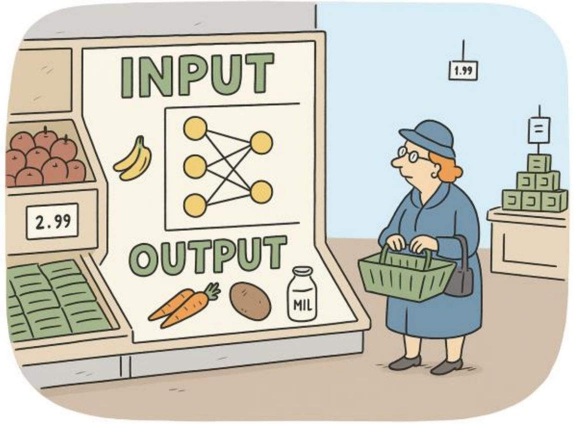
A Decade of Deep Recommender Systems:

Foundations and Trends

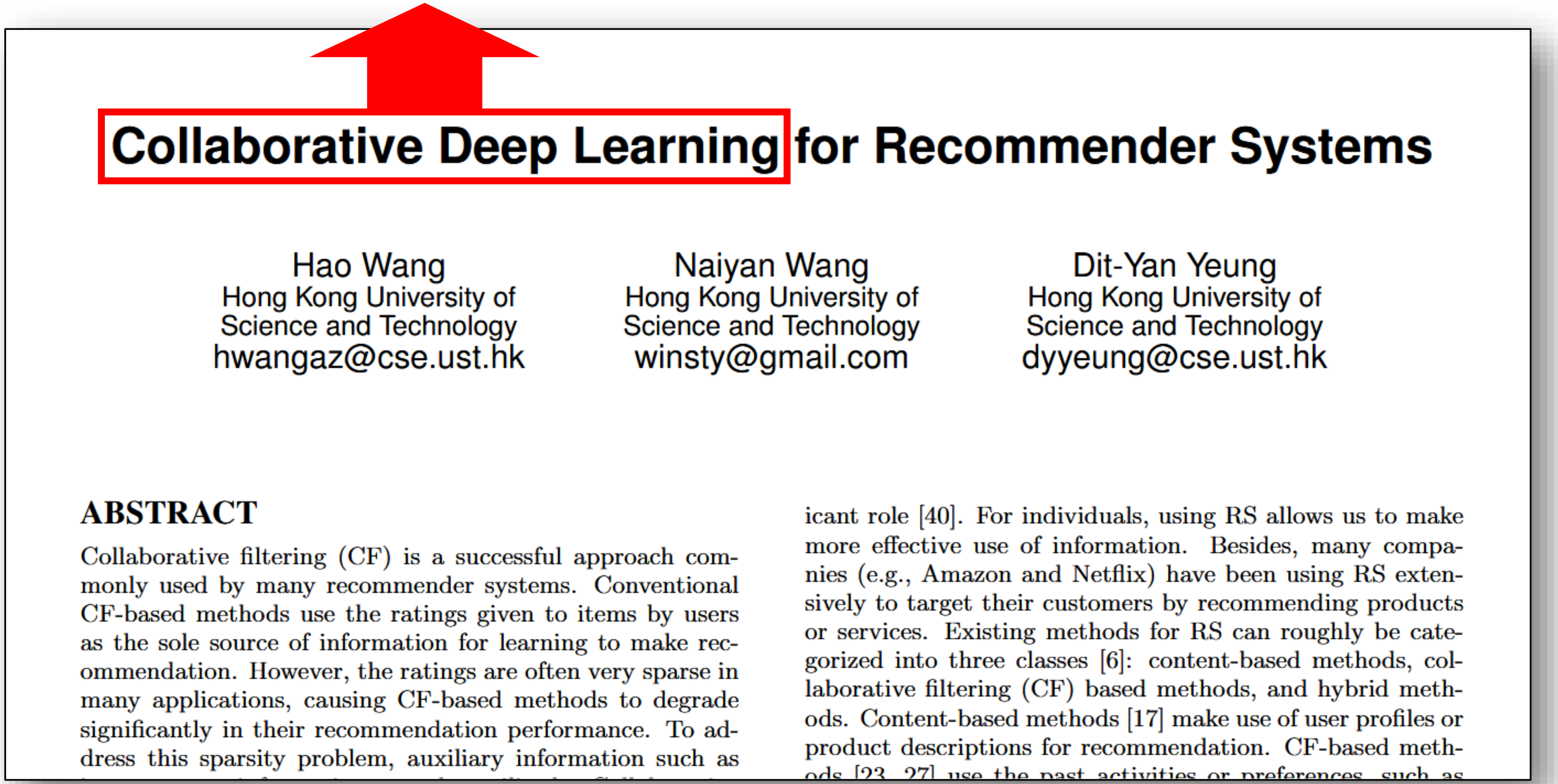
KDD 2025 Test of Time Award Presentation
for the KDD 2015 Paper "Collaborative Deep Learning for Recommender Systems"

Hao Wang Naiyan Wang Dit-Yan Yeung
Rutgers University
www.wanghao.in



Collaborative Deep Learning (CDL)

One of the first deep learning recommender systems



Collaborative Deep Learning for Recommender Systems

Hao Wang
Hong Kong University of Science and Technology
hwangaz@cse.ust.hk

Naiyan Wang
Hong Kong University of Science and Technology
winsty@gmail.com

Dit-Yan Yeung
Hong Kong University of Science and Technology
dyyeung@cse.ust.hk

ABSTRACT

Collaborative filtering (CF) is a successful approach commonly used by many recommender systems. Conventional CF-based methods use the ratings given to items by users as the sole source of information for learning to make recommendation. However, the ratings are often very sparse in many applications, causing CF-based methods to degrade significantly in their recommendation performance. To address this sparsity problem, auxiliary information such as

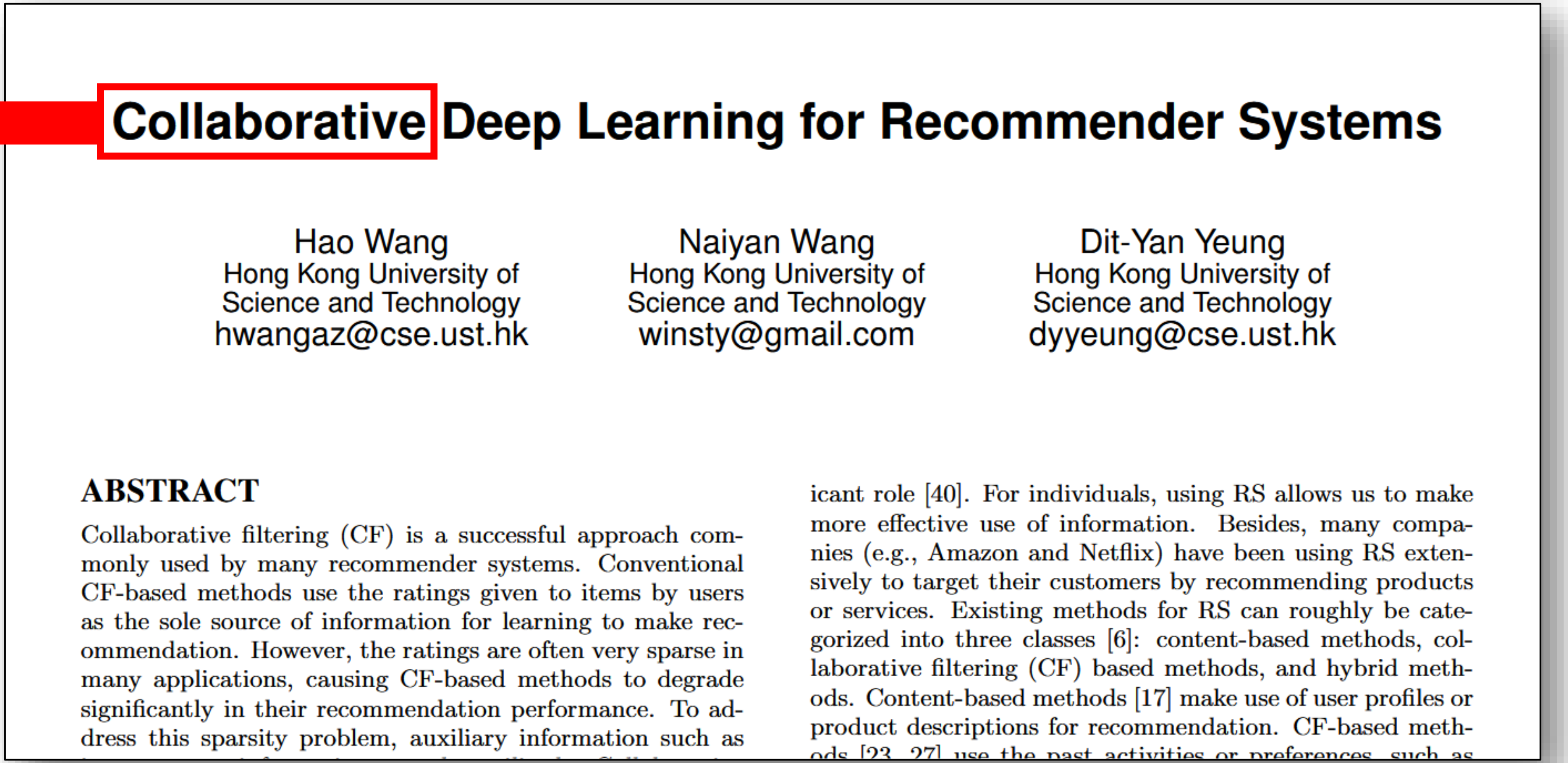
icant role [40]. For individuals, using RS allows us to make more effective use of information. Besides, many companies (e.g., Amazon and Netflix) have been using RS extensively to target their customers by recommending products or services. Existing methods for RS can roughly be categorized into three classes [6]: content-based methods, collaborative filtering (CF) based methods, and hybrid methods. Content-based methods [17] make use of user profiles or product descriptions for recommendation. CF-based methods [23, 27] use the past activities or preferences, such as

[Collaborative deep learning for recommender systems. **W**WY. *ArXiv* 2014, *KDD* 2015]

What is
Collaborative Deep Learning (CDL)?

Collaborative Deep Learning (CDL)

Federated learning?



Collaborative Deep Learning for Recommender Systems

Hao Wang
Hong Kong University of Science and Technology
hwangaz@cse.ust.hk

Naiyan Wang
Hong Kong University of Science and Technology
winsty@gmail.com

Dit-Yan Yeung
Hong Kong University of Science and Technology
dyyeung@cse.ust.hk

ABSTRACT

Collaborative filtering (CF) is a successful approach commonly used by many recommender systems. Conventional CF-based methods use the ratings given to items by users as the sole source of information for learning to make recommendation. However, the ratings are often very sparse in many applications, causing CF-based methods to degrade significantly in their recommendation performance. To address this sparsity problem, auxiliary information such as

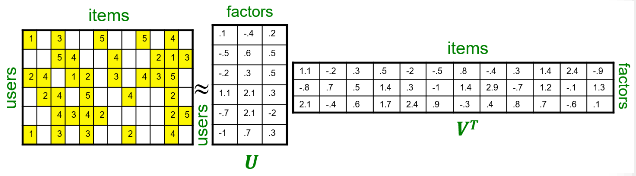
icant role [40]. For individuals, using RS allows us to make more effective use of information. Besides, many companies (e.g., Amazon and Netflix) have been using RS extensively to target their customers by recommending products or services. Existing methods for RS can roughly be categorized into three classes [6]: content-based methods, collaborative filtering (CF) based methods, and hybrid methods. Content-based methods [17] make use of user profiles or product descriptions for recommendation. CF-based methods [23, 27] use the past activities or preferences, such as

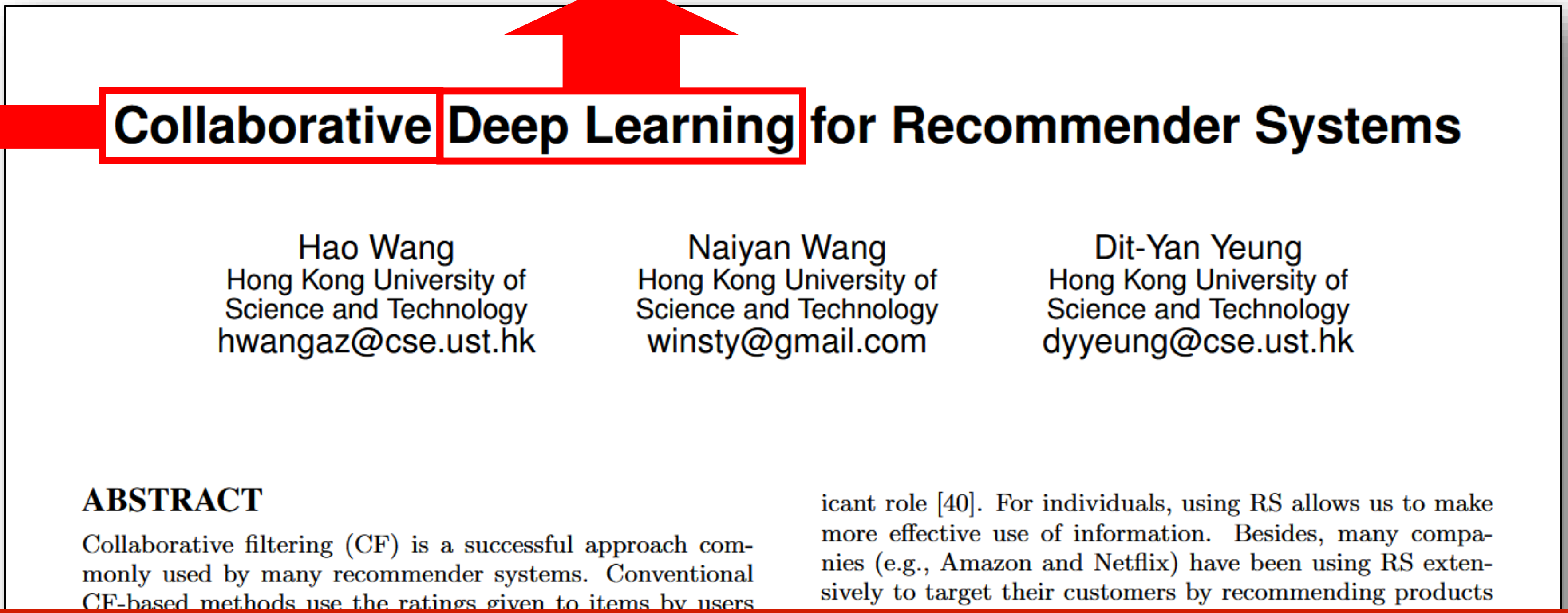
Collaborative Deep Learning (CDL)

End-to-end deep learning of compact user & item features

Collaborative filtering

Low-rank matrix factorization





Collaborative Deep Learning for Recommender Systems

Hao Wang
Hong Kong University of Science and Technology
hwangaz@cse.ust.hk

Naiyan Wang
Hong Kong University of Science and Technology
winsty@gmail.com

Dit-Yan Yeung
Hong Kong University of Science and Technology
dyyeung@cse.ust.hk

ABSTRACT

Collaborative filtering (CF) is a successful approach commonly used by many recommender systems. Conventional CF-based methods use the ratings given to items by users

icant role [40]. For individuals, using RS allows us to make more effective use of information. Besides, many companies (e.g., Amazon and Netflix) have been using RS extensively to target their customers by recommending products

CDL jointly performs
collaborative filtering and deep learning of user & item features

The Problem We Saw in 2014~2015

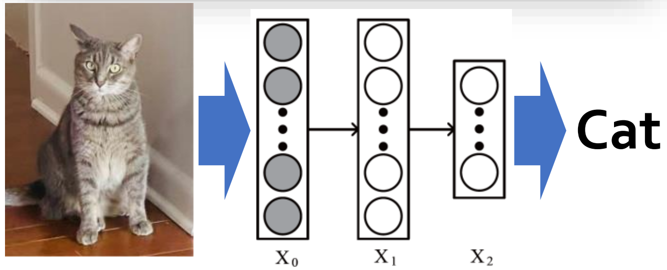
- **2 years** after the **ImageNet** moment
    - In 2012, AlexNet cuts the ImageNet error rate by half, starting the deep learning revolution


The Economist

From not working to neural networking

The artificial-intelligence boom is based on an old idea, but with a modern twist

- Existing deep learning methods are **limited**
    - Only work for classification and regression
    - Not clear how to perform recommendation



Rating Matrix:

movie \ user	1	2	3	4	5
1	✓	?	?	?	?
2	✓	?	?	✓	?
3	?	?	✓	?	?
4	?	✓	?	?	✓
5	✓	?	?	?	?

Movie videos, descriptions, etc.

Matrix Completion:

Observed Preferences: ✓

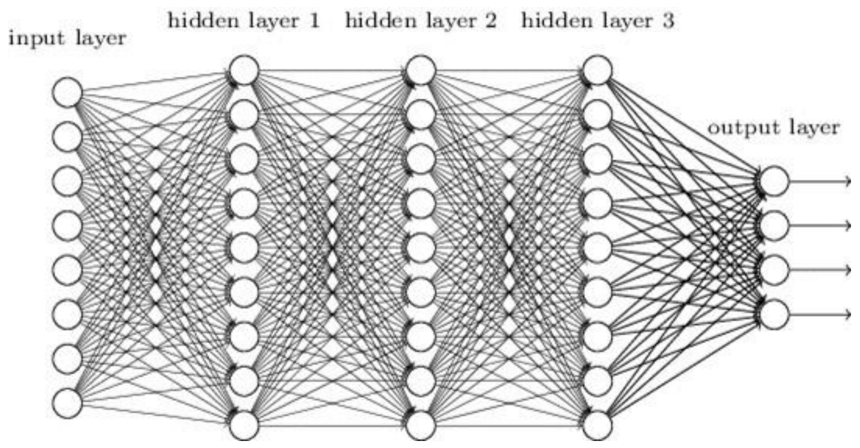
To Predict: ?

Beginning of 2014: Fundamental Limitation of Early Deep Learning

Perception

Deep Learning

High dimensional input:
Text, Images, Videos



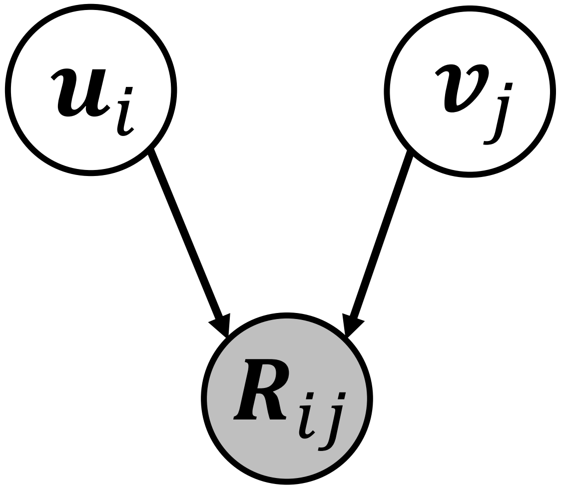
[ImageNet classification with deep convolutional neural networks. KSH. *NIPS* 2012]

Inference

Graphical Models

User i's latent vector (embedding)

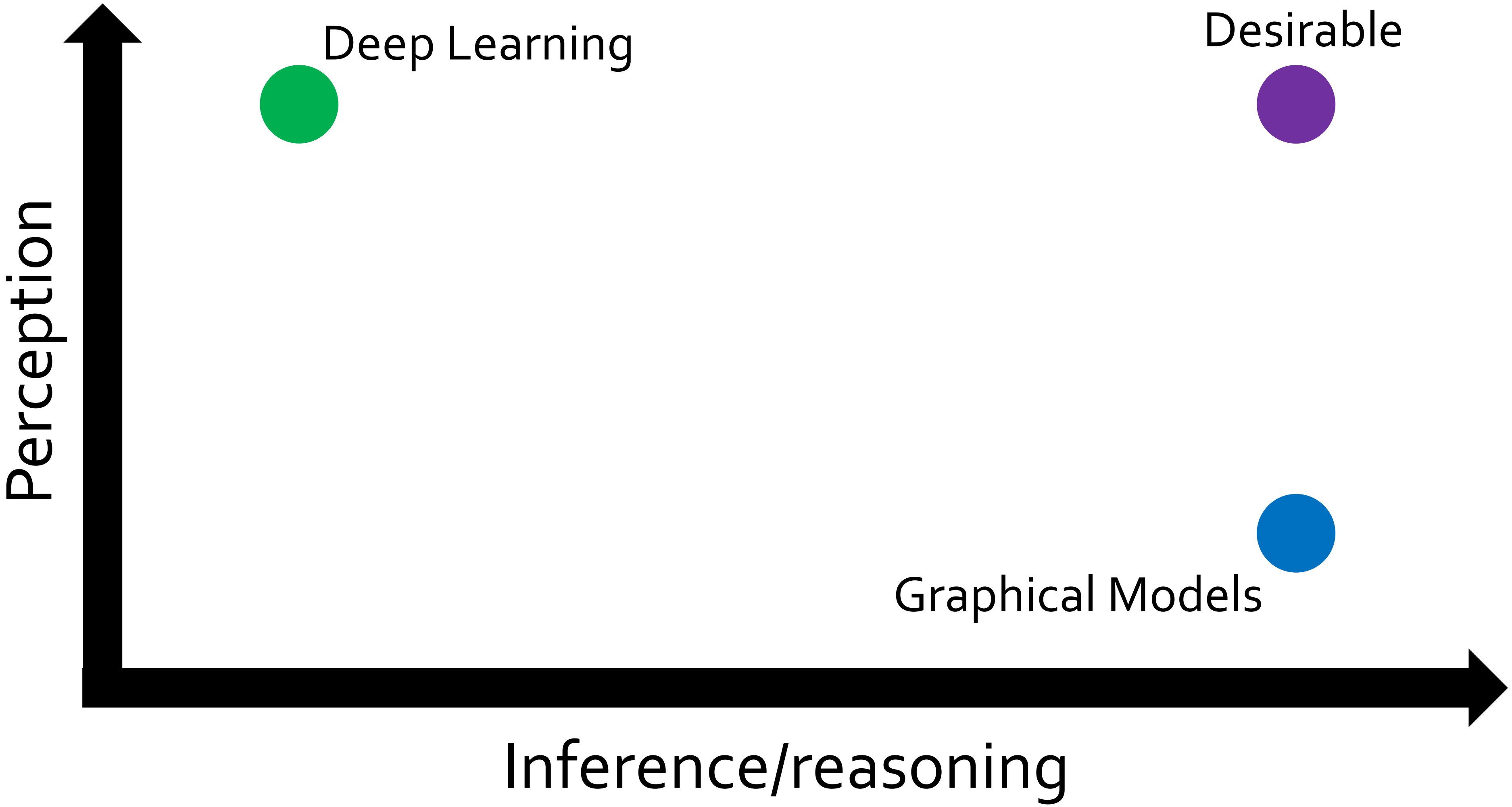
Item j's latent vector (embedding)



Rating that user i gives item j

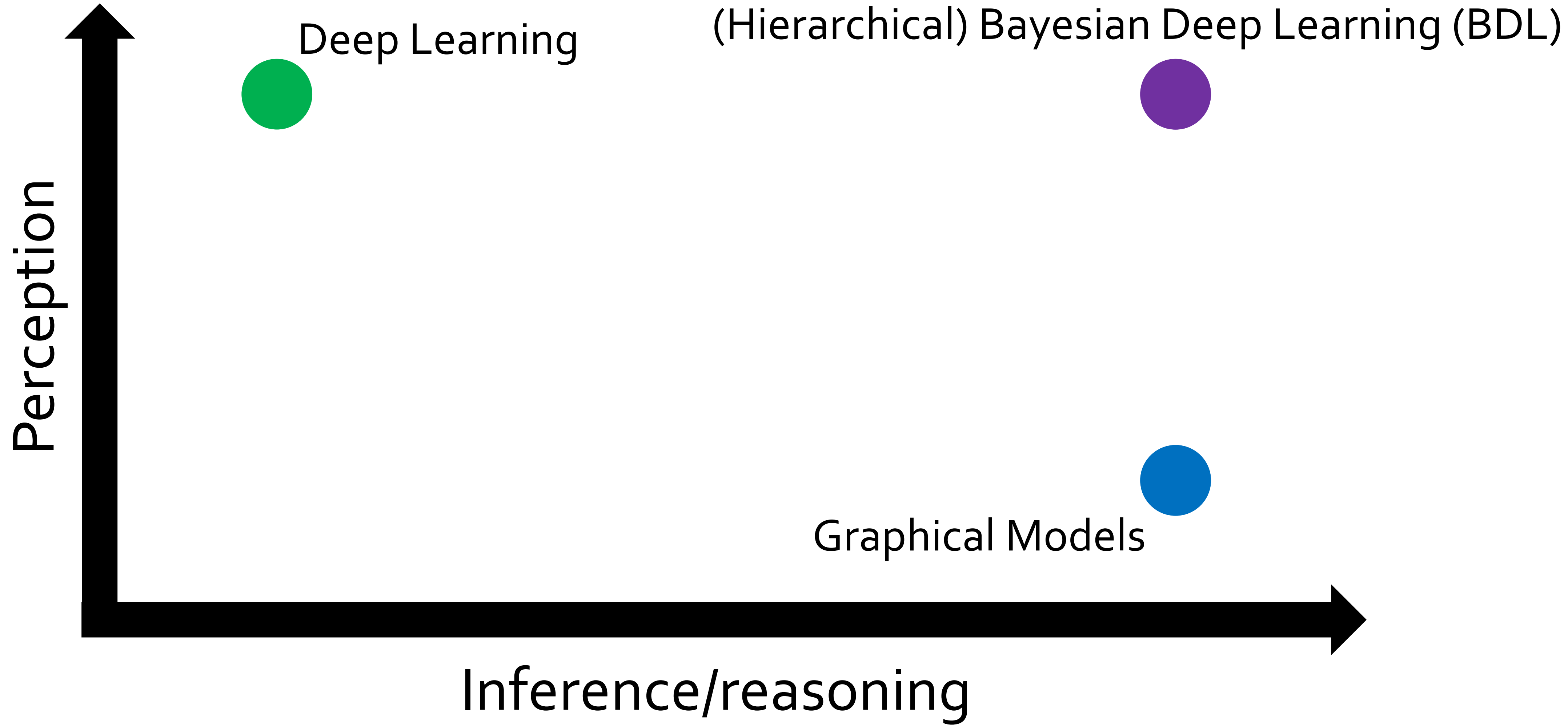
[Probabilistic matrix factorization. SM. *NIPS* 2007]
[Collaborative topic modeling for recommending scientific articles. WB. *KDD* 2011]

Best of Both Worlds?



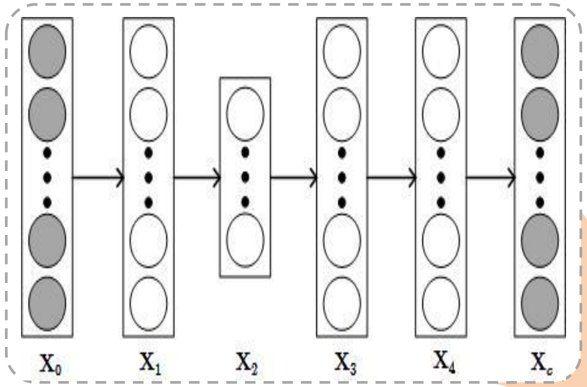
[Collaborative deep learning for recommender systems. **W**WY. *ArXiv* 2014, *KDD* 2015]
[Towards Bayesian deep learning: A framework and some existing methods. **W**Y. *TKDE* 2016.]
[A survey on Bayesian deep learning. **W**Y. *ACM Computing Surveys* 2020.]

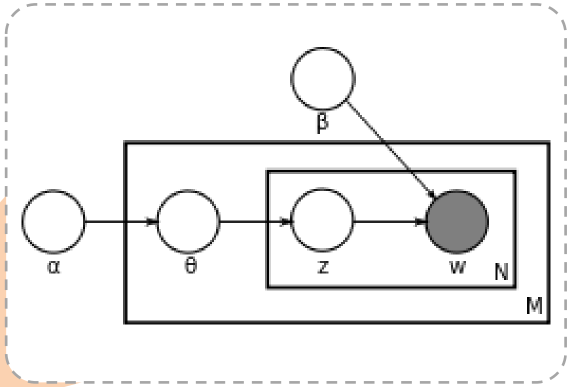
Spring of 2014: (Hierarchical) Bayesian Deep Learning



[Collaborative deep learning for recommender systems. **W**WY. *ArXiv* 2014, *KDD* 2015]
[Towards Bayesian deep learning: A framework and some existing methods. **W**Y. *TKDE* 2016.]
[A survey on Bayesian deep learning. **W**Y. *ACM Computing Surveys* 2020.]

Bayesian Deep Learning (BDL)





Deep component

Probabilistic DL models
(Perception)

Graphical component

Graphical models
(Inference and reasoning)

Bayesian deep learning (BDL)

[Towards Bayesian deep learning: A framework and some existing methods. **W**Y. *TKDE* 2016.]
[A survey on Bayesian deep learning. **W**Y. *ACM Computing Surveys* 2020.]

Example: Movie Recommender Systems

Deep component

Uses video, plot, actors, etc.
Content understanding

movie \ user	1	2	3	4	5
1	✓	?	?	?	?
2	✓	?	?	✓	?
3	?	?	✓	?	?
4	?	✓	?	?	✓
5	✓	?	?	?	?

Graphical component

Uses preferences, similarities
Recommendation

Bayesian deep learning (BDL)

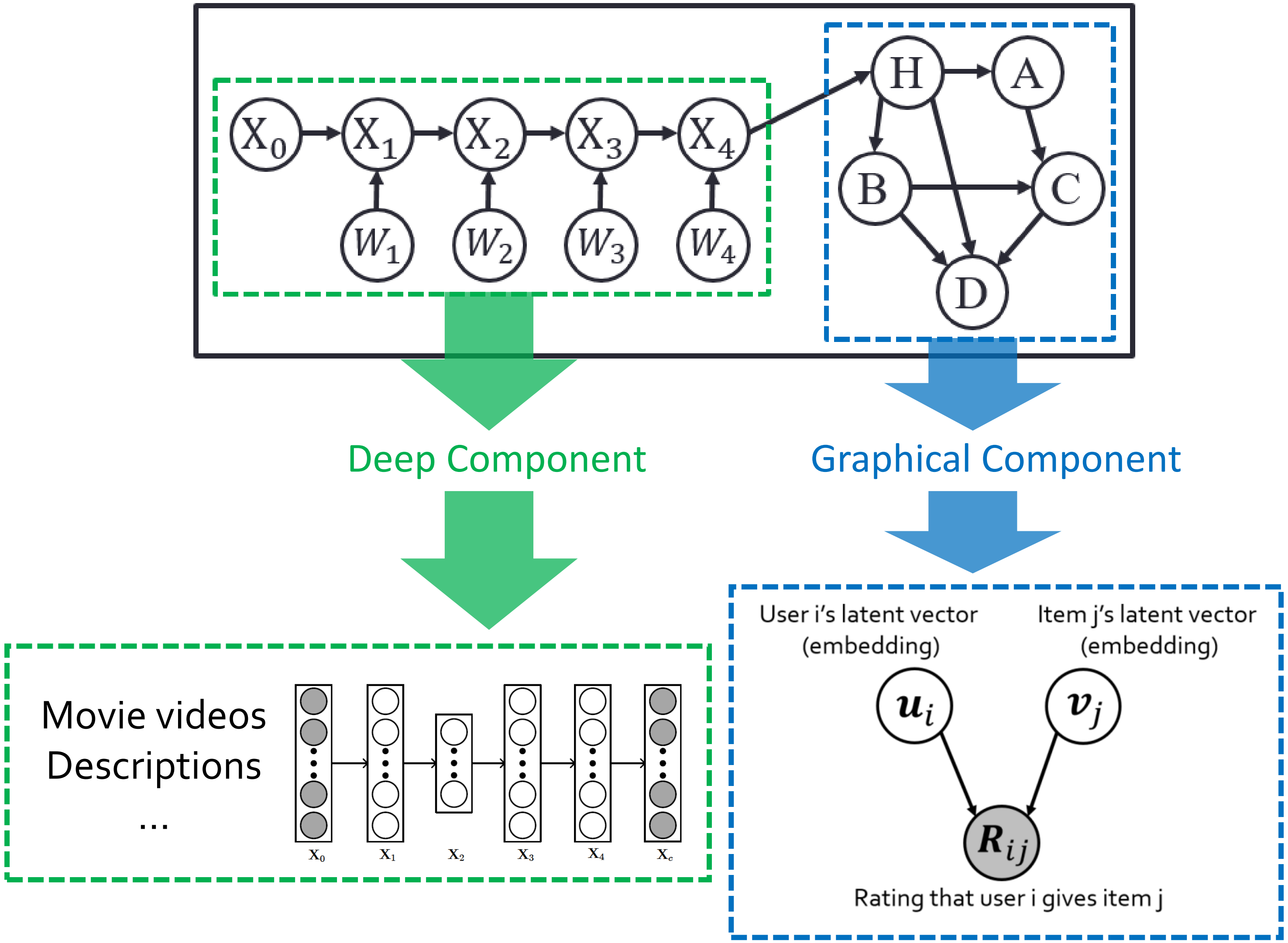
[Collaborative deep learning for recommender systems. **W**WY. *KDD* 2015]
[Collaborative recurrent autoencoder. **W**SY. *NIPS* 2016a]
[Zero-shot recommender systems. DDW**W**. *ICLR-W* 2022]

Amazon Personalize

Partially deployed in "Amazon Personalize"

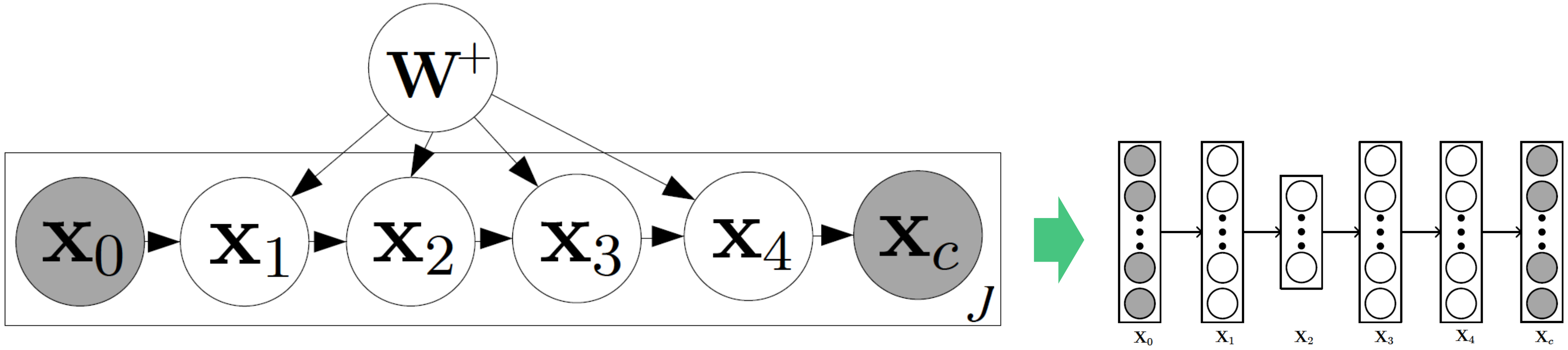
BDL: A Principled Probabilistic Framework

In the context of Collaborative Deep Learning (CDL)



[Towards Bayesian deep learning: A framework and some existing methods. **W**Y. *TKDE* 2016]

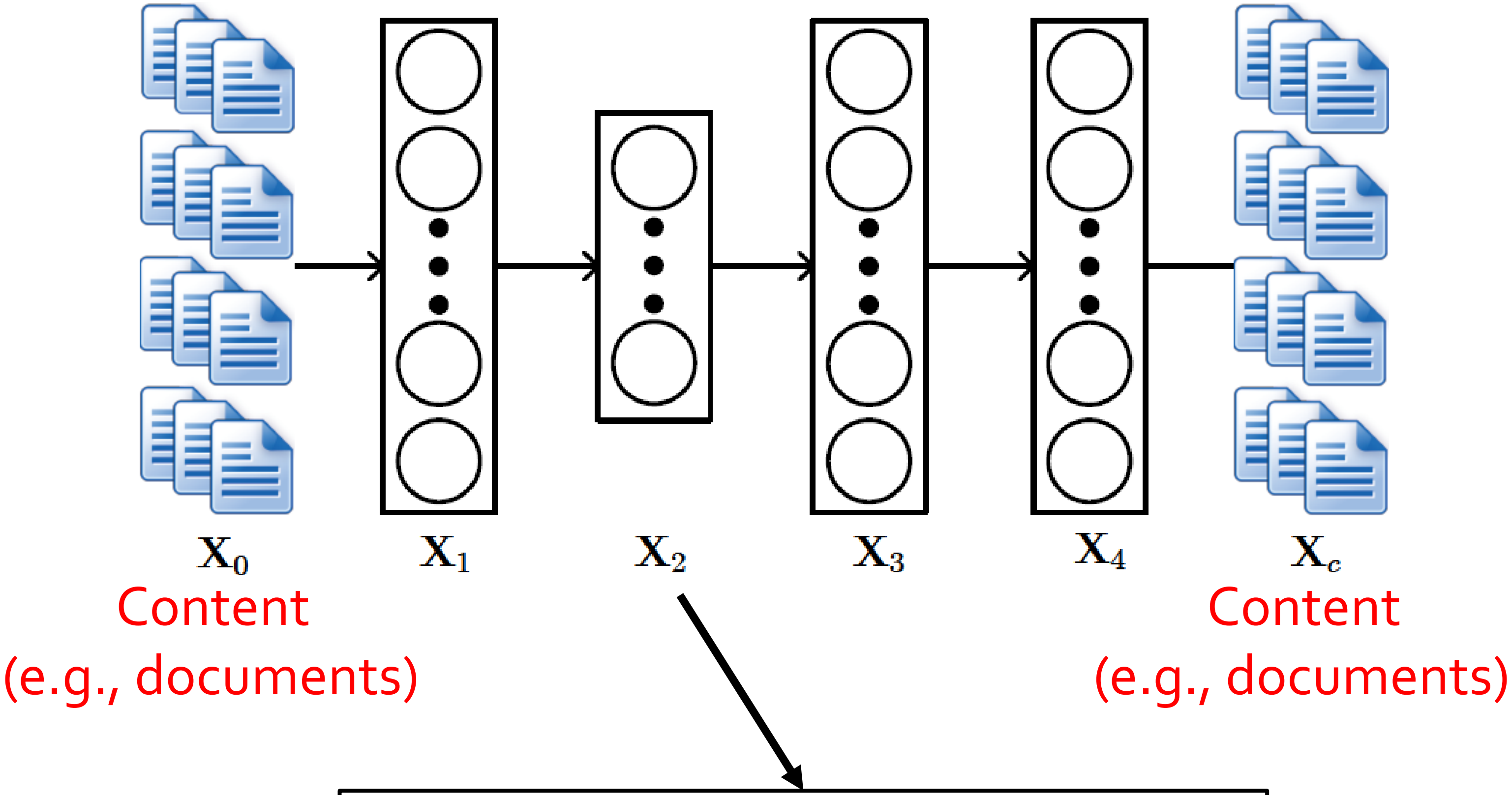
CDL (Step 1 of 4): Start from Middle-Layer Representation



Start from probabilistic autoencoder

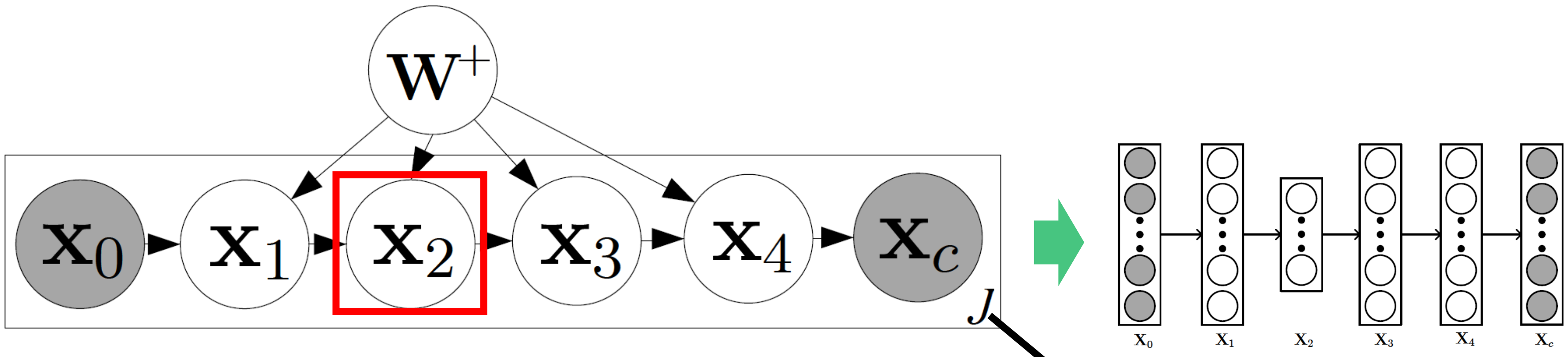
X_2: Middle-layer representation

Autoencoder (AE)



X_2: Middle-layer representation

CDL (Step 1 of 4): Start from Middle-Layer Representation



Start from probabilistic autoencoder

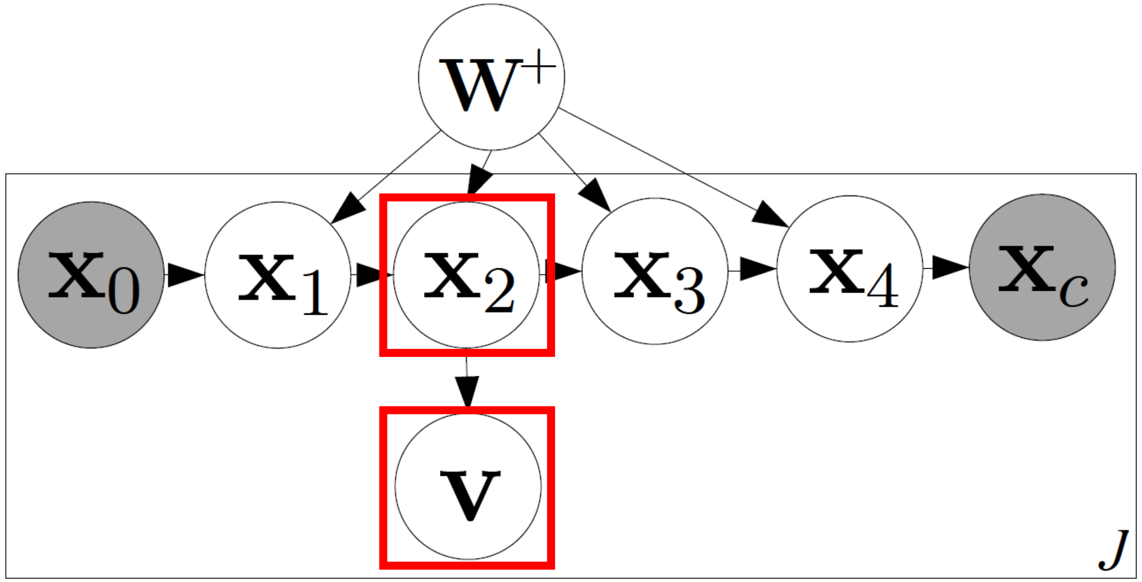
Number of items (e.g., movies)

X_2: Middle-layer representation

Observed variables (**given**)

Latent variables & parameters **to learn**

CDL (Step 2 of 4): Generate Item j's Latent Vector $\boldsymbol{v_j}$



Generate the **latent vector for item j** from $\boldsymbol{X_2}$:

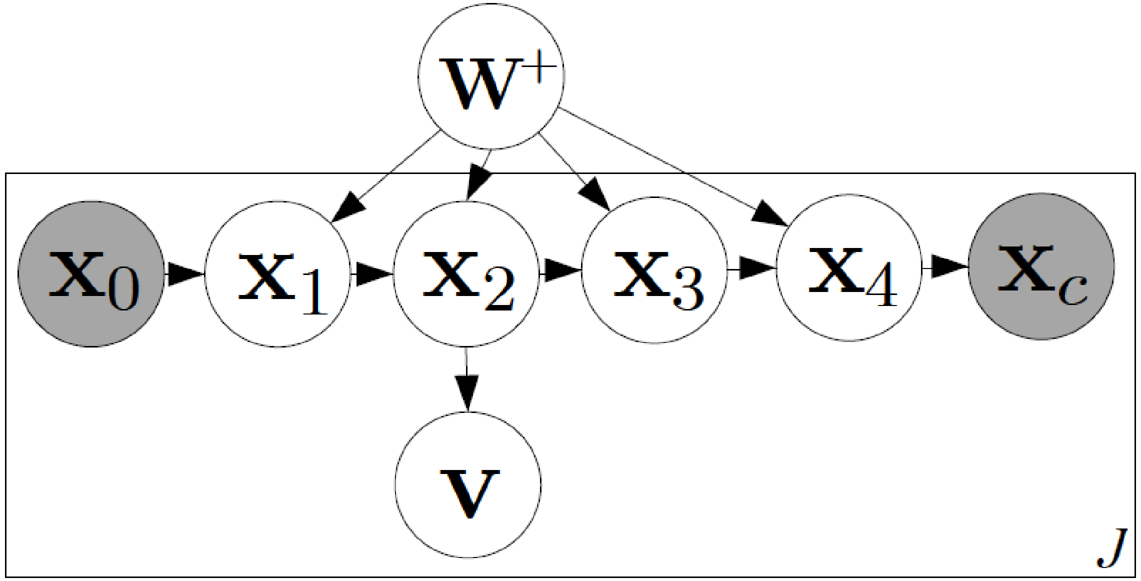
$$\mathbf{v}_j \sim \mathcal{N}(\mathbf{X}_2, \lambda_v^{-1}\mathbf{I})$$

$\boldsymbol{X_2}$: Item embedding based only on **content** (e.g., movie descriptions)

$\boldsymbol{v_j}$: Item embedding based on both **content** and **user preferences**

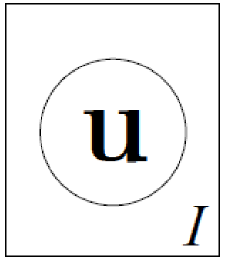
More relevant to recommendation

CDL (Step 3 of 4): Generate User i's Latent Vector $\boldsymbol{u_i}$

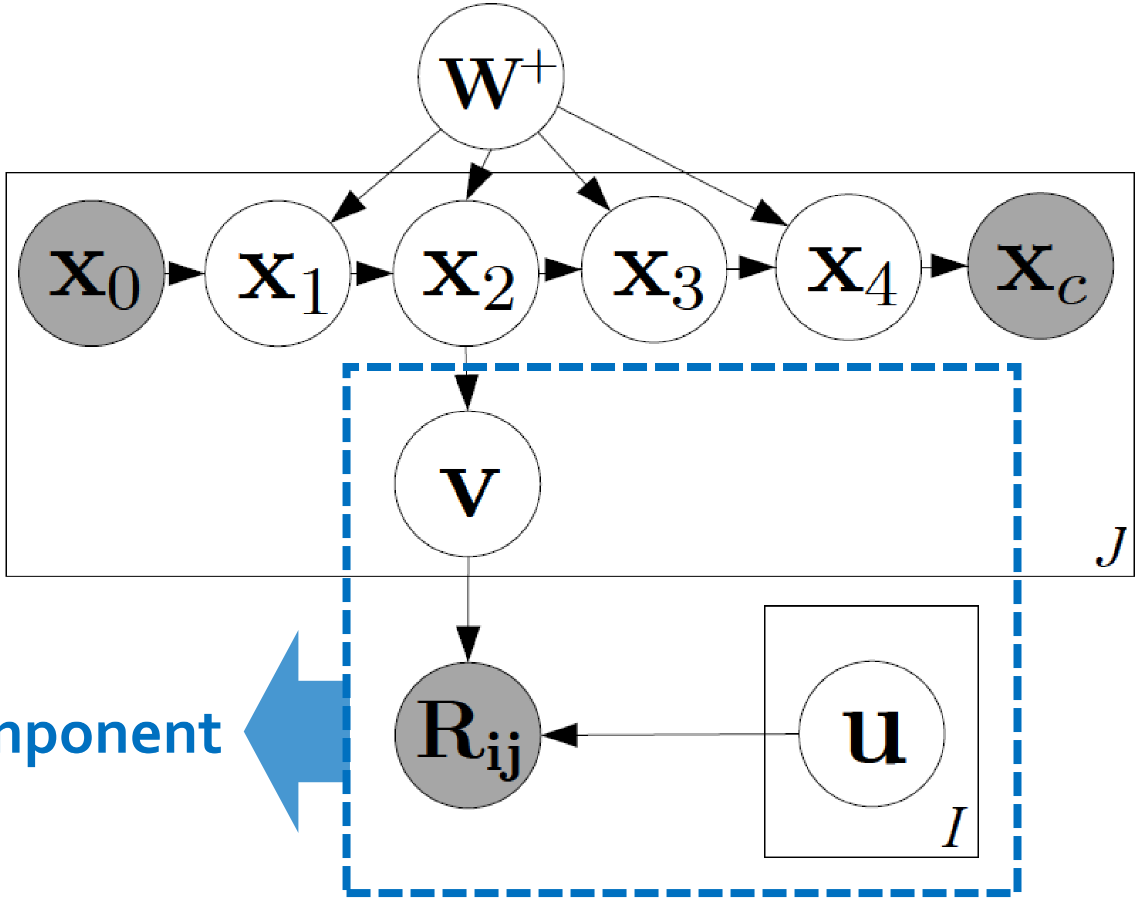


Generate the **latent vector for user i**:

$$\mathbf{u}_i \sim \mathcal{N}(\mathbf{0}, \lambda_u^{-1}\mathbf{I})$$



CDL (Step 4 of 4): Generate Ratings $\boldsymbol{R_{ij}}$ from $\boldsymbol{u_i^T v_j}$



Graphical Component

Given:

- **Movie descriptions** $\mathbf{x}_0$ $\mathbf{x}_c$
- **Observed Ratings** $\mathbf{R}_{ij}$

Infer:

- **User vector** $\mathbf{u}$
- **Item vector** $\mathbf{v}$

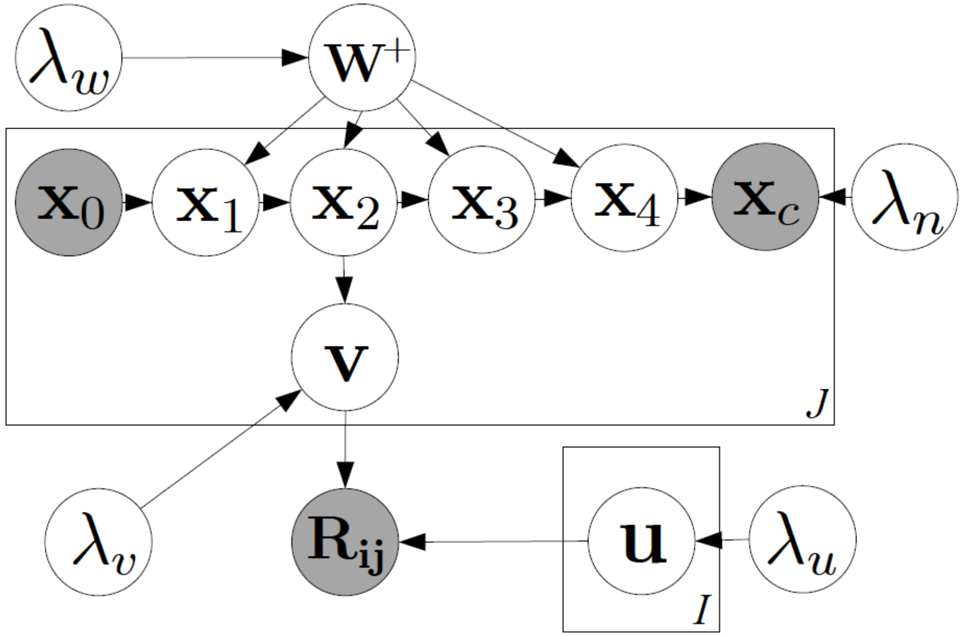
Predict:

- **Unknown ratings** $\mathbf{R}$

Generate the **rating user i gives item j**: $\mathbf{R}_{ij} \sim \mathcal{N}(\mathbf{u}_i^T \mathbf{v}_j, \lambda_r^{-1})$

Overview: Collaborative Deep Learning (CDL)

Graphical model:

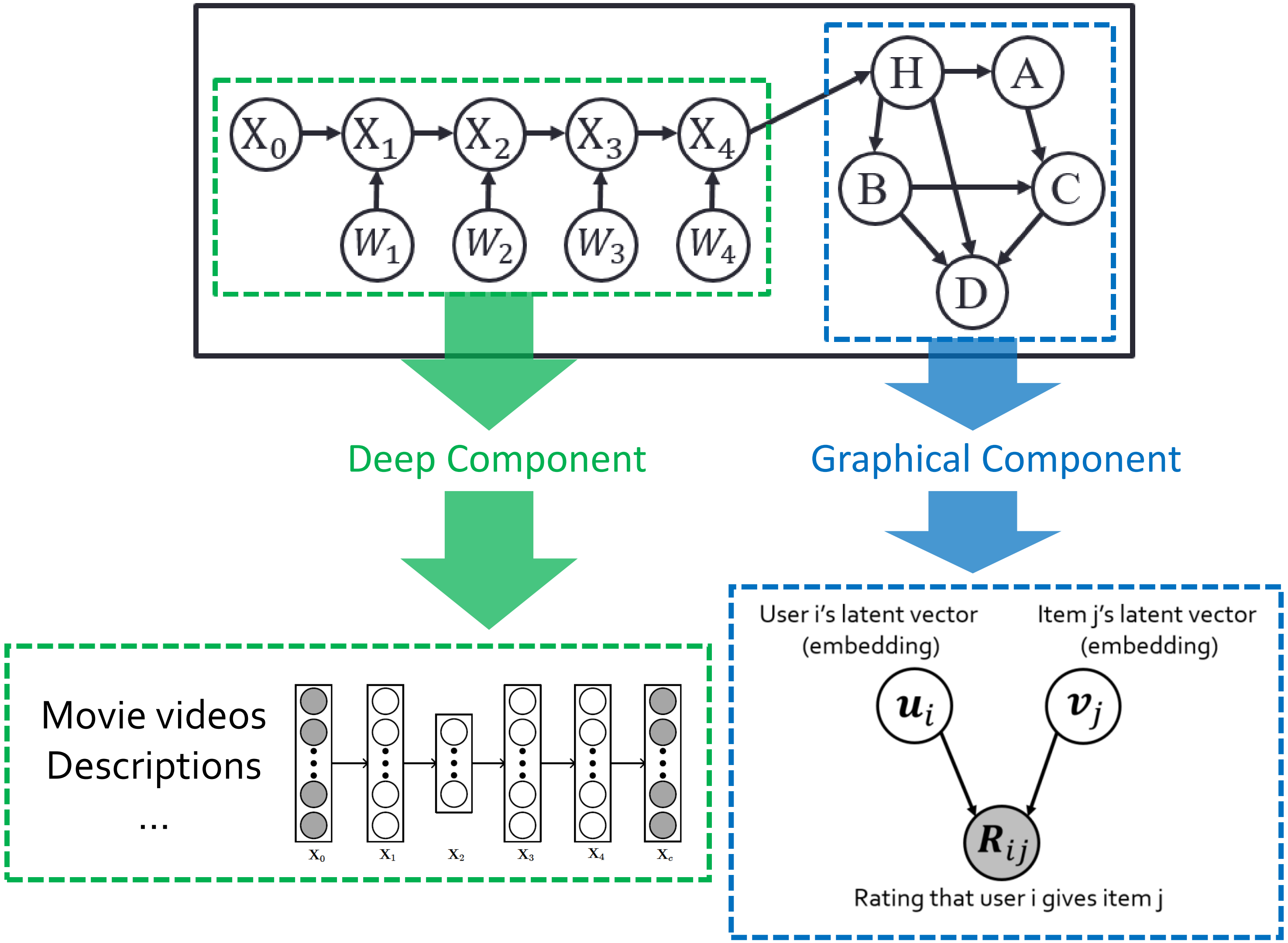


$\lambda_w, \lambda_n, \lambda_v, \lambda_u$:

hyperparameters to control the **variance** of Gaussian distributions

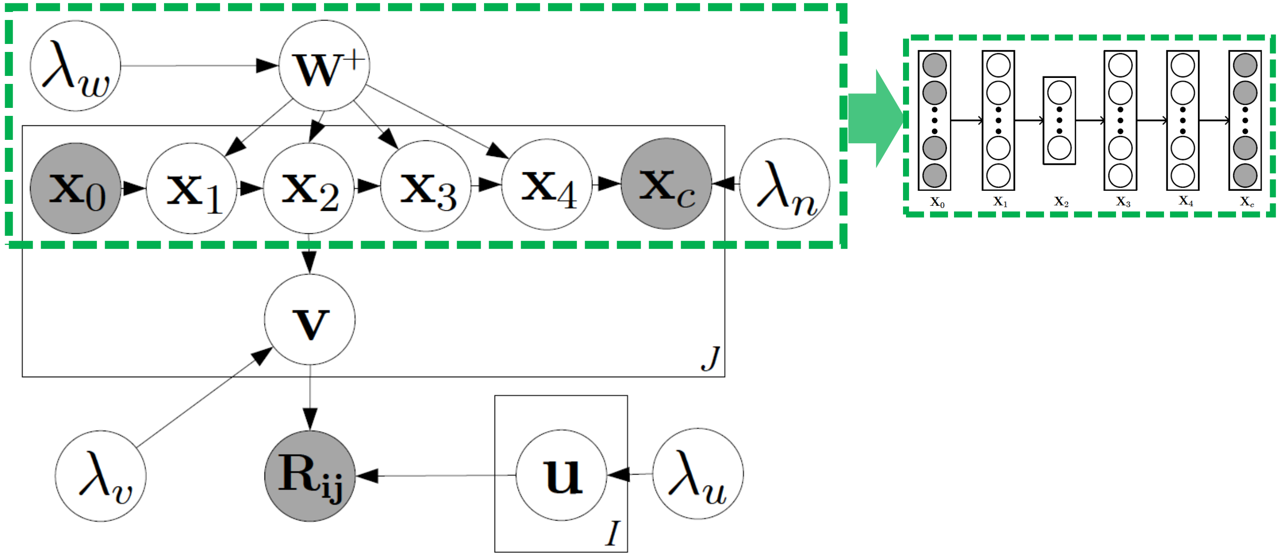
BDL: A Principled Probabilistic Framework (Recap)

In the context of CDL



[Towards Bayesian deep learning: A framework and some existing methods. **W**Y. *TKDE* 2016]

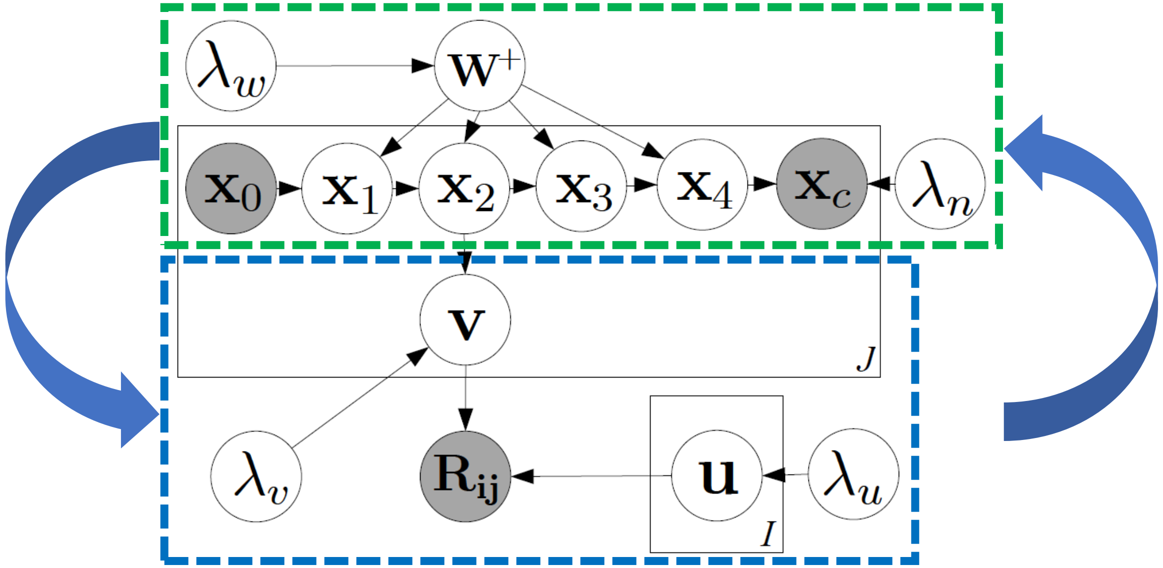
Graphical Model of CDL with Two Components



Collaborative Deep Learning

Graphical Model of CDL with Two Components

Boost recommendation accuracy



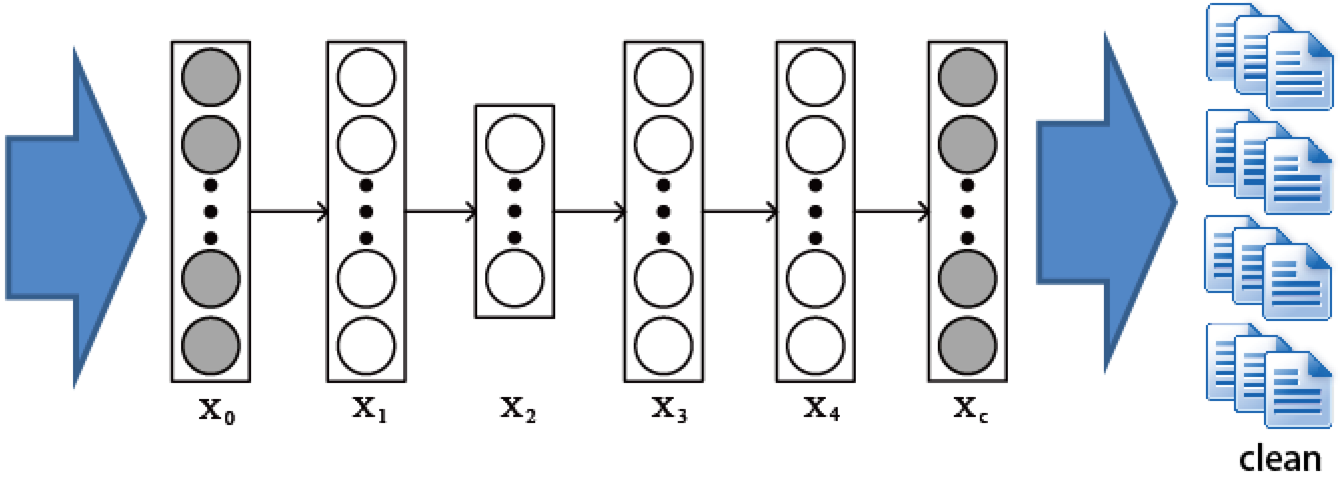
Boost representation learning

Collaborative Deep Learning

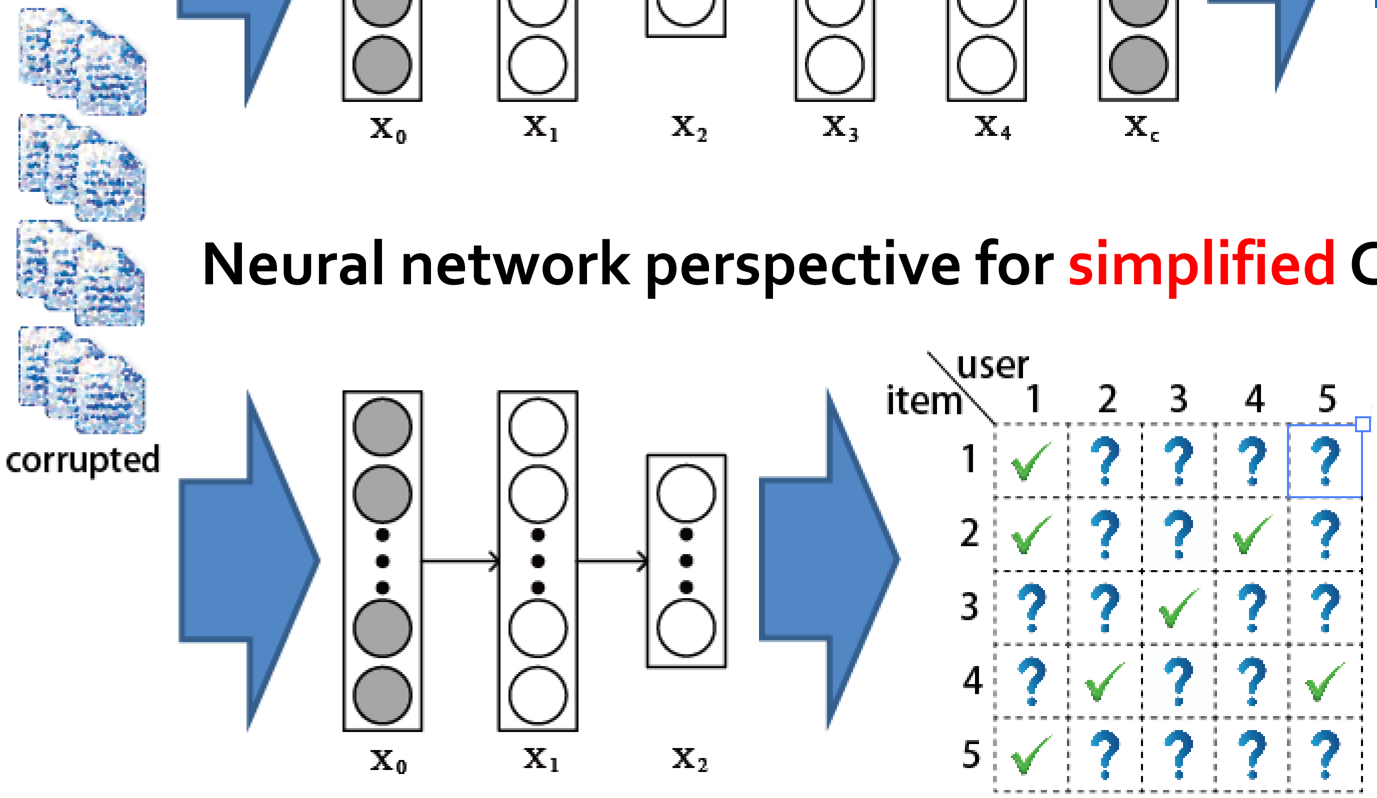
Trained end-to-end

- **Boost each other's performance**
- More powerful representation
- Infer missing ratings from content
- Infer missing content from ratings

Collaborative Deep Learning (CDL)

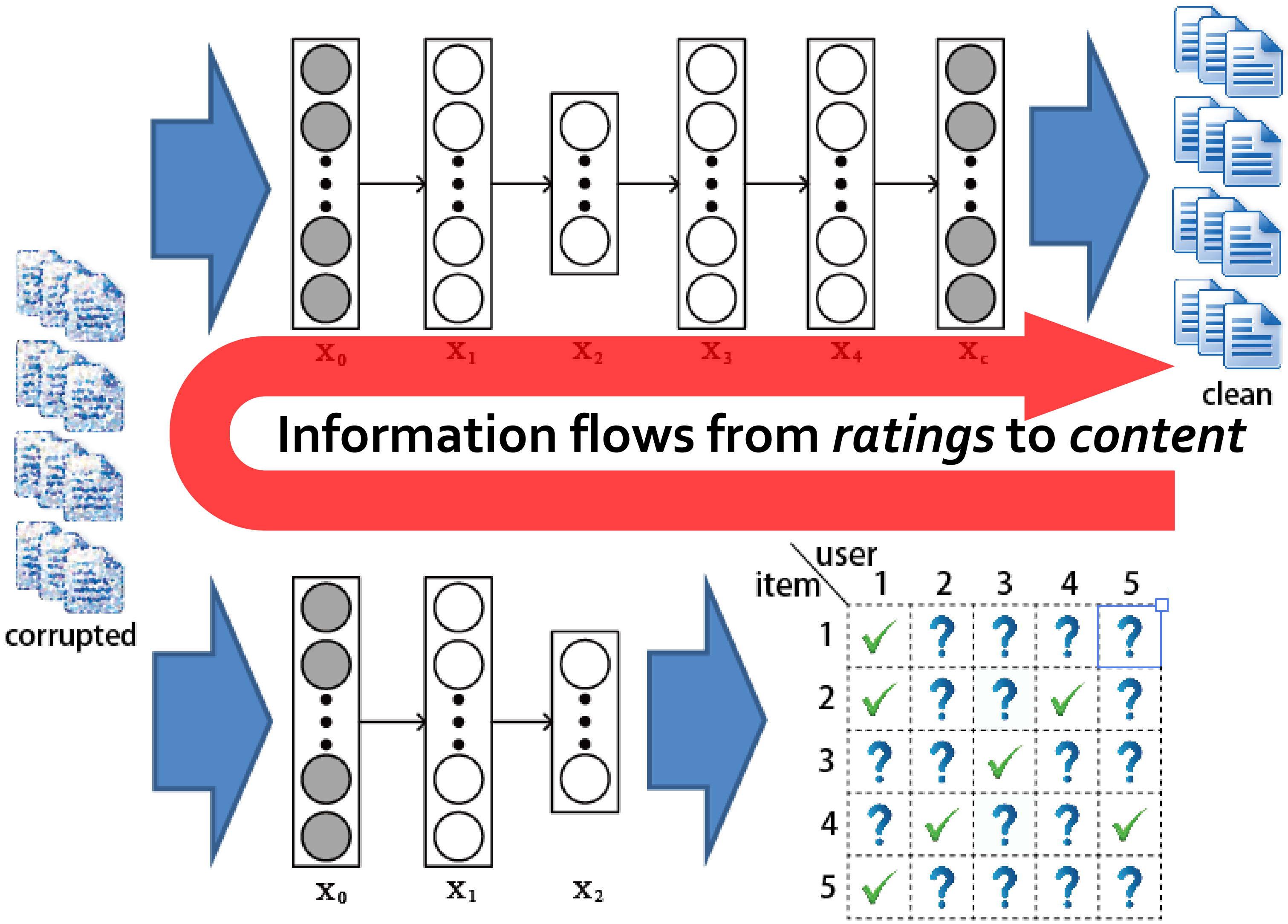


Neural network perspective for simplified CDL

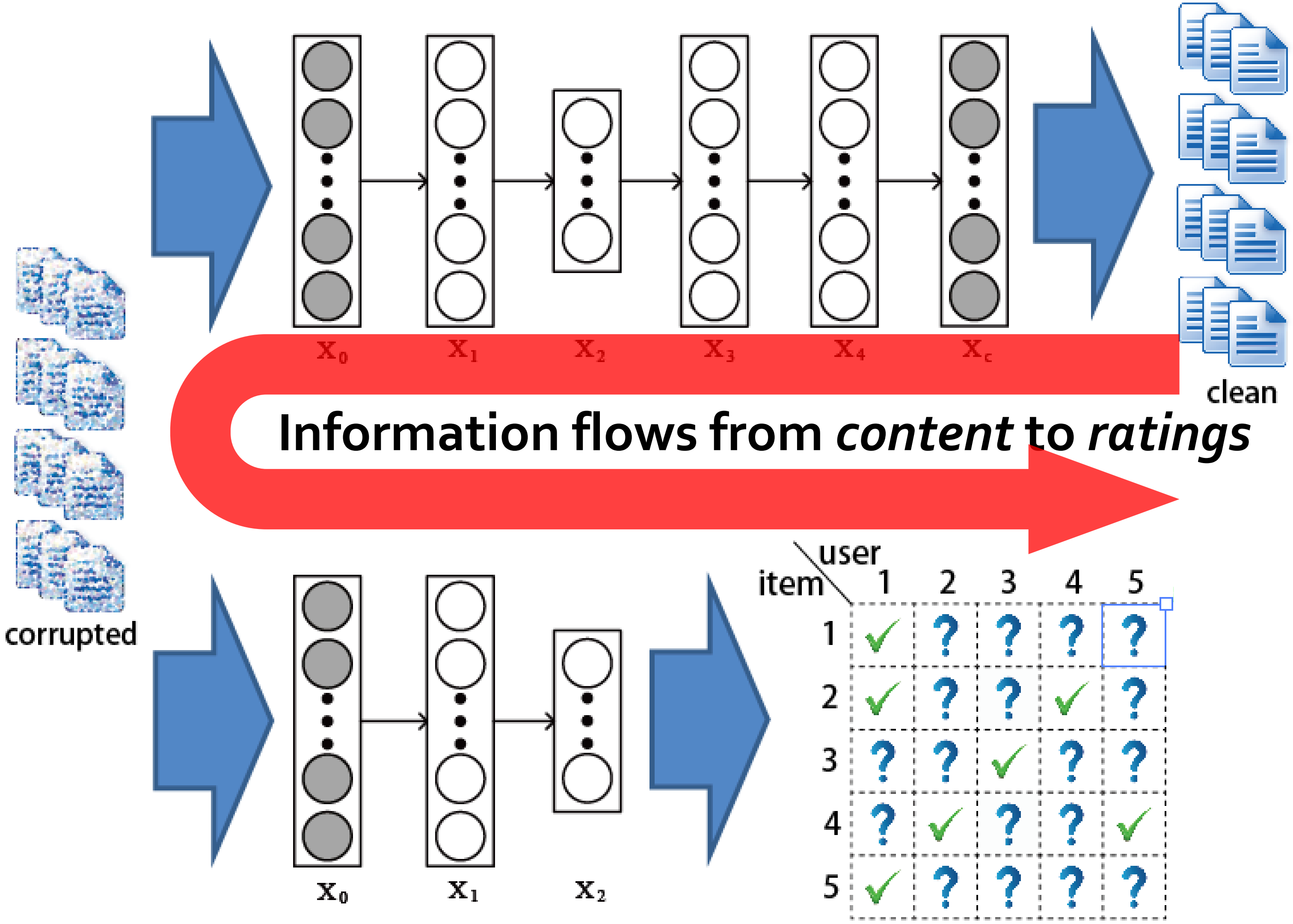


2015 Slides

Collaborative Deep Learning (CDL)

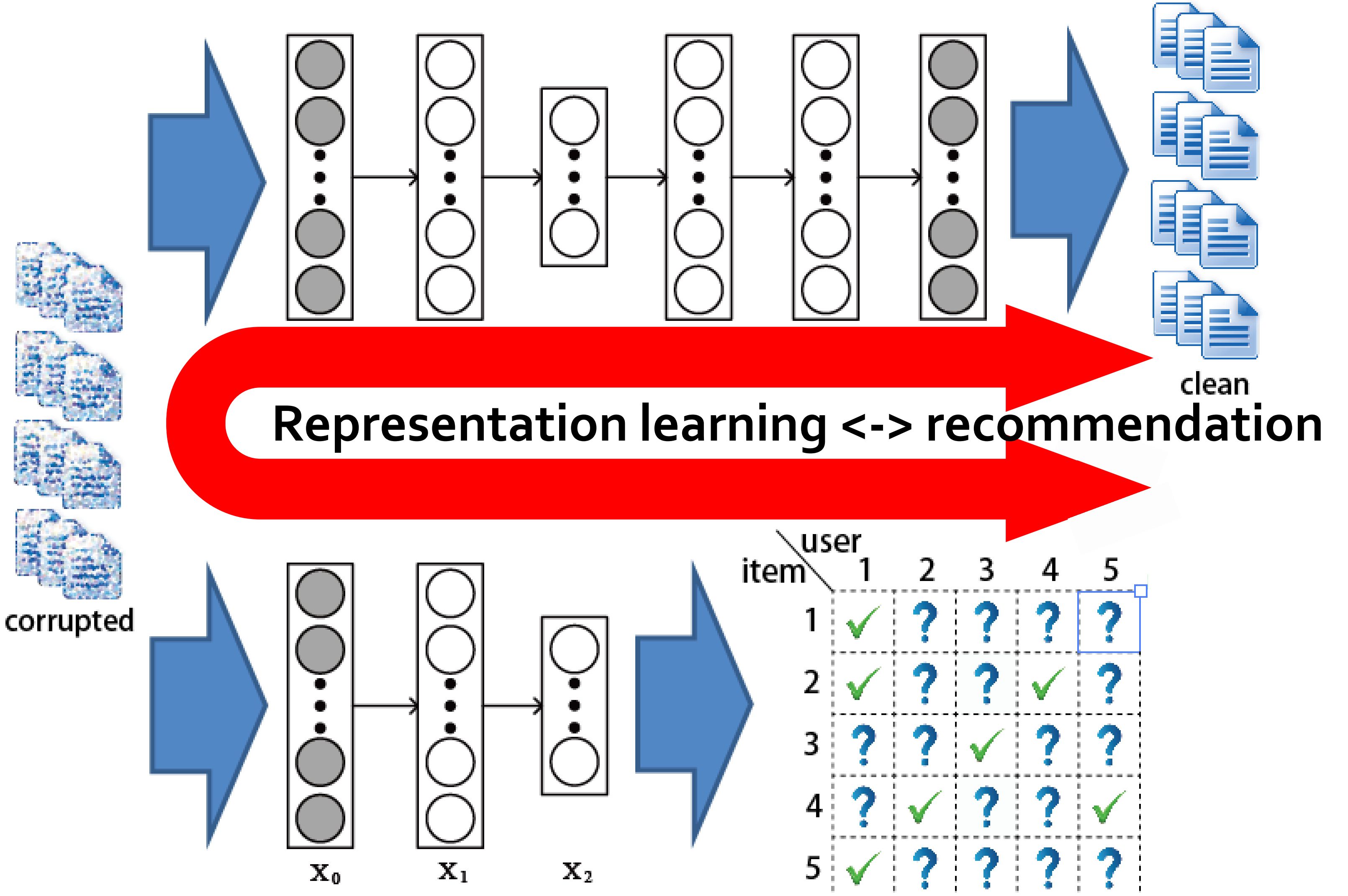


Collaborative Deep Learning (CDL)



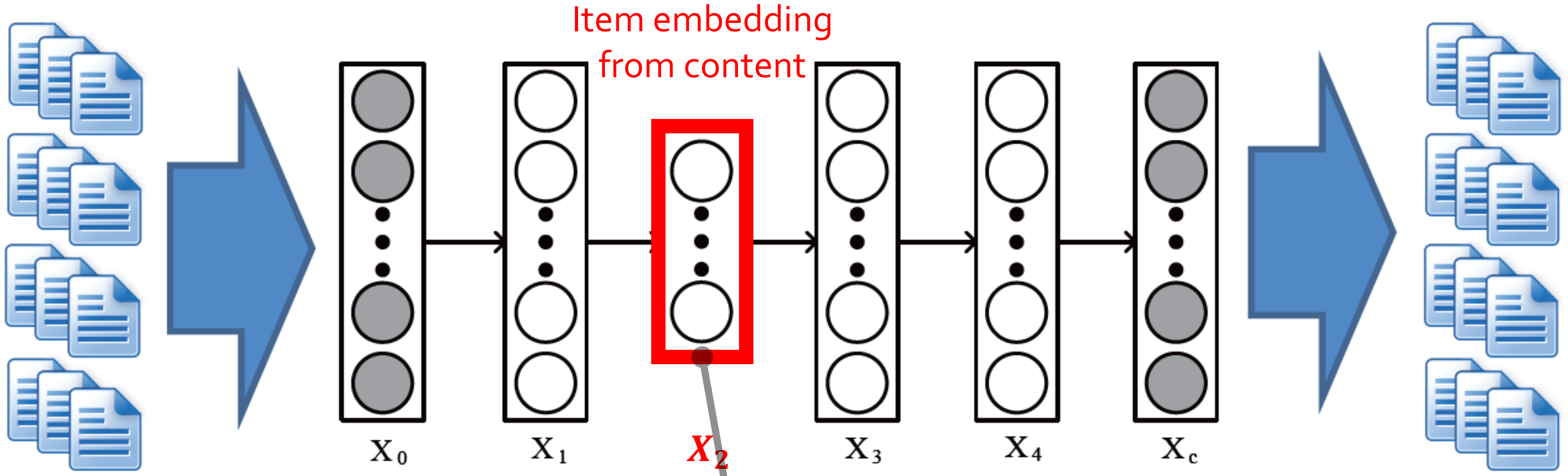
2015
Slides

Collaborative Deep Learning (CDL)



Collaborative Deep Learning (Simplified)

Movie content: Descriptions, videos, directors, etc.



Minimize $||\boldsymbol{x_2} - \boldsymbol{v_j}||_2^2$

Rating matrix $\boldsymbol{R}$ (users × items)

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

≈

User matrix $\boldsymbol{U}$ (users × factors)

.1	-.4	.2
-.5	.6	.5
-.2	.3	.5
1.1	2.1	.3
-.7	2.1	-2
-1	.7	.3

Item matrix $\boldsymbol{V^T}$ (factors × items)

1.1	-.2	.3	.5	-2	-.5	.8	-.4	.3	1.4	2.4	-.9
-.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7	1.2	-.1	1.3
2.1	-.4	.6	1.7	2.4	.9	-.3	.4	.8	.7	-.6	.1

Item embedding from ratings $\boldsymbol{v_j}$

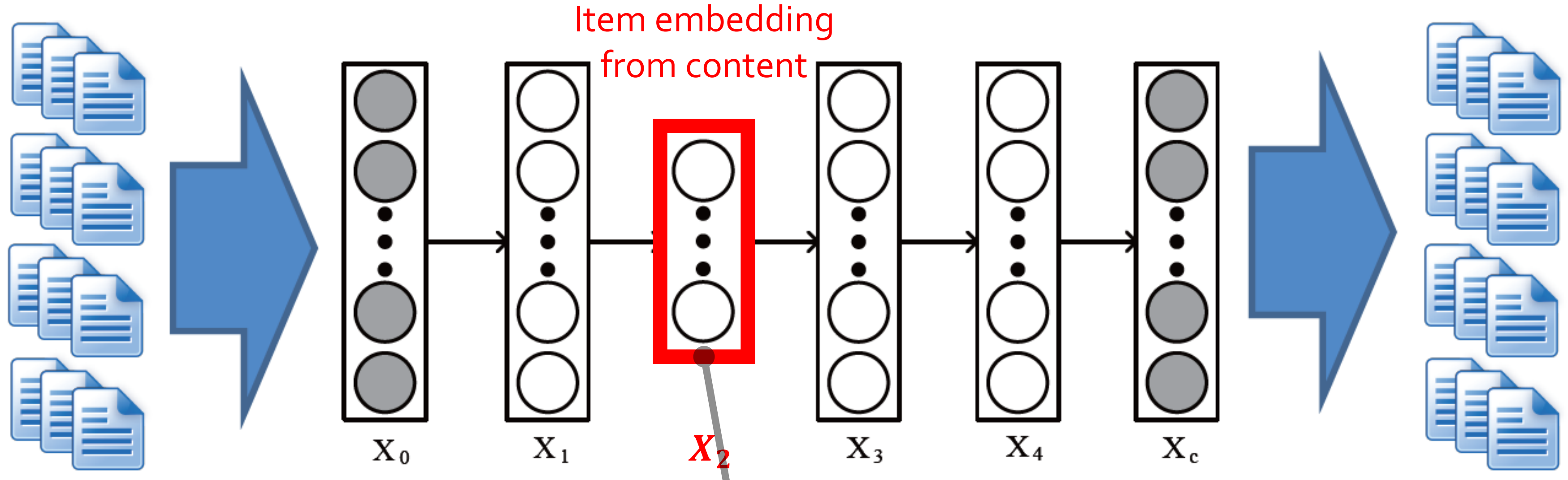
[Rating figure adapted from a slide by Jure]

Collaborative Deep Learning (CDL): Alternate Updates

Deep component (autoencoder)

Alternate between updating

Graphical component (matrix factorization)



Minimize $||\boldsymbol{x_2} - \boldsymbol{v_j}||_2^2$

Rating matrix $\boldsymbol{R}$ (users × items)

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

≈

User matrix $\boldsymbol{U}$ (users × factors)

.1	-.4	.2
-.5	.6	.5
-.2	.3	.5
1.1	2.1	.3
-.7	2.1	-2
-1	.7	.3

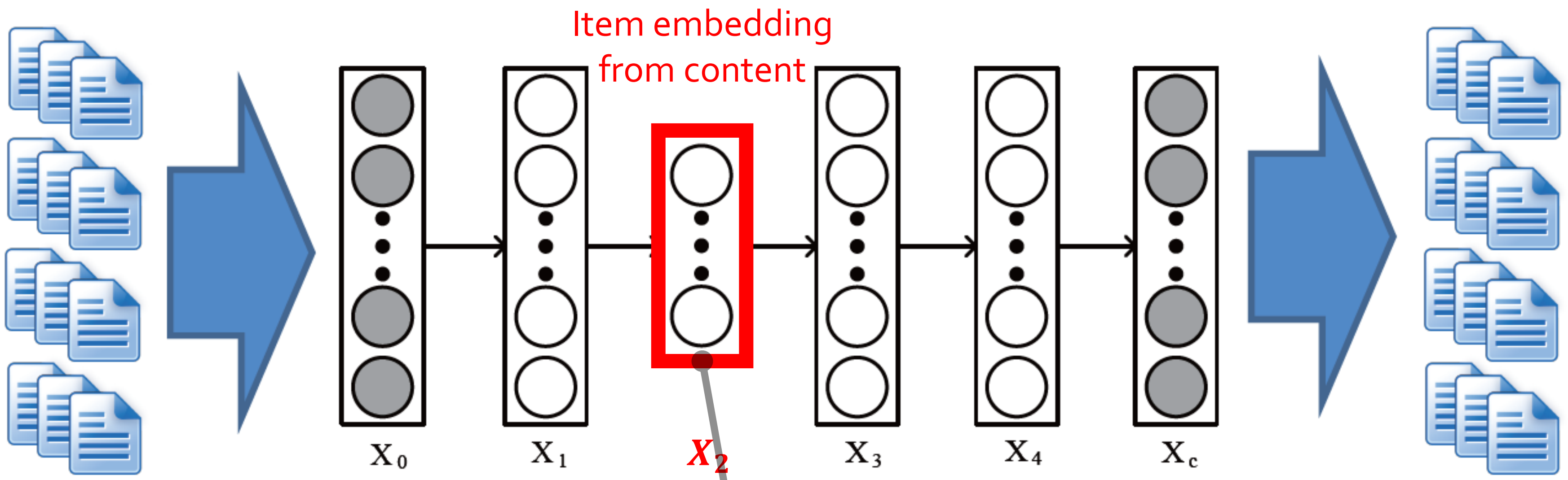
Item matrix $\boldsymbol{V^T}$ (factors × items)

1.1	-.2	.3	.5	-2	-.5	.8	-.4	.3	1.4	2.4	-.9
-.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7	1.2	-.1	1.3
2.1	-.4	.6	1.7	2.4	.9	-.3	.4	.8	.7	-.6	.1

Item embedding from ratings $\boldsymbol{v_j}$

CDL: Updating the Deep Component (Autoencoder)

Movie content: Descriptions, videos, directors, etc.



Learn AE: Minimize Recon. loss $||x_4 - x_c||_2^2$

Minimize regress. (L2) loss $||x_2 - v_j||_2^2$

Objective function to update the deep component (the autoencoder):

$$\min_{\text{network parameters}} ||x_4 - x_c||_2^2 + ||x_2 - v_j||_2^2$$

items

1.1	-.2	.3	.5	-2	-.5	.8	-.4	.3	1.4	2.4	-.9
-.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7	1.2	-.1	1.3
2.1	-.4	.6	1.7	2.4	.9	-.3	.4	.8	.7	-.6	.1

factors

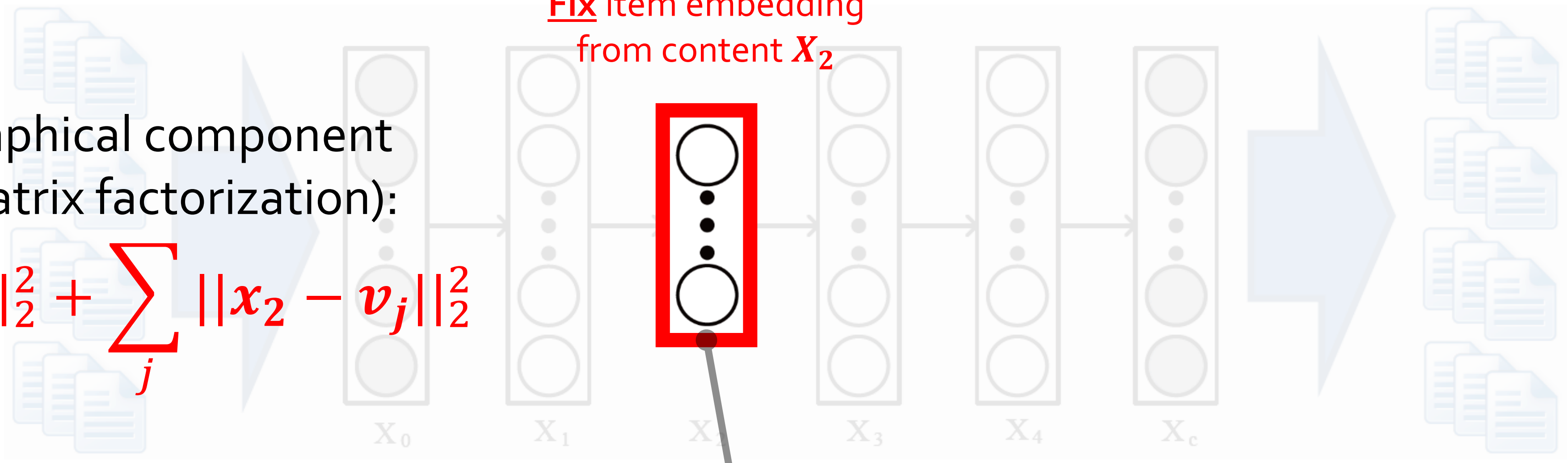
Fix item embedding from ratings v_j

Item matrix V^T

CDL: Updating the Graphical Component (Matrix Factorization)

Update the graphical component (regularized matrix factorization):

$$\min_{U,V} ||R - UV^T||_2^2 + \sum_j ||x_2 - v_j||_2^2$$



$$\min_{U,V} ||R - UV^T||_2^2$$

$$\min_{V} ||x_2 - v_j||_2^2$$

Rating matrix R (users × items):

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

≈

User matrix U (users × factors):

.1	-.4	.2
-.5	.6	.5
-.2	.3	.5
1.1	2.1	.3
-.7	2.1	-2
-1	.7	.3

Item matrix V^T (factors × items):

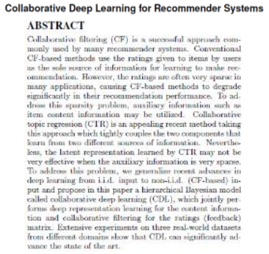
1.1	-.2	.3	.5	-2	-.5	.8	-.4	.3	1.4	2.4	-.9
-.8	.7	.5	1.4	.3	-1	1.4	2.9	-.7	1.2	-.1	1.3
2.1	-.4	.6	1.7	2.4	.9	-.3	.4	.8	.7	-.6	.1

Item embedding from ratings v_j

Experimental Setup

	citeulike-a	citeulike-t	Netflix
#users	5551	7947	407261
#items	16980	25975	9228
#ratings	204987	134860	15348808

Content information

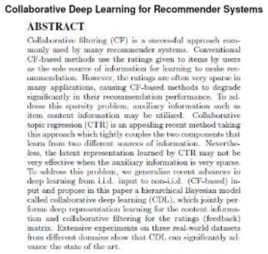


Collaborative Deep Learning for Recommender Systems

ABSTRACT

Collaborative filtering (CF) is a successful approach commonly used by many recommender systems. Conventional CF-based methods use the ratings given to items by users as the sole source of information for learning to make recommendation. However, the ratings are often very sparse in many applications, causing CF-based methods to degrade significantly in their recommendation performance. To address this sparsity problem, auxiliary information such as item content information may be utilized. Collaborative topic regression (CTR) is an appealing recent method taking this approach which tightly couples the two components that learn from two different sources of information. Nevertheless, the latent representation learned by CTR may not be very effective when the auxiliary information is very sparse. To address this problem, we generalize recent advances in deep learning from i.i.d. input to non-i.i.d. (CF-based) input and propose in this paper a hierarchical Bayesian model called collaborative deep learning (CDL), which jointly performs deep representation learning for the content information and collaborative filtering for the ratings (feedback) matrix. Extensive experiments on three real-world datasets from different domains show that CDL can significantly advance the state of the art.

Titles and abstracts

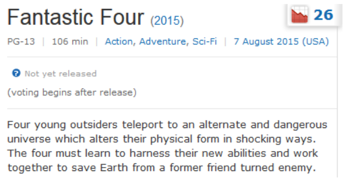


Collaborative Deep Learning for Recommender Systems

ABSTRACT

Collaborative filtering (CF) is a successful approach commonly used by many recommender systems. Conventional CF-based methods use the ratings given to items by users as the sole source of information for learning to make recommendation. However, the ratings are often very sparse in many applications, causing CF-based methods to degrade significantly in their recommendation performance. To address this sparsity problem, auxiliary information such as item content information may be utilized. Collaborative topic regression (CTR) is an appealing recent method taking this approach which tightly couples the two components that learn from two different sources of information. Nevertheless, the latent representation learned by CTR may not be very effective when the auxiliary information is very sparse. To address this problem, we generalize recent advances in deep learning from i.i.d. input to non-i.i.d. (CF-based) input and propose in this paper a hierarchical Bayesian model called collaborative deep learning (CDL), which jointly performs deep representation learning for the content information and collaborative filtering for the ratings (feedback) matrix. Extensive experiments on three real-world datasets from different domains show that CDL can significantly advance the state of the art.

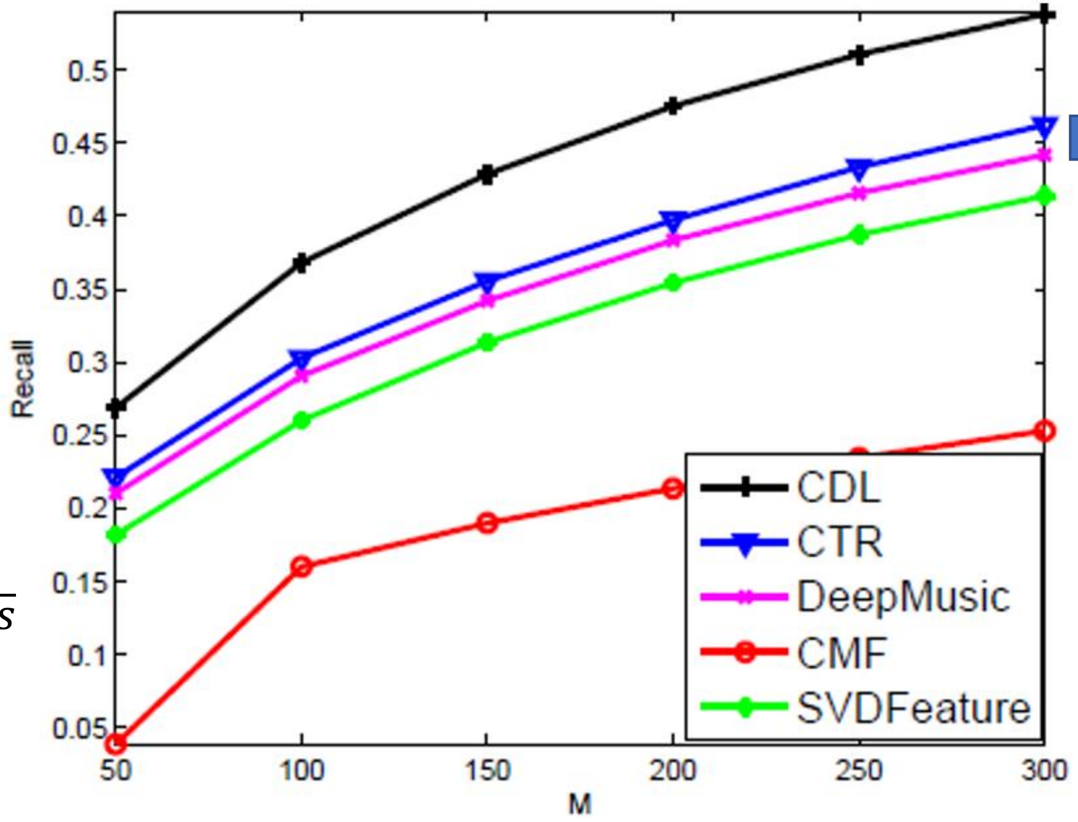
Titles and abstracts



Movie descriptions

[Collaborative topic modeling for recommending scientific articles. WB. *KDD* 2011]
[Collaborative topic regression with social regularization for tag recommendation. WCL. *IJCAI* 2013]

Empirical Results: Recall@M in *citeulike-t*



CTR & DeepMusic best baselines

y-axis: recall@M

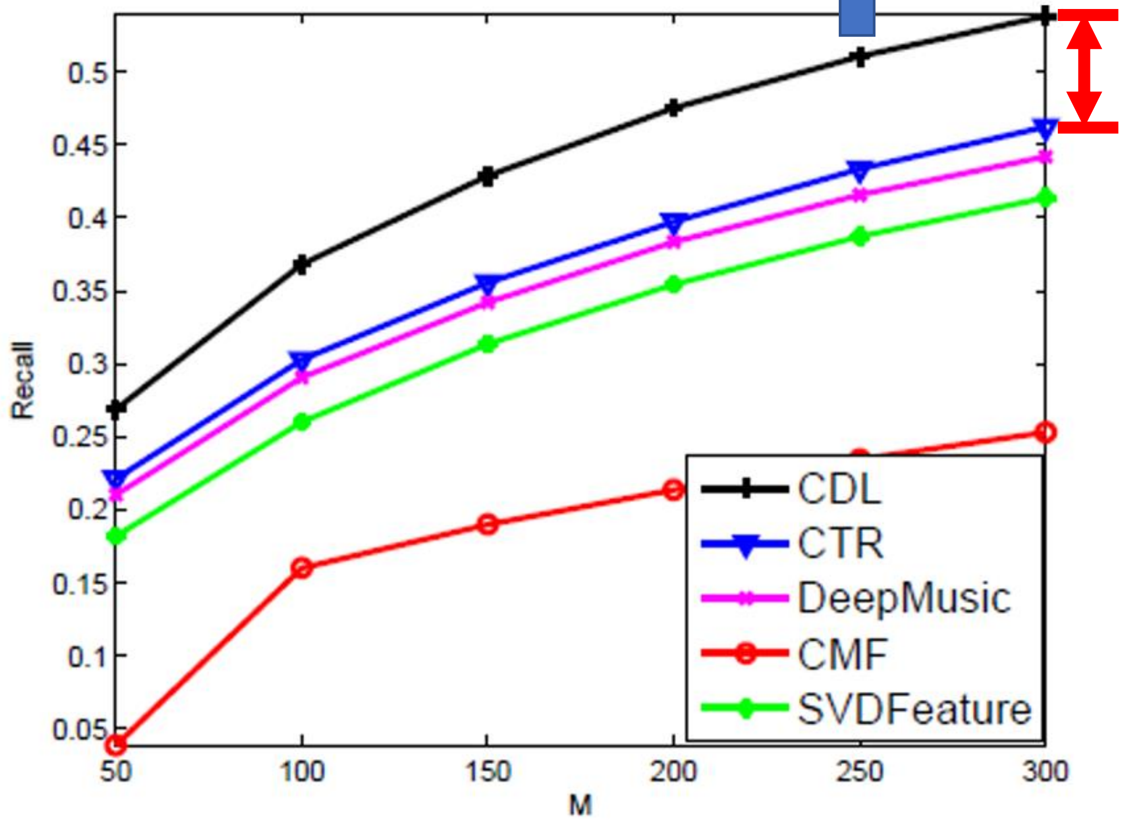
$$\frac{number\ of\ hits}{total\ correct\ items}$$

x-axis: number of recommended items M

Empirical Results: Recall@M in *citeulike-t*

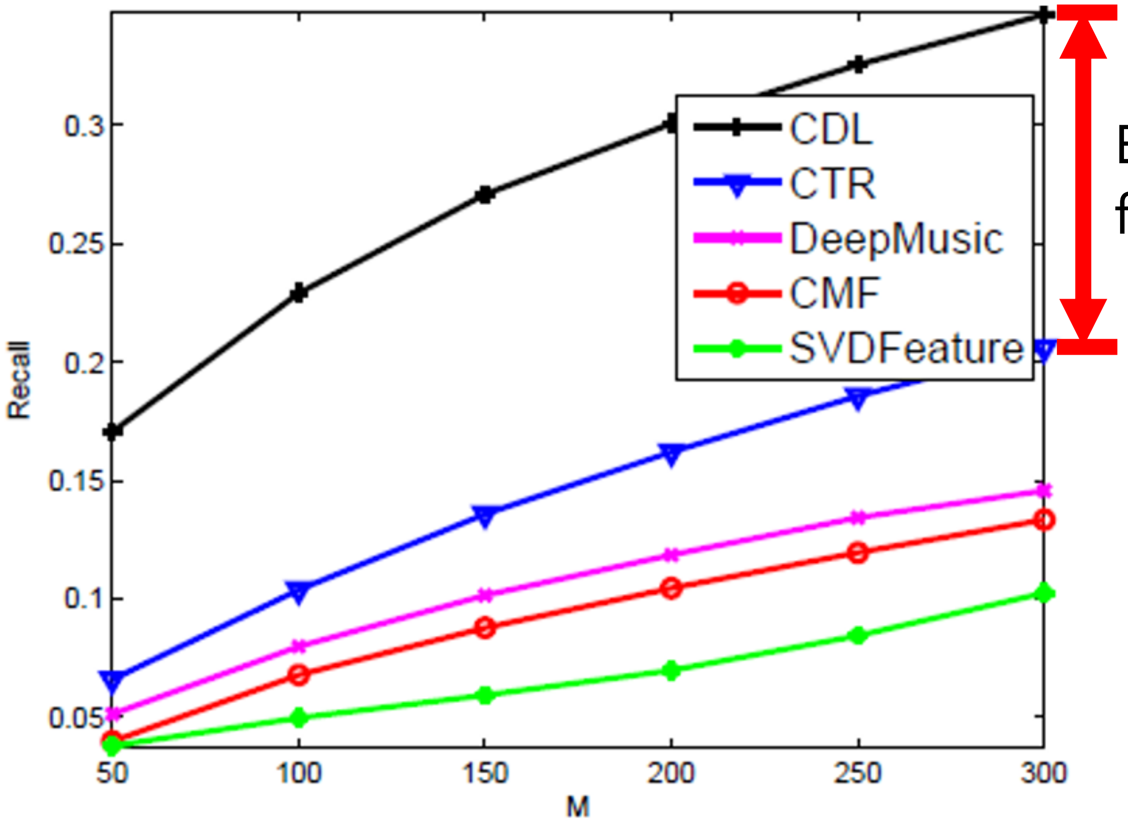
Our method: **CDL**

Boost recall from **46%** to **54%**



8% absolute improvement

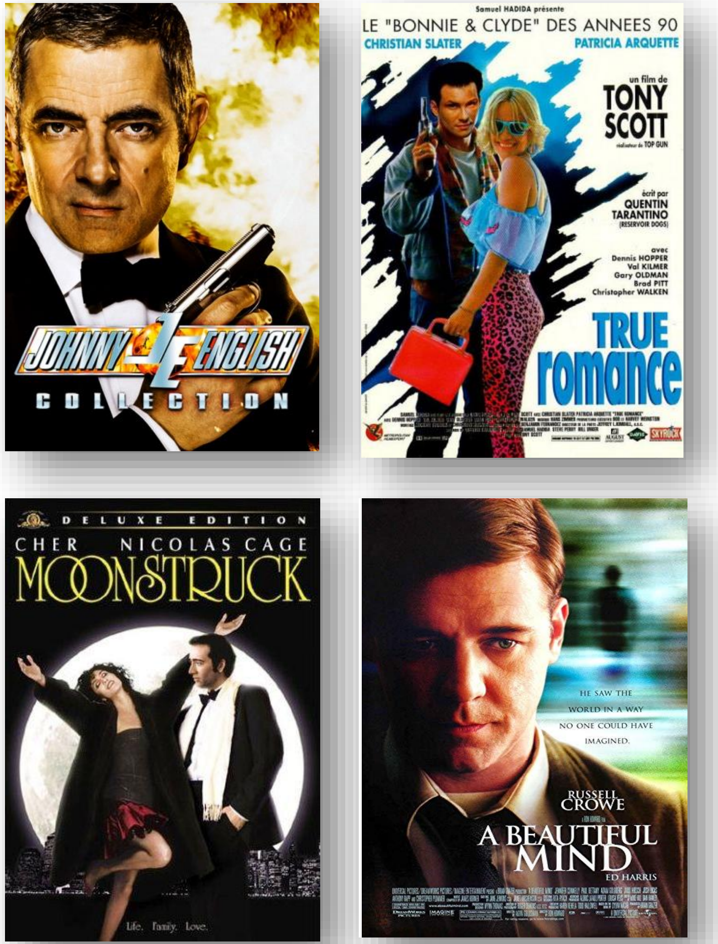
Empirical Results: Recall@M in *citeulike-t* (Sparse Ratings)



Boost recall from **21%** to **35%**

14% absolute improvement

CDL for Sparse Ratings



Content information:
Plots, directors, actors, etc.

movie \ user	1	2	3	4	5
1	✓	?	?	?	?
2	✓	?	?	✓	?
3	?	?	✓	?	?
4	?	✓	?	?	✓
5	✓	?	?	?	?

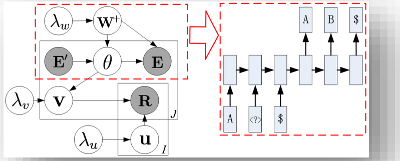
Sparse rating matrix

Follow Up: NeurIPS 2016 Paper

Collaborative Recurrent Autoencoder: Recommend while Learning to Fill in the Blanks

Hao Wang, Xingjian Shi, Dit-Yan Yeung
Hong Kong University of Science and Technology
{hwangaz,xshiab,dyyeung}@cse.ust.hk

- **Fully generative model**: Jointly performs *recommendation* and masked autoregressive text *generation*
- Replace fully connected layers with recurrent neural nets
- **Attention mechanism**: Aggregate *variable*-length text into *fixed*-length embeddings



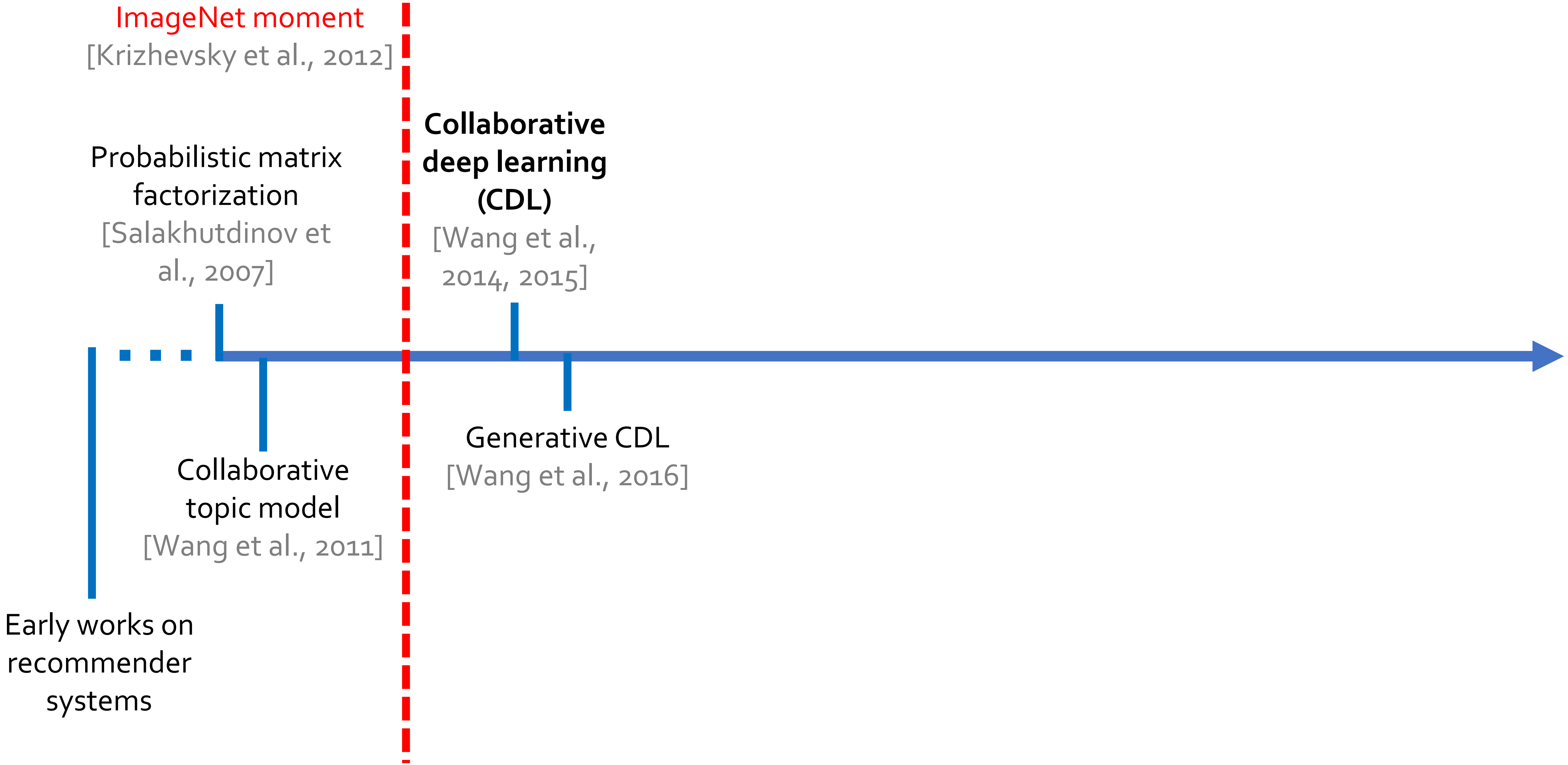
[Wang et al. *NeurIPS* 2016]

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova
Google AI Language
{jacobdevlin,mingweichang,kentonl,kristout}@google.com

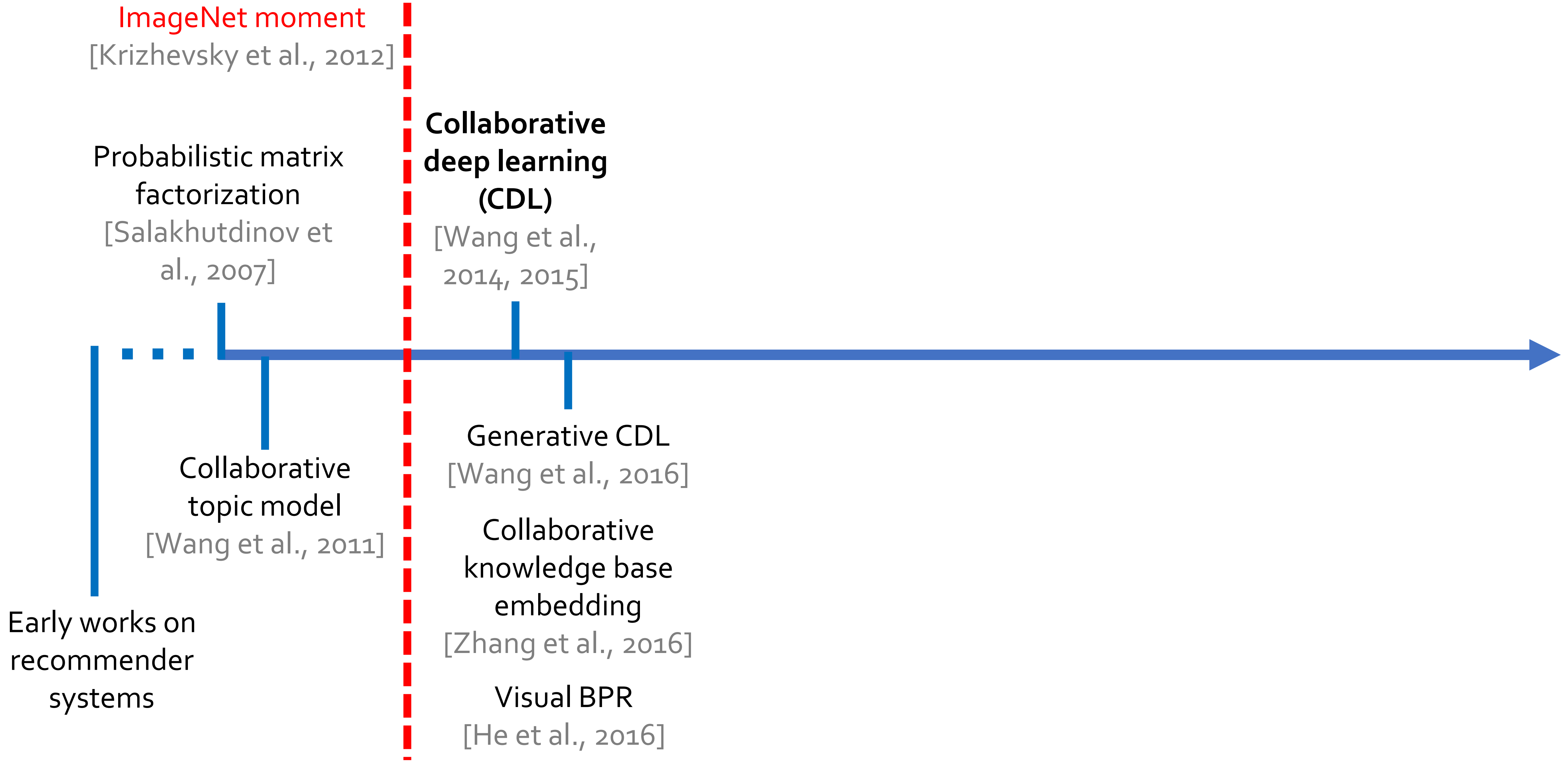
BERT [Devlin et al. *ACL* 2019]

Timeline: (Deep) Recommender Systems

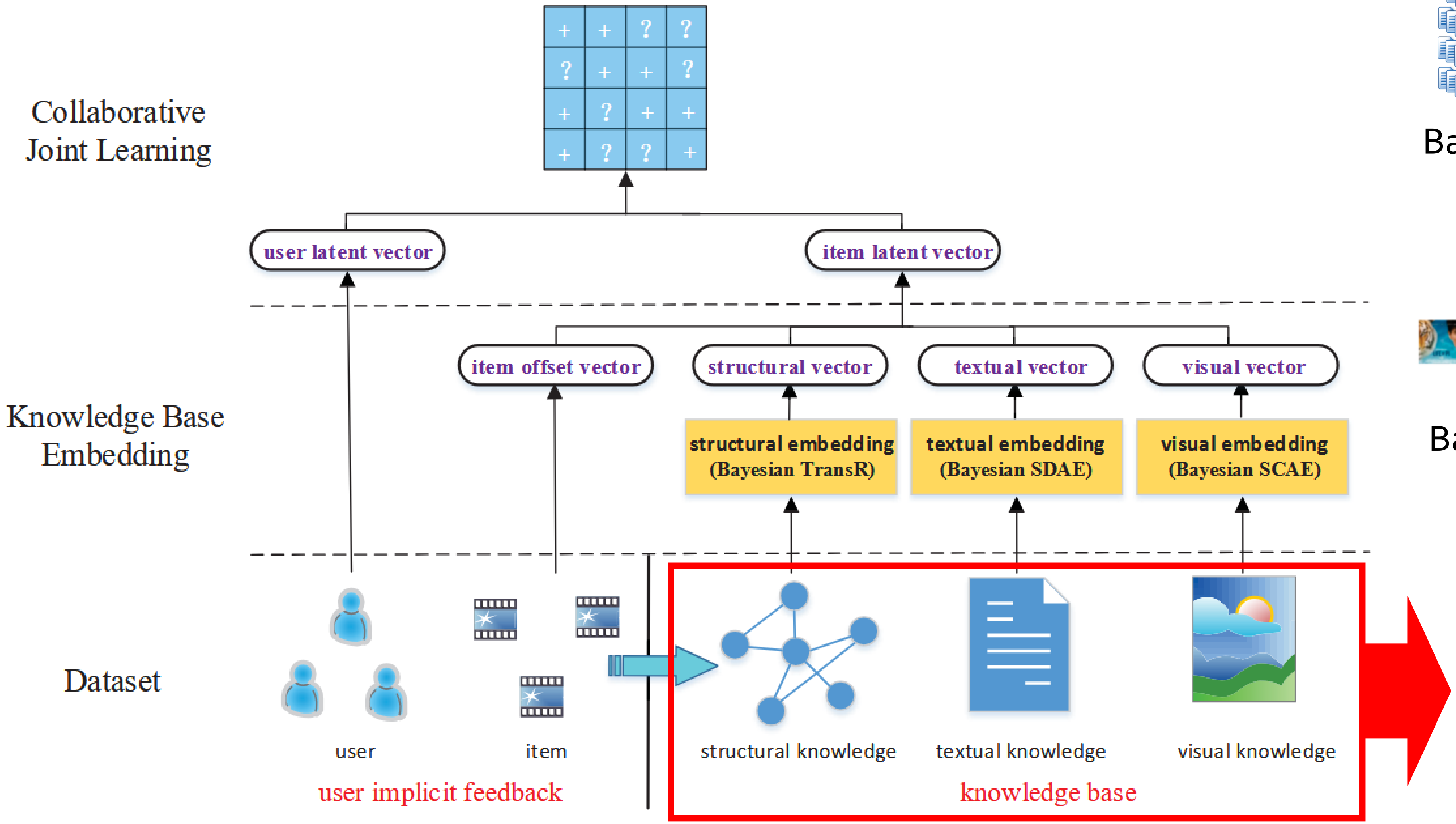


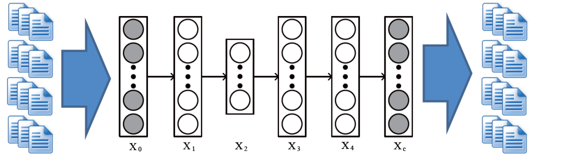
What Happened Since?

Timeline after CDL: Deep Recommender Systems

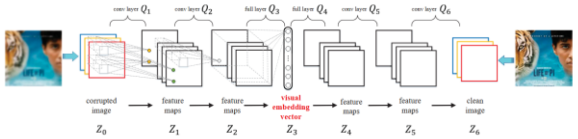


Collaborative Knowledge Base Embedding (KDD 2016)





Bayesian autoencoder in CDL to process **text**



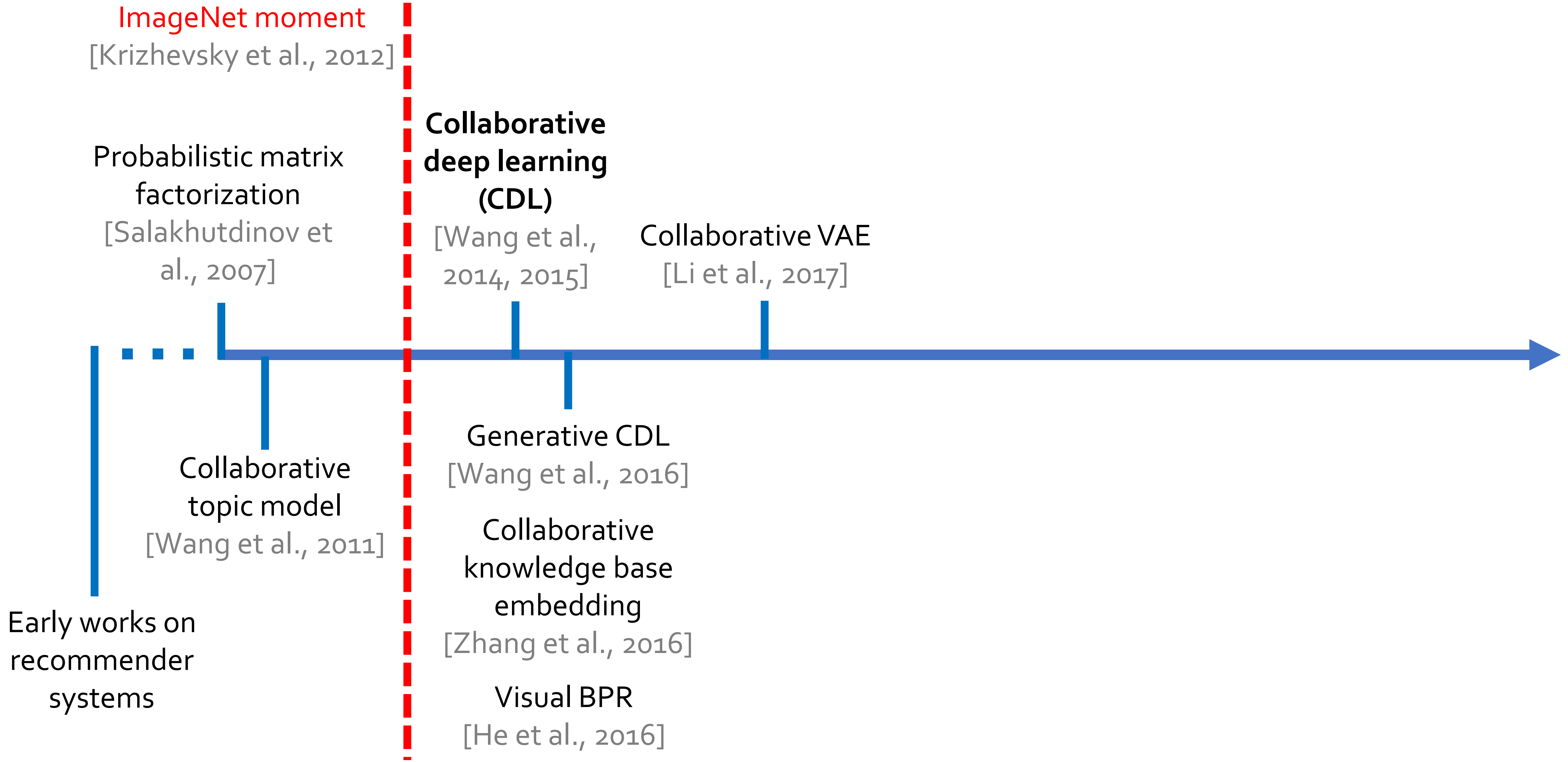
Bayesian *conv.* autoencoder to process **images**

Different modalities in knowledge bases:

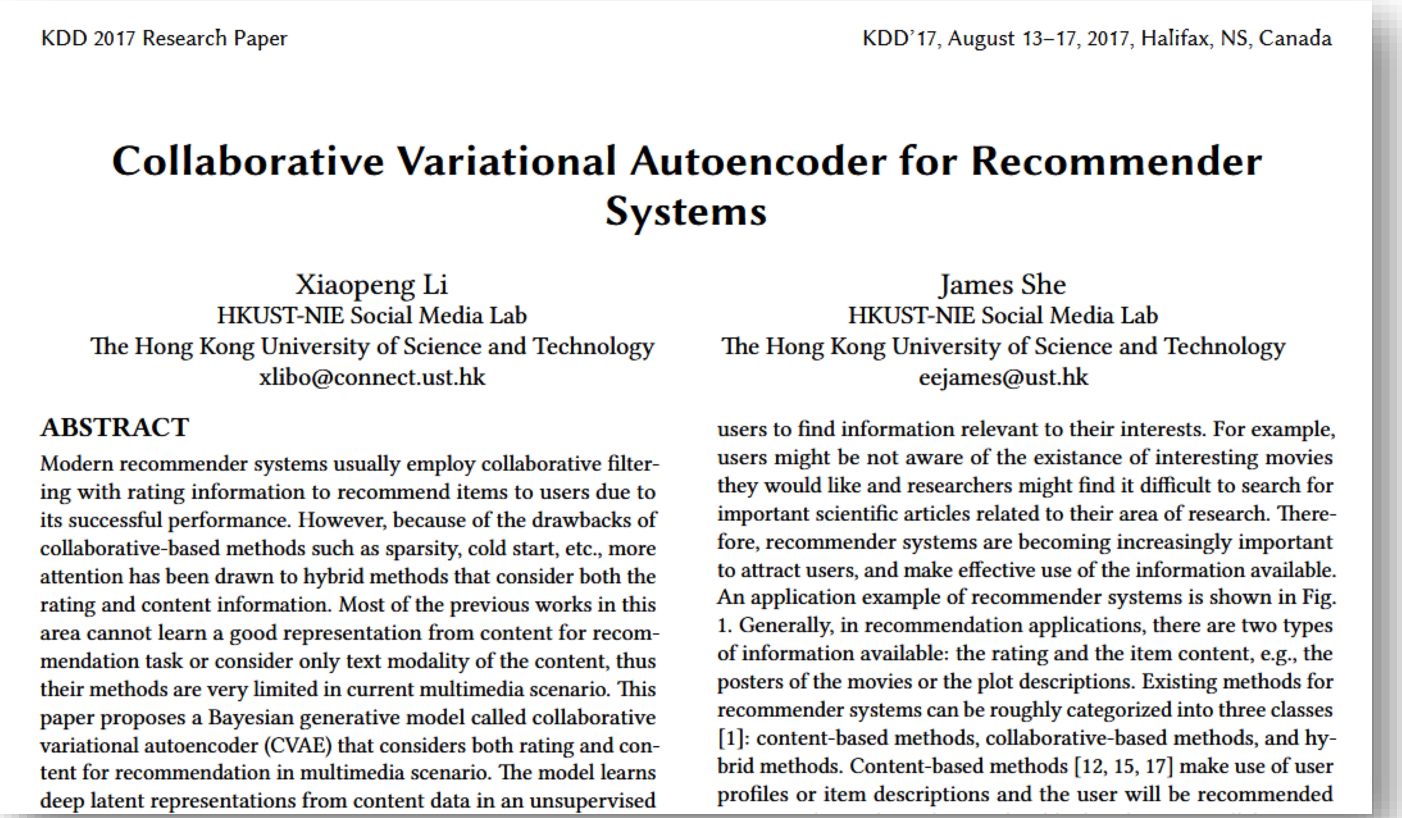
- Graphs
- Text
- Images

[Collaborative knowledge base embedding for recommender systems. ZYLXM. *KDD* 2016]

Timeline: Deep Recommender Systems



Collaborative Variational Autoencoder (KDD 2017)



KDD 2017 Research Paper — KDD'17, August 13–17, 2017, Halifax, NS, Canada

Collaborative Variational Autoencoder for Recommender Systems

Xiaopeng Li
HKUST-NIE Social Media Lab
The Hong Kong University of Science and Technology
xlibo@connect.ust.hk

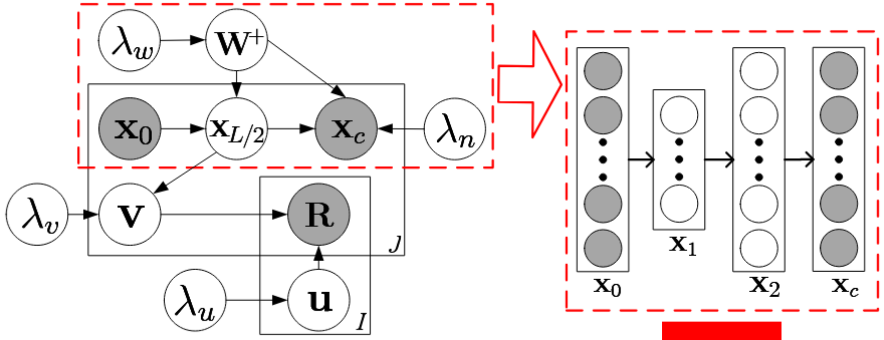
James She
HKUST-NIE Social Media Lab
The Hong Kong University of Science and Technology
eejames@ust.hk

ABSTRACT

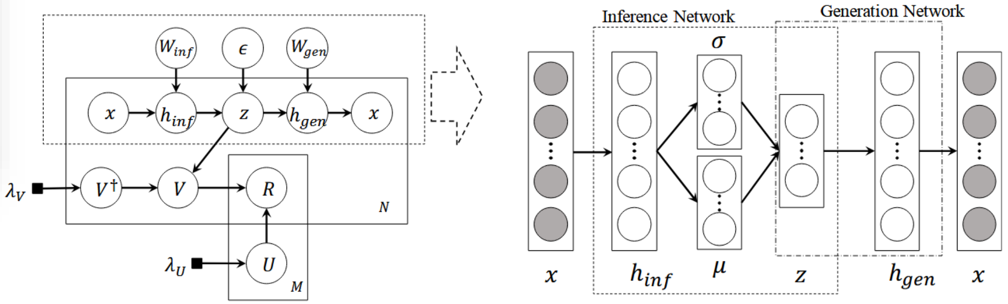
Modern recommender systems usually employ collaborative filtering with rating information to recommend items to users due to its successful performance. However, because of the drawbacks of collaborative-based methods such as sparsity, cold start, etc., more attention has been drawn to hybrid methods that consider both the rating and content information. Most of the previous works in this area cannot learn a good representation from content for recommendation task or consider only text modality of the content, thus their methods are very limited in current multimedia scenario. This paper proposes a Bayesian generative model called collaborative variational autoencoder (CVAE) that considers both rating and content for recommendation in multimedia scenario. The model learns deep latent representations from content data in an unsupervised

users to find information relevant to their interests. For example, users might be not aware of the existance of interesting movies they would like and researchers might find it difficult to search for important scientific articles related to their area of research. Therefore, recommender systems are becoming increasingly important to attract users, and make effective use of the information available. An application example of recommender systems is shown in Fig. 1. Generally, in recommendation applications, there are two types of information available: the rating and the item content, e.g., the posters of the movies or the plot descriptions. Existing methods for recommender systems can be roughly categorized into three classes [1]: content-based methods, collaborative-based methods, and hybrid methods. Content-based methods [12, 15, 17] make use of user profiles or item descriptions and the user will be recommended

Probabilistic autoencoder



Replace the **probabilistic autoencoder** in CDL with a **variational autoencoder** (VAE)

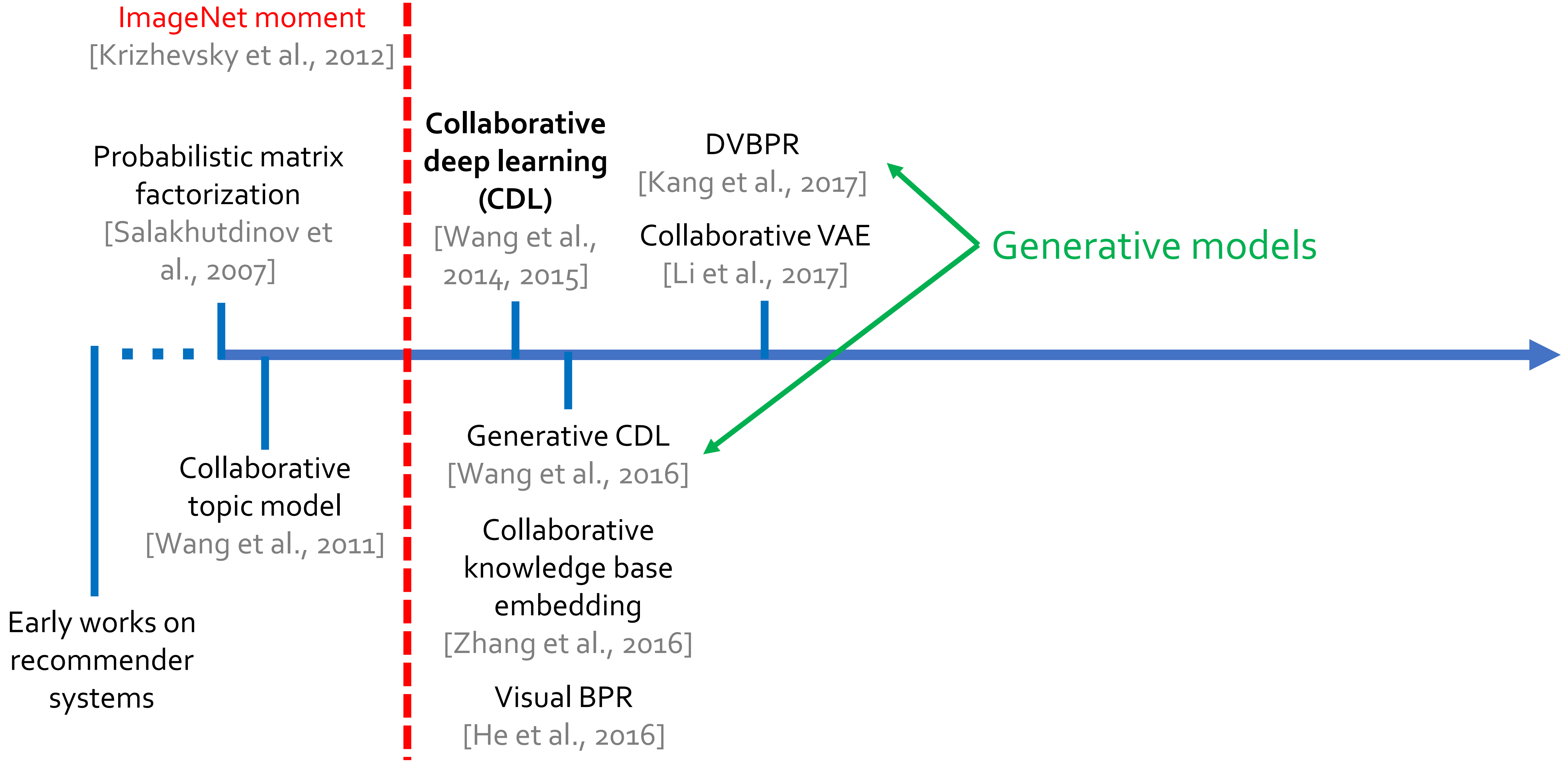


Variational autoencoder (VAE)

- More robust to noise
- Better recommendation performance

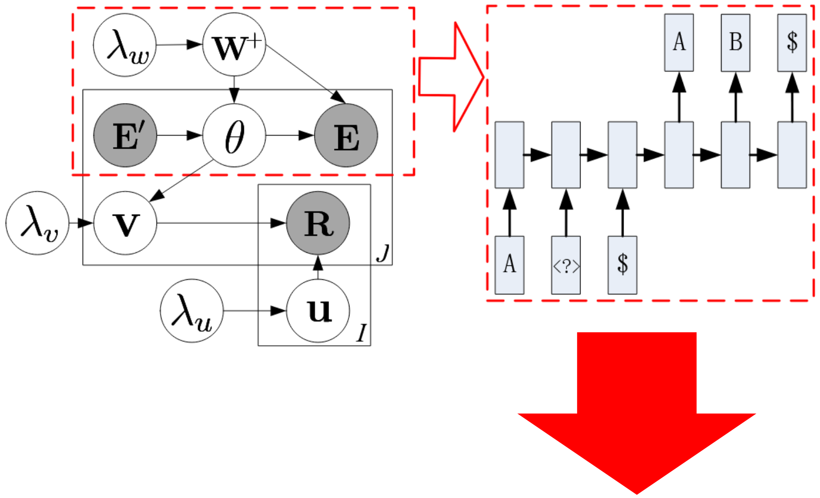
[Collaborative variational autoencoder for recommender systems. LS. *KDD* 2017]

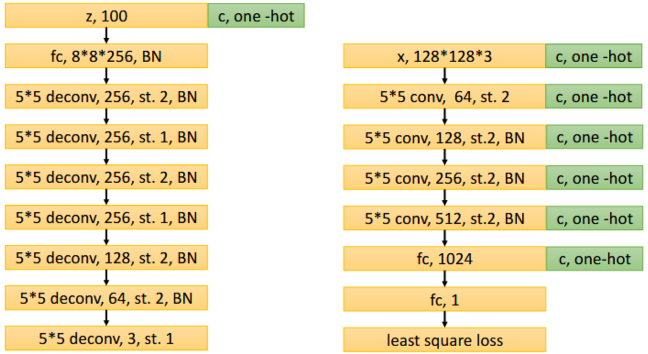
Timeline: Deep Recommender Systems



Recommendation and Design with Generative Image Models (ICDM 2017)

[Wang et al. *NeurIPS* 2016]



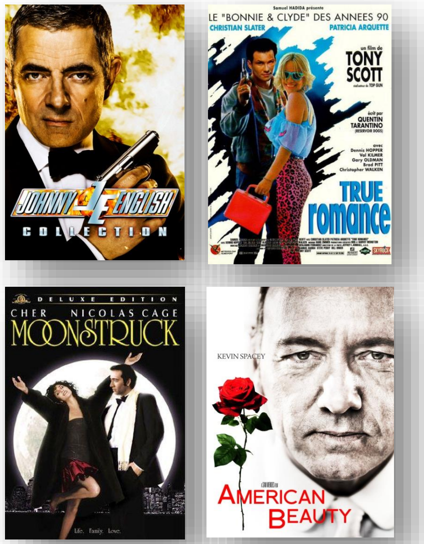


GAN generator GAN discriminator

The hope: Generated designs better match the user's preference

[Collaborative recurrent autoencoder: Recommend while learning to fill in the blanks. WSY. *NeurIPS* 2016]
[Visually-aware fashion recommendation and design with generative image models. KFWM. *ICDM* 2017]

The CDL Hypothesis in 2015



Content information:
Movie descriptions, etc.

movie \ user	1	2	3	4	5
1	✓	?	?	?	?
2	✓	?	?	✓	?
3	?	?	✓	?	?
4	?	✓	?	?	✓
5	✓	?	?	?	?

Rating matrix

Content information i trinsically
hierarchical an w level

Rely on m ti-l er **neural nets**
to extrac -level features

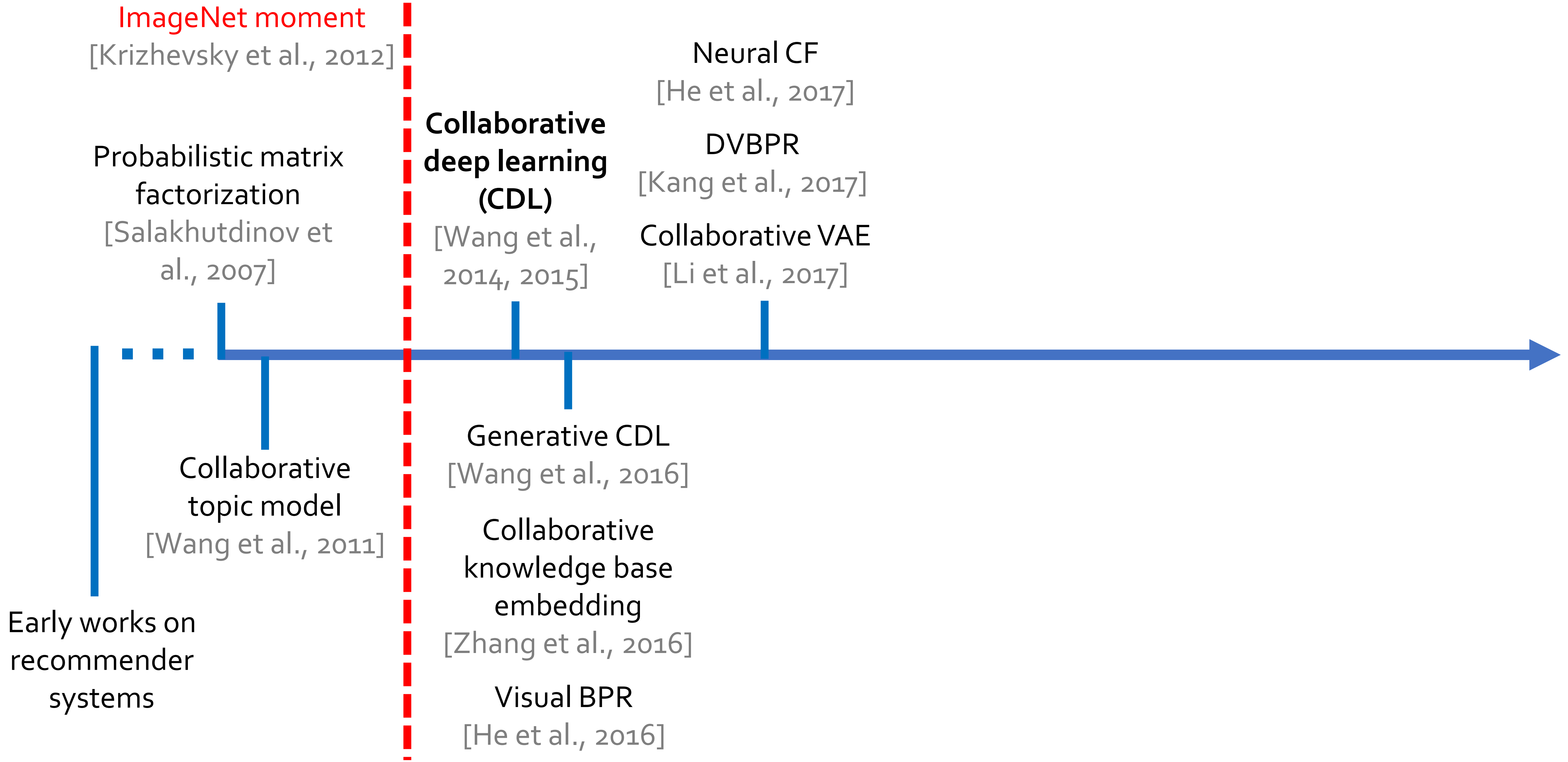
Rating already
captures **high-l** nantics

Linear mo like matrix
factorization mostly suffice

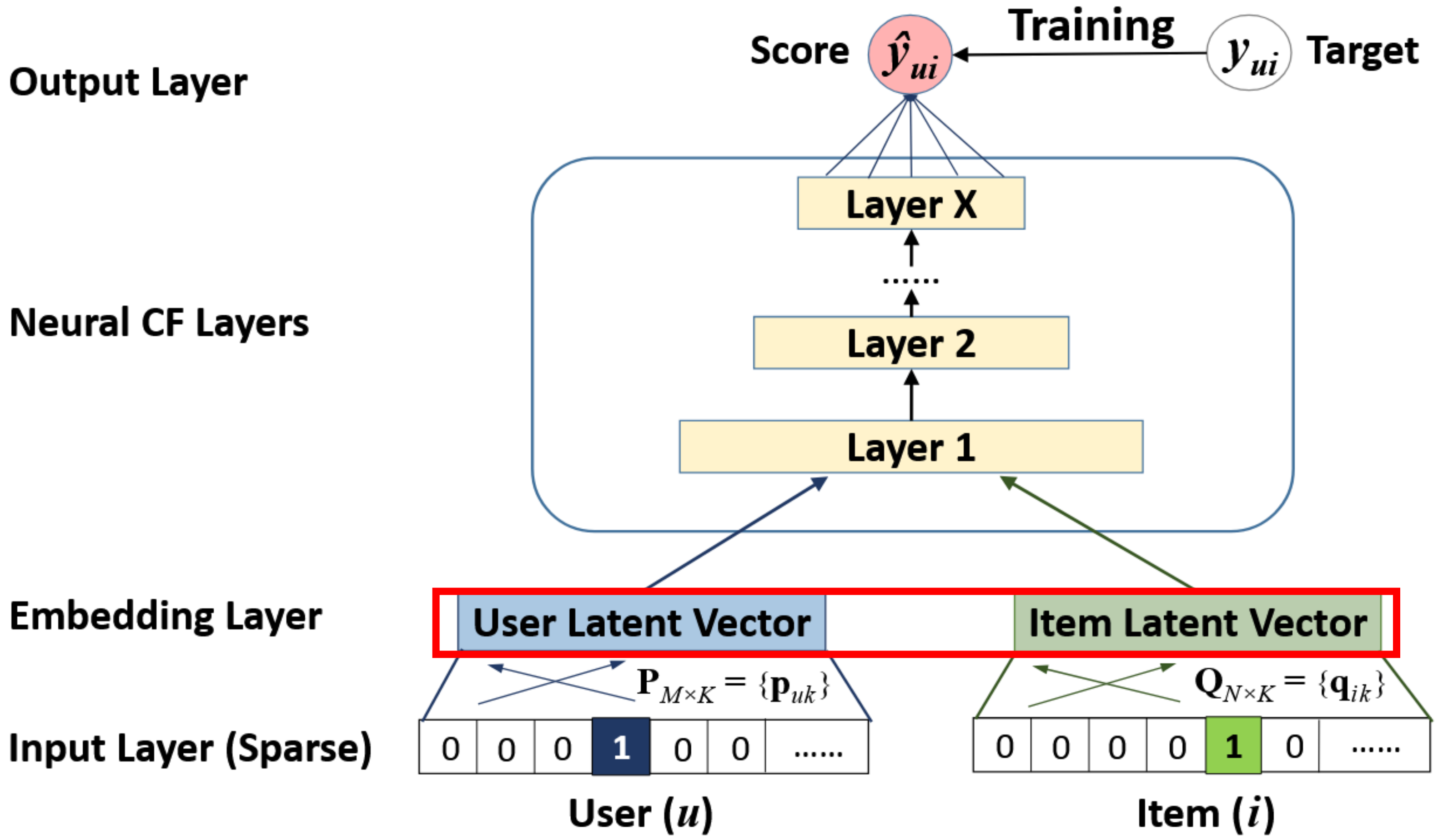
[Collaborative deep learning for recommender systems. WWY. *KDD* 2015]
[Neural collaborative filtering vs. matrix factorization revisited. RKZA. RecSys 2020]

Deep Recommender Systems *without* Content Information

Timeline: Deep Recommender Systems



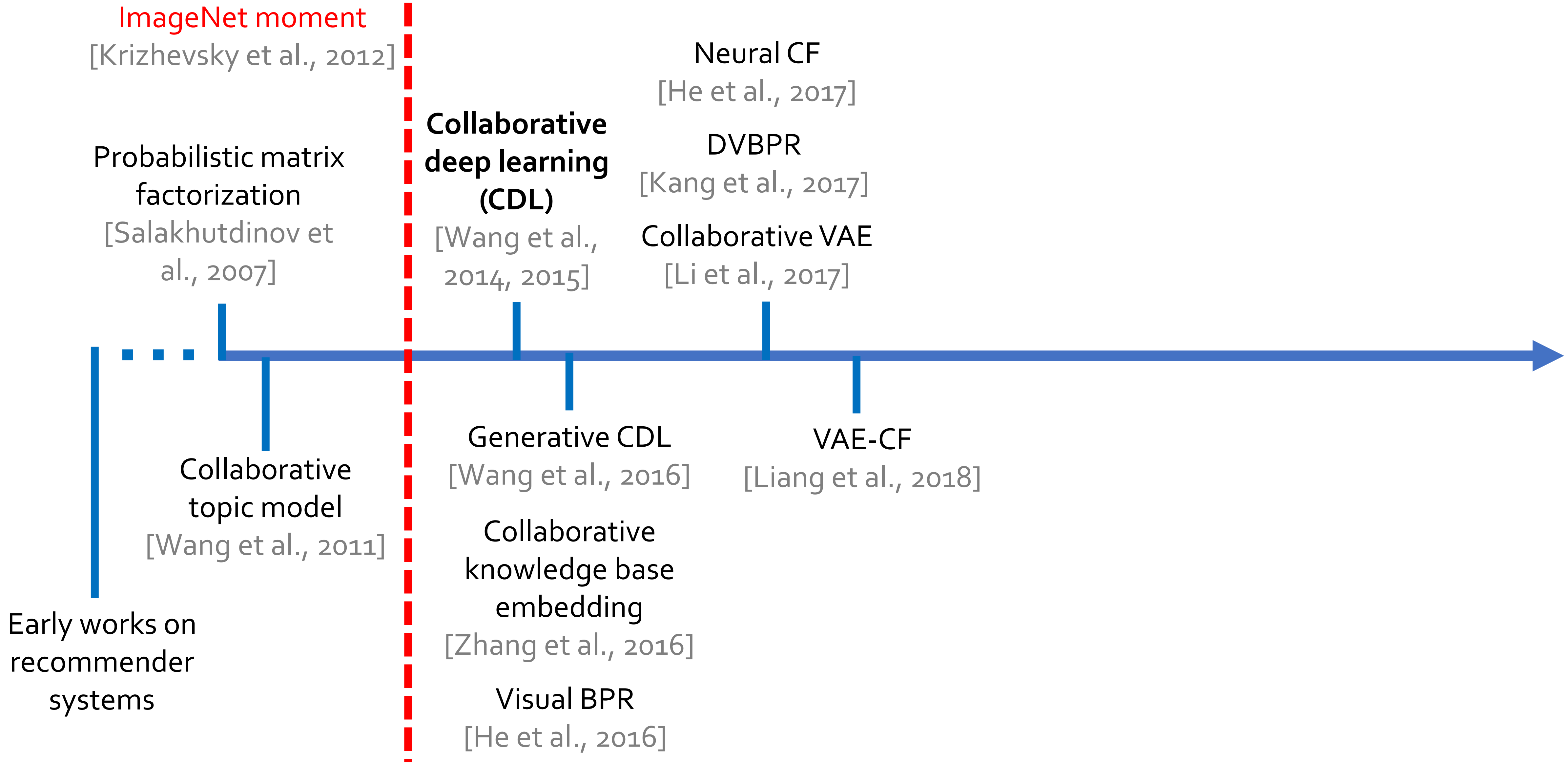
Neural Collaborative Filtering (WWW 2017)



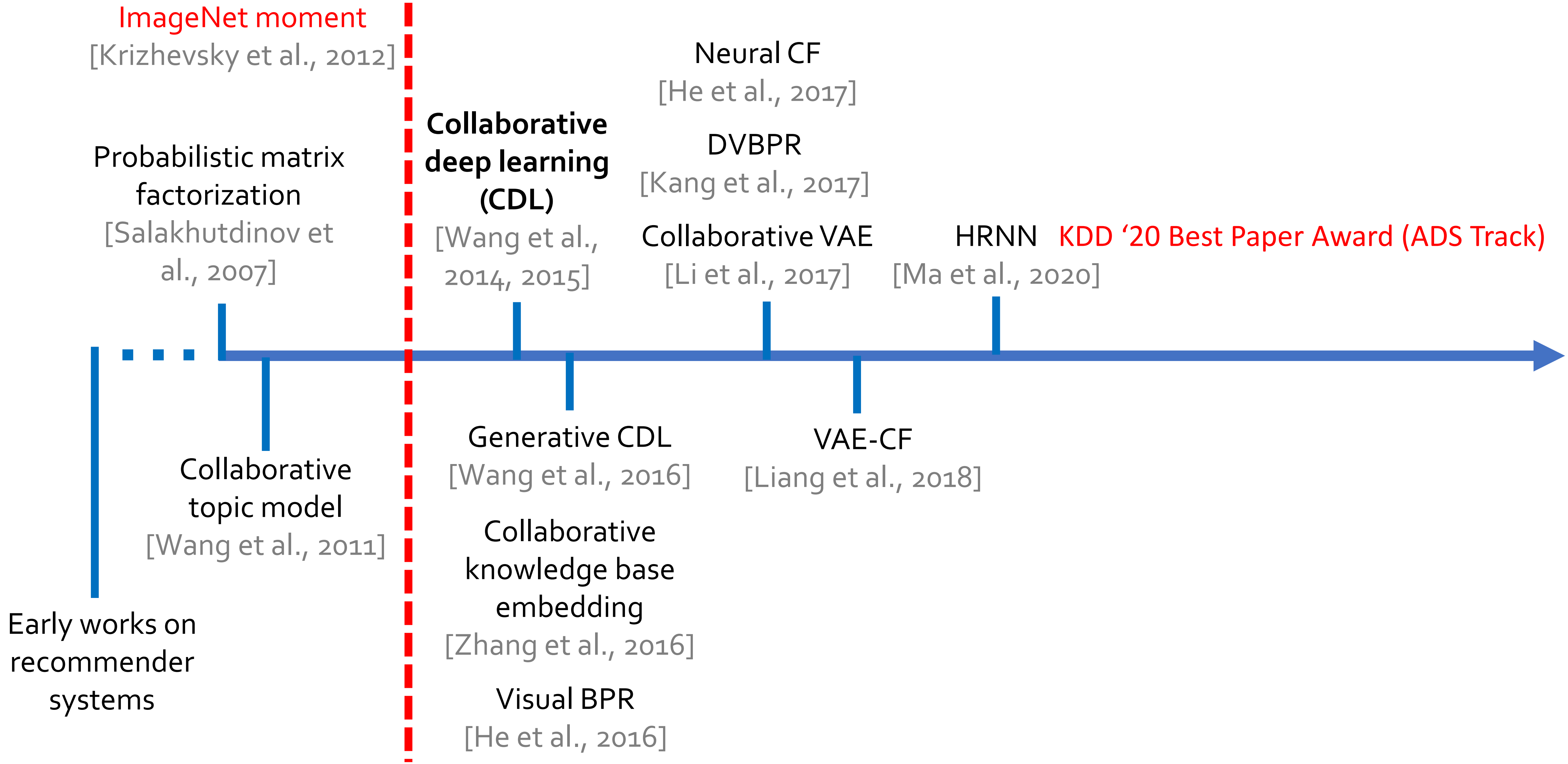
CDL: **Nonlinear** transformation of item **content**, **linear interaction** between users and items
NCF: **No support for** item **content**, **nonlinear interaction** between users and items

[Neural collaborative filtering. HLZNHC. *WWW* 2017]

Timeline: Deep Recommender Systems

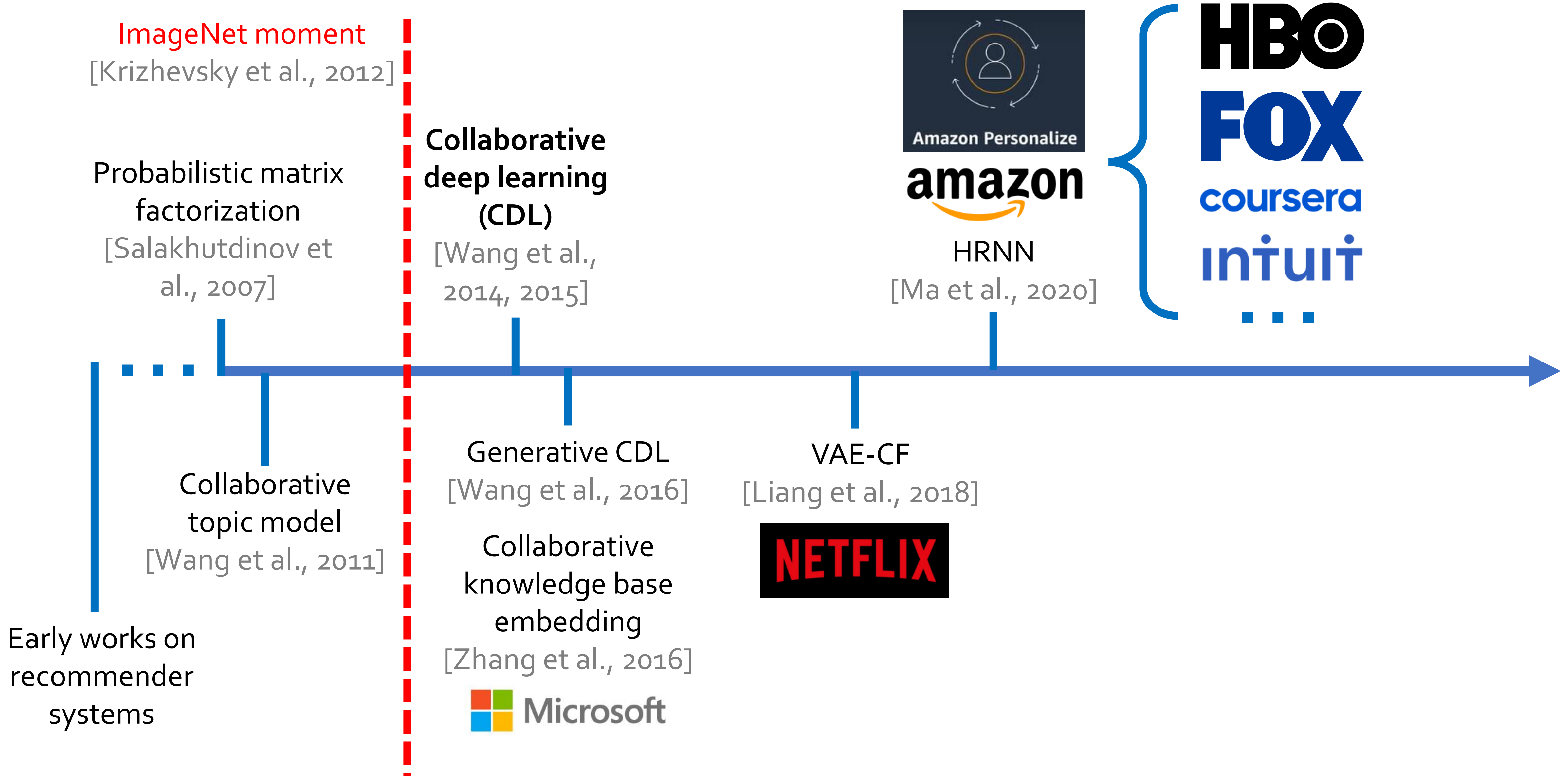


Timeline: Deep Recommender Systems



Deep Recommender Systems in the Industry

Deep Recommender Systems in the Industry



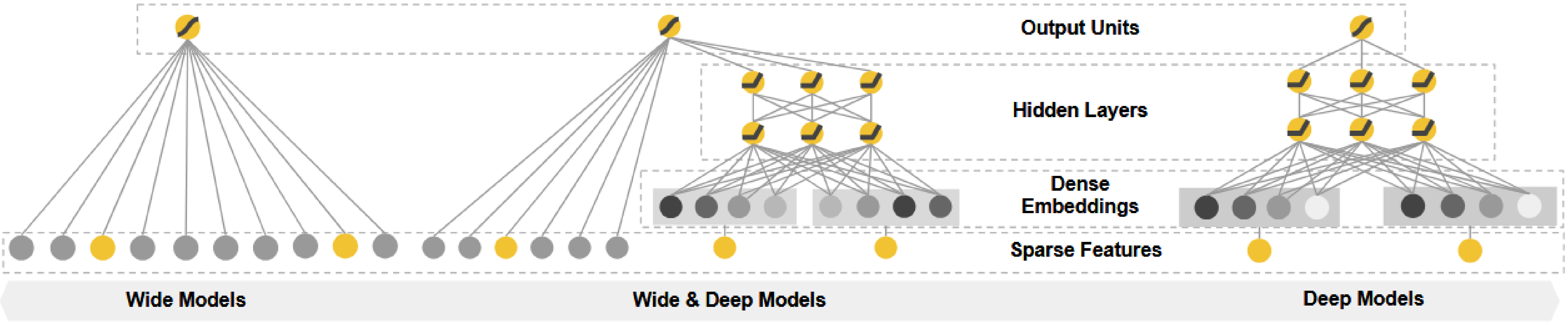
Deep Recommender Systems in the Industry

Google: Wide and deep learning [Cheng et al., DLRS 2016]

Wide & Deep Learning for Recommender Systems

Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, Hemal Shah
Google Inc.*

Deployed at Google Play, Google's App Store

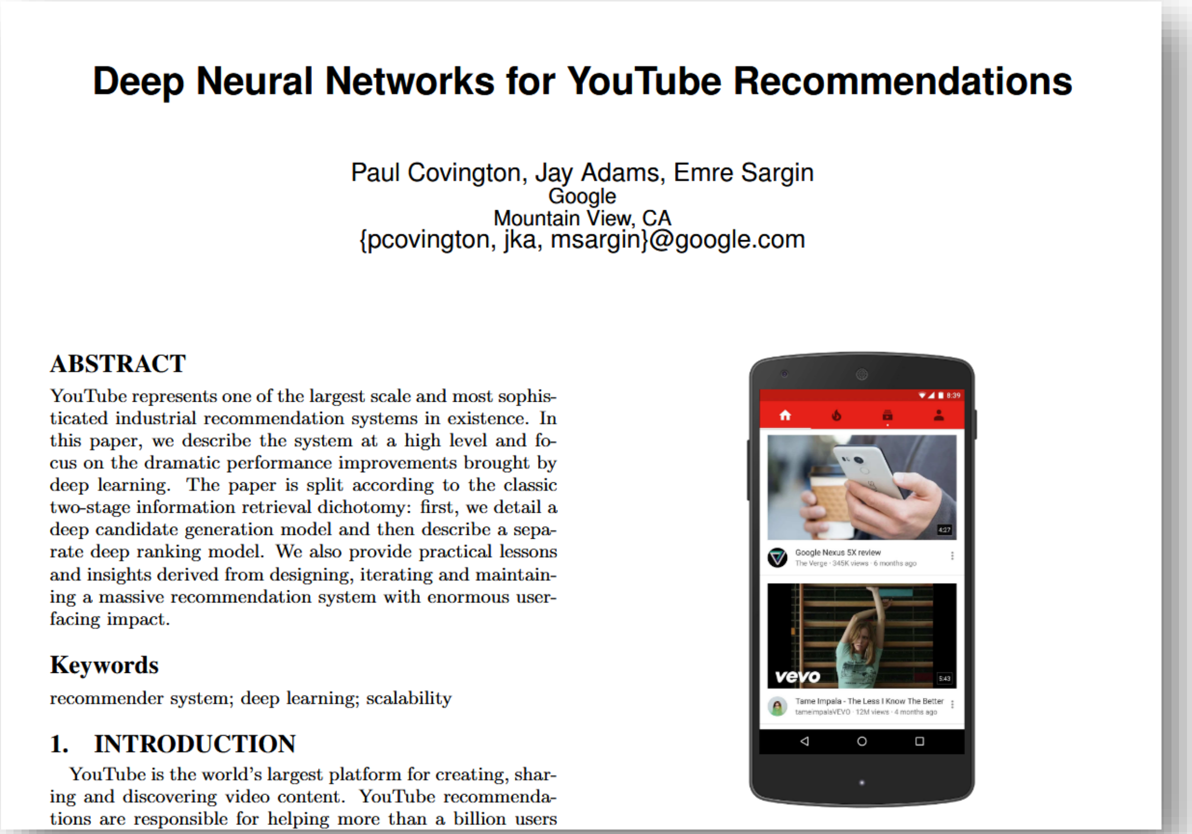


Linear, wide model: Binary features, feature interaction (e.g., AND(gender=female, language=en))

Nonlinear, deep model: Sparse features (e.g., user's installed apps)

Deep Recommender Systems in the Industry

YouTube (Google): Deep Neural Nets for YouTube Recommendations [Covington et al., RecSys 2016]



Deep Neural Networks for YouTube Recommendations

Paul Covington, Jay Adams, Emre Sargin
Google
Mountain View, CA
{pcovington, jka, msargin}@google.com

ABSTRACT

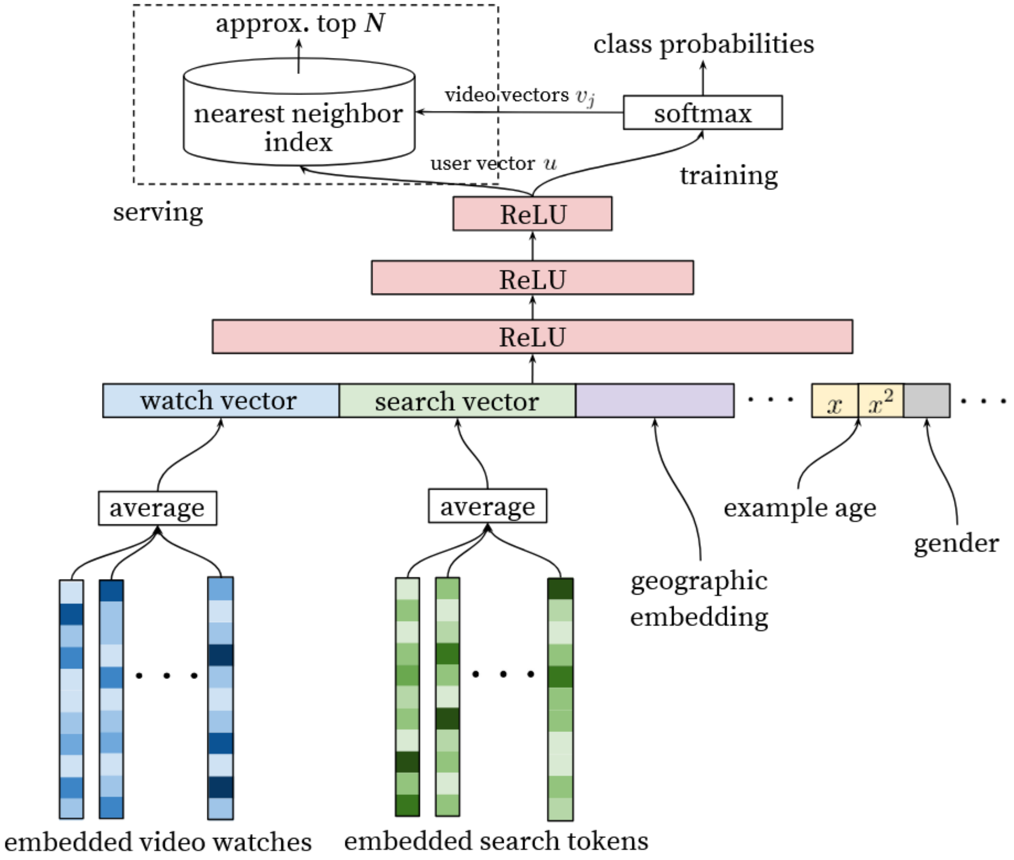
YouTube represents one of the largest scale and most sophisticated industrial recommendation systems in existence. In this paper, we describe the system at a high level and focus on the dramatic performance improvements brought by deep learning. The paper is split according to the classic two-stage information retrieval dichotomy: first, we detail a deep candidate generation model and then describe a separate deep ranking model. We also provide practical lessons and insights derived from designing, iterating and maintaining a massive recommendation system with enormous user-facing impact.

Keywords

recommender system; deep learning; scalability

1. INTRODUCTION

YouTube is the world's largest platform for creating, sharing and discovering video content. YouTube recommendations are responsible for helping more than a billion users



Deployed and evaluated at YouTube

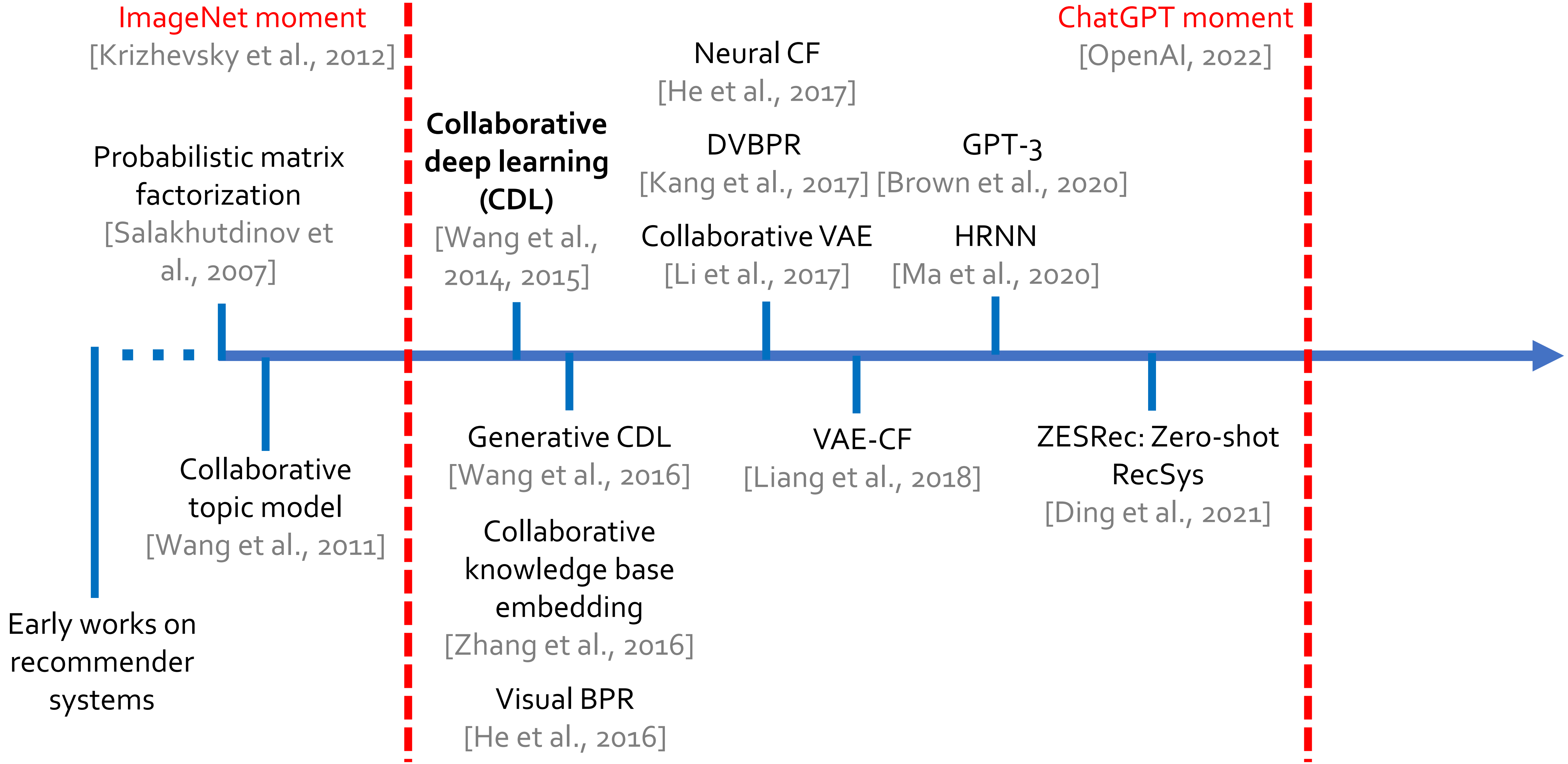
- Architecture similar to neural CF (NCF)
- Concatenate embeddings of video watches, search tokens, age, gender, etc.

Deep Recommender Systems in the Industry

- **Microsoft:** Collaborative knowledge base embedding [KDD 2016]
- **Netflix:** VAE for collaborative filtering [WWW 2018]
- **Amazon:** Hierarchical RNN [KDD 2020]
- **Google:** Wide and deep learning [DLRS 2016]
- **YouTube (Google):** Deep neural nets for YouTube recommendations [RecSys 2016]
- **Airbnb:** Applying deep learning to Airbnb search [KDD 2019]
- **LinkedIn:** Talent search and recommendation systems at LinkedIn [SIGIR 2018]
- **Meta:** Deep learning recommendation model, [arXiv 2019]
- **Twitter/X:** Relevance ranking for real-time tweet search [CIKM 2020]

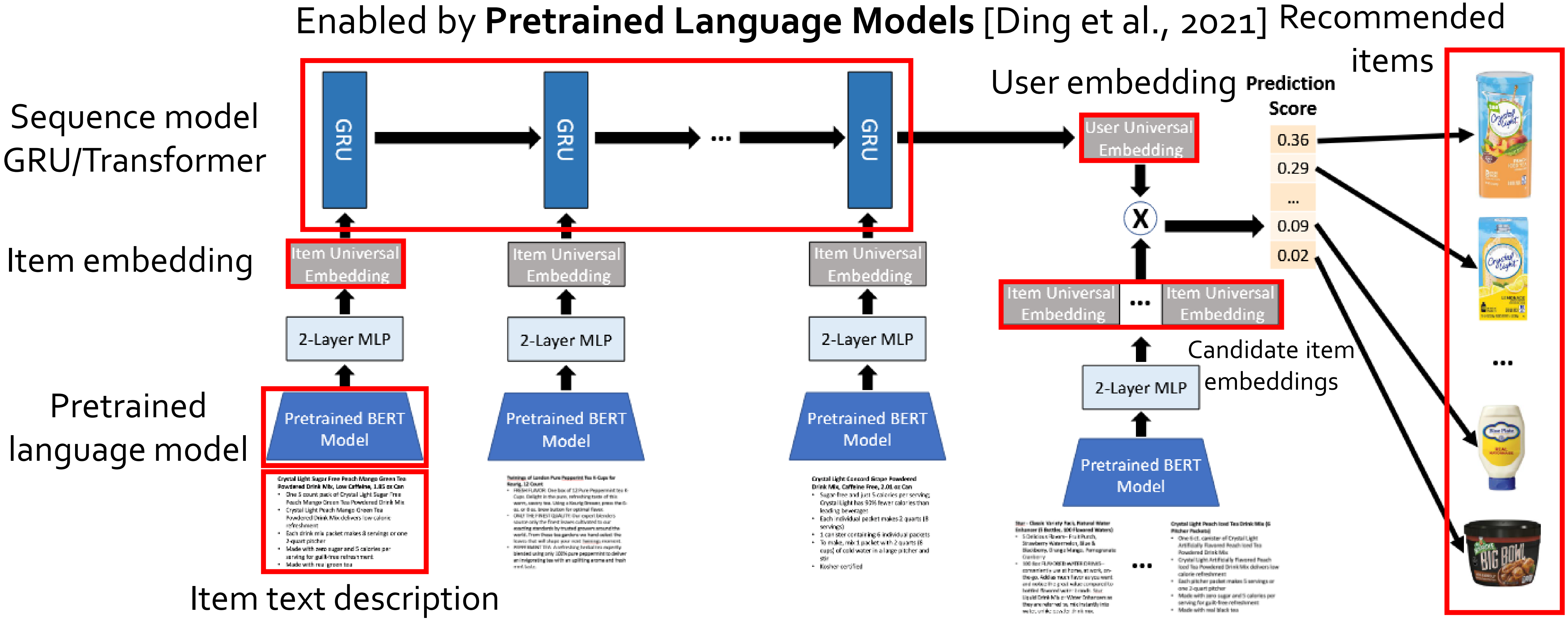
What’s Next

Timeline: Deep Recommender Systems



ZESRec: ZEro-Shot Recommender Systems

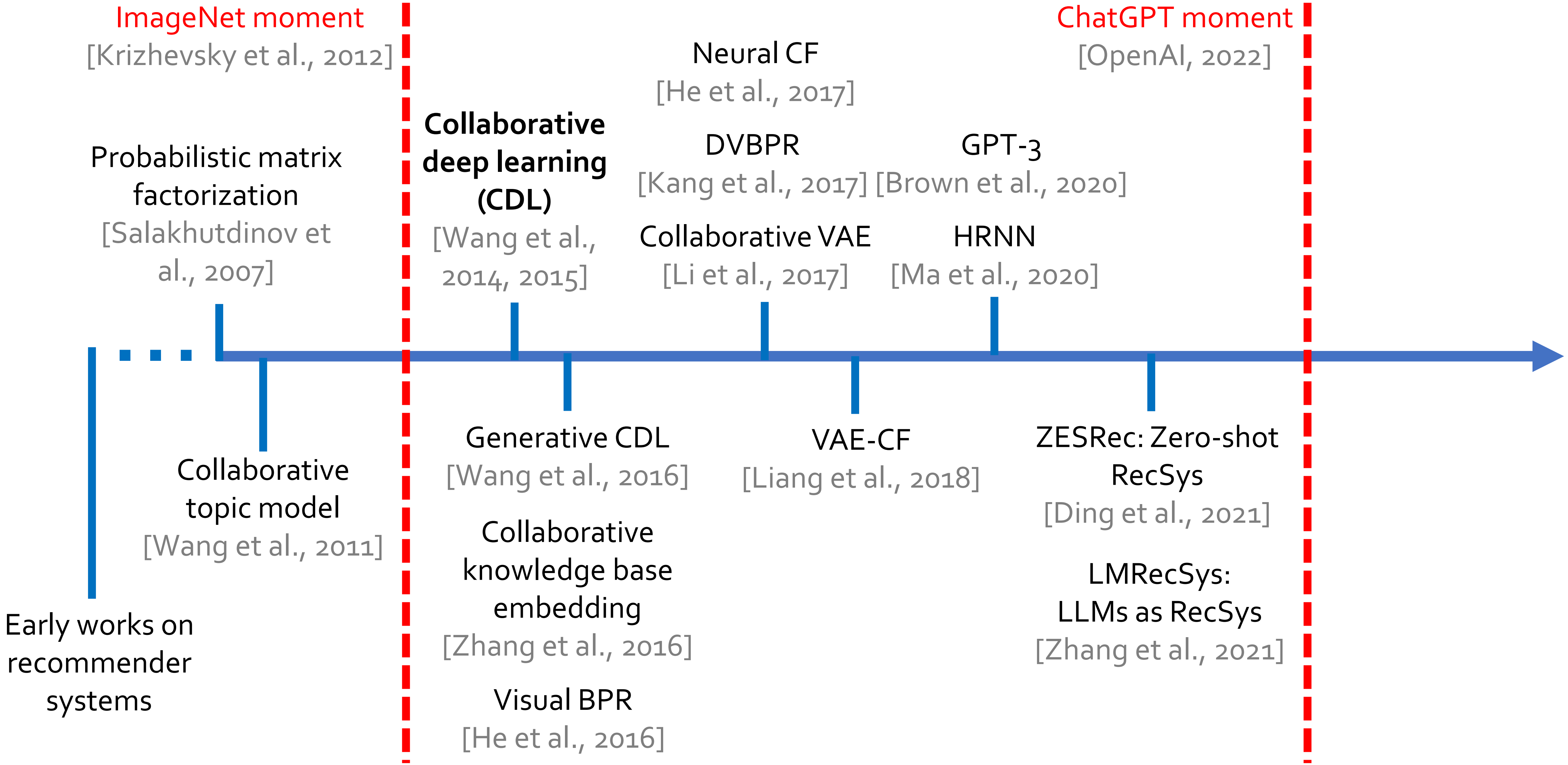
Enabled by **Pretrained Language Models** [Ding et al., 2021]



- Everything is grounded in natural **language** (w/ language models); language as universal item ID
- Enable **zero-shot** recommendation (both new users and new items)

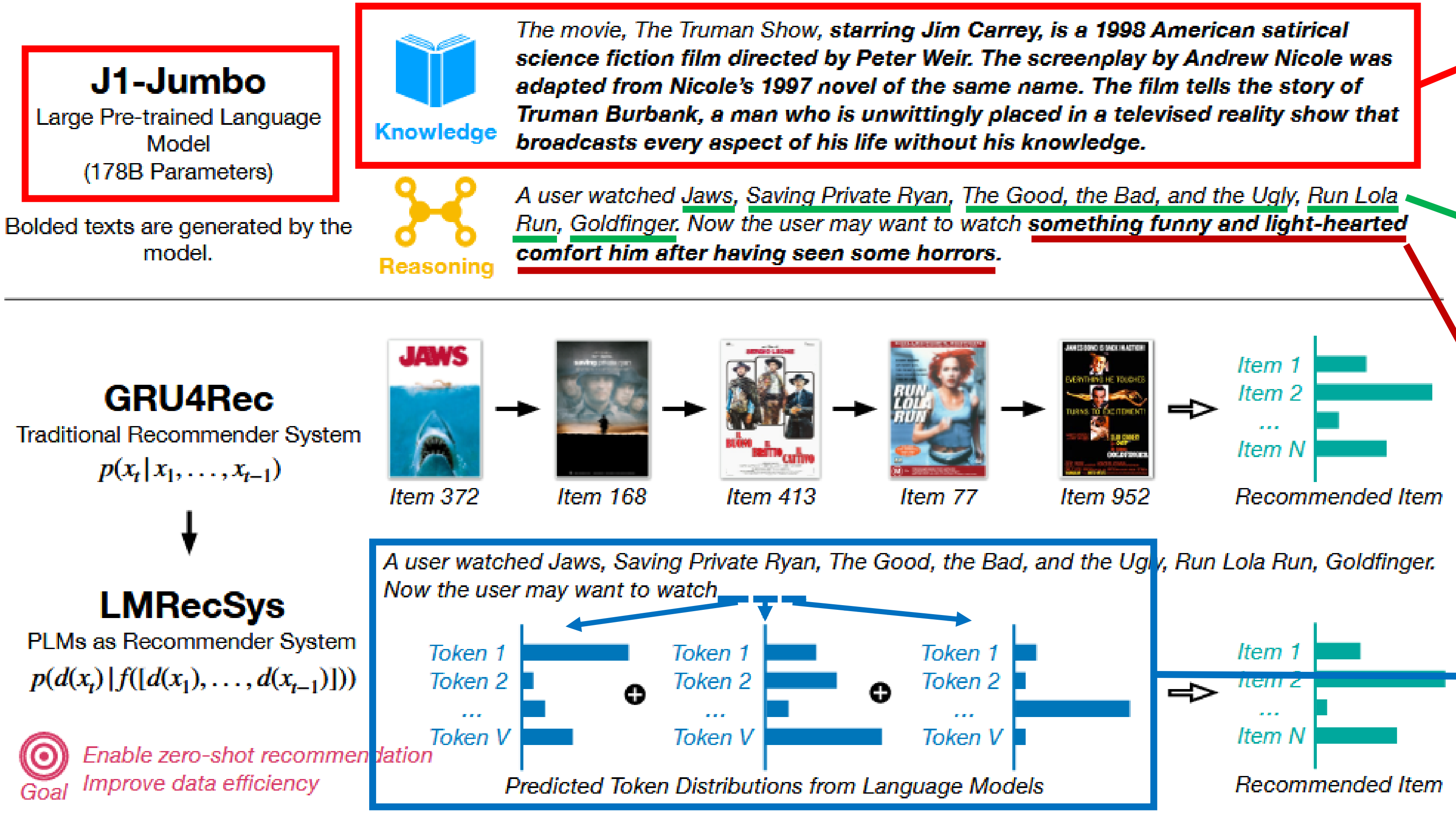
[Zero-Shot Recommender Systems. DMDWW. *arXiv* 2021, *ICLR-W* 2022]

Timeline: Deep Recommender Systems



LMRecSys: Large Language Models (LLMs)

as Zero-Shot Recommender Systems [Zhang et al., 2021]



Knowledge: Feed movie descriptions into the LLM

History: List user's watched movies using natural language; feed them to the LLM

Reasoning: LLM reasons about user preference

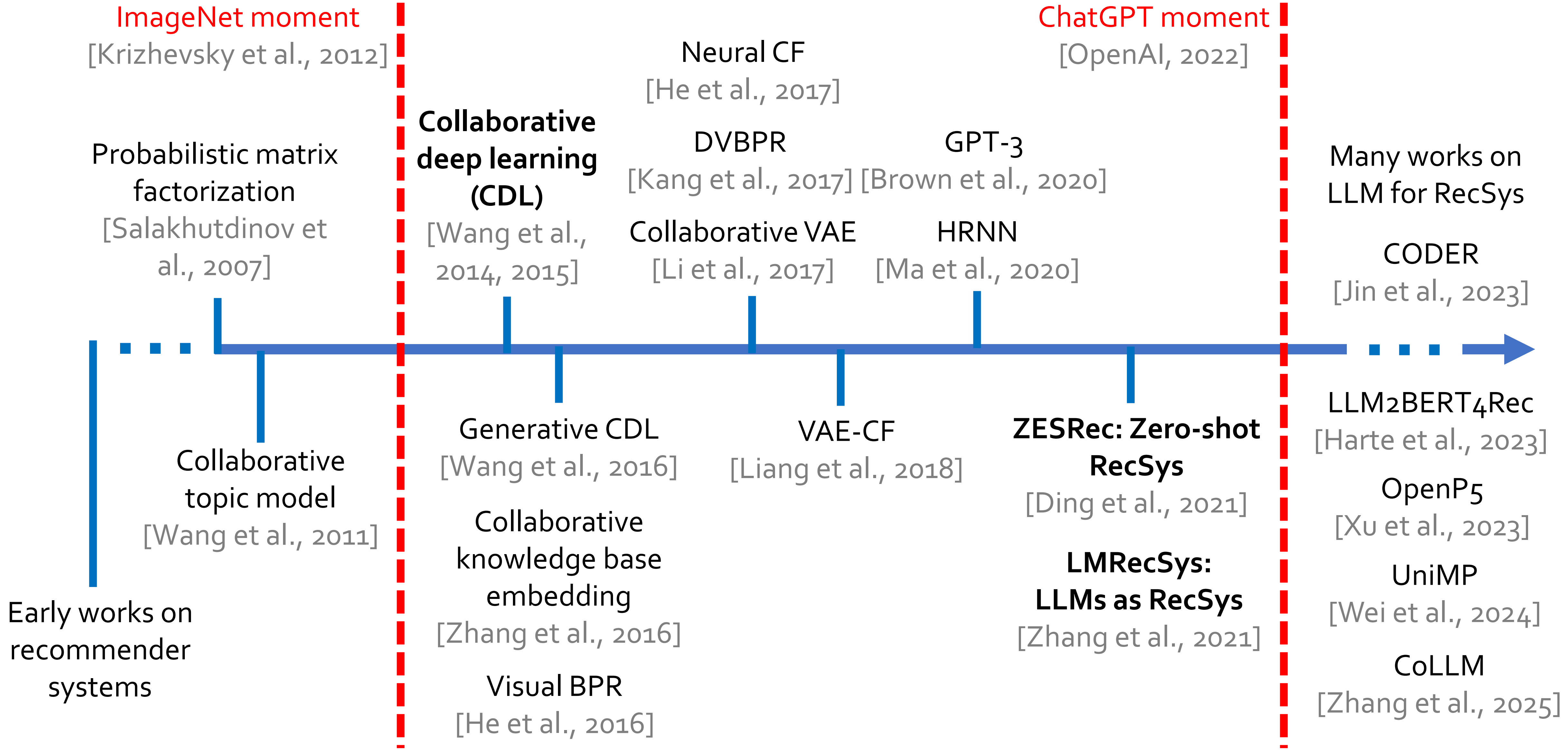
Recommendation: LLM generates movie names token by token

Addressed challenges

- Linguistic bias: Grammar words dominate token probabilities
- Out of vocabulary: Recommend movies that do not exist, i.e., hallucination

[Language Models as Recommender Systems: Evaluations and Limitations. ZDSMZDW. *NeurIPS-W* 2021]

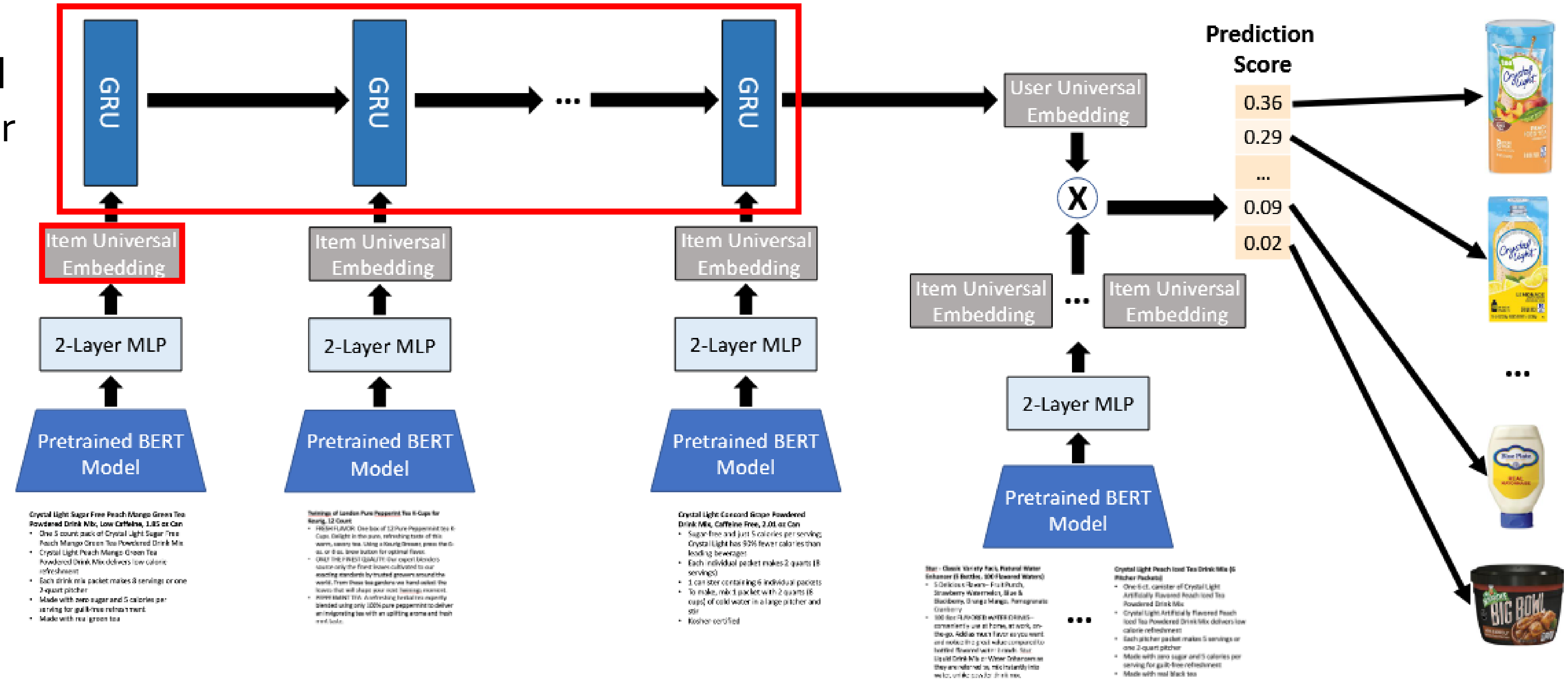
Timeline: Deep Recommender Systems



LLM2BERT4Rec [Harte et al., 2023] as Improved ZESRec

Sequence model GRU/Transformer

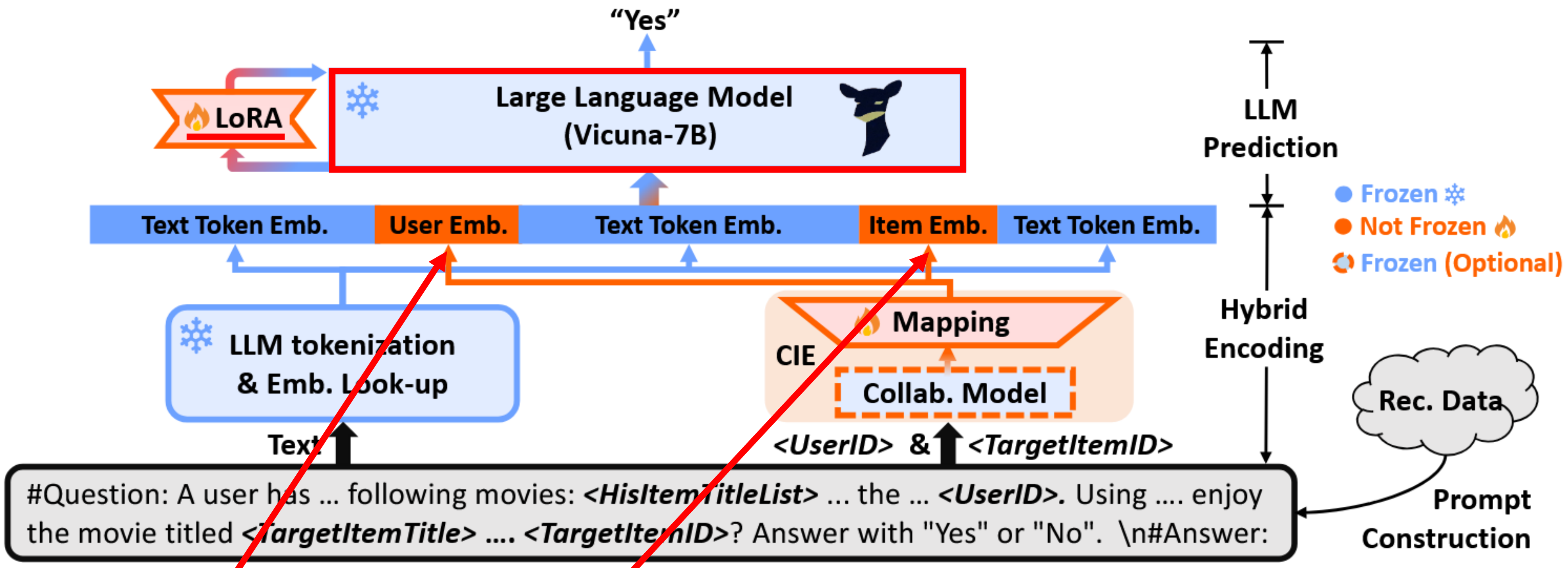
Item embedding

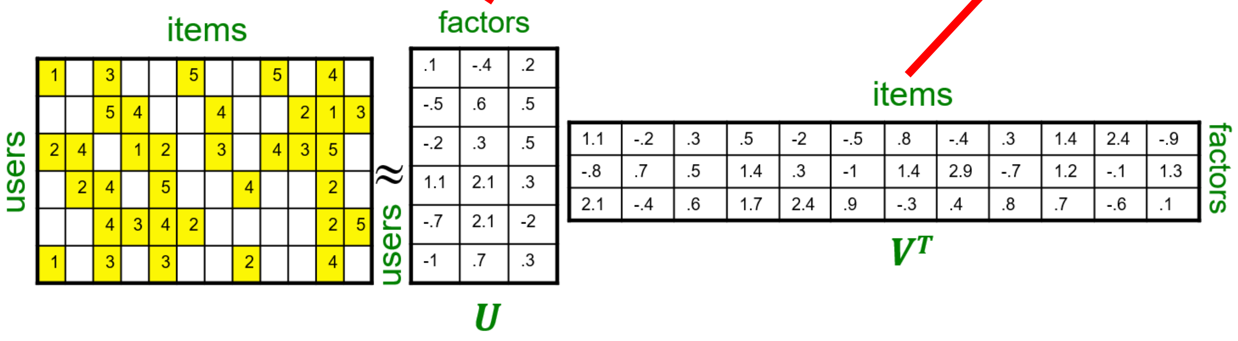


- **Sequence model:** Replace GRU (RNN) in ZESRec [Ding et al., 2021] with a Transformer
- **Embedding model:** Replace BERT in ZESRec with OpenAI's text embedding

[Leveraging Large Language Models for Sequential Recommendation. HZLKJF. *RecSys-EA* 2023]

CoLLM [Zhang et al., 2025]: Incorporating Collaborative Filtering Information into LLMs





- Use Vicuna-7B as the **LLM**
- User & item embeddings from **collaborative filtering**
- **Finetune** with LoRA [Hu et al., 2021]

[CoLLM: Integrating Collaborative Embeddings into Large Language Models for Recommendation. ZFZBWH. *TKDE* 2025]

To Wrap Up

The Next Decade of Recommender Systems

From task-specific recommenders to **general** recommenders
(e.g., foundation models and agentic models)

- **Zero-shot** and few-shot recommendation to become the norm
  - Early work: From CDL [Wang et al., 2015] to ZESRec [Ding et al., 2021]
- Tighter integration between recommendation and **generation**
  - Early work: DVBPR [Kang et al., 2017], LMRecSys [Zhang et al., 2021]
- Generation introduces **new risks** (hallucination, toxicity, etc.)
  - Survey: [Deldjoo et al., 2025]

Acknowledgement

Graduate students

Zihao Xu
Hengyi Wang
Jingquan Yan
Haizhou Shi
Yi Wang
Dhaivat Patel
Weiyi Qin
Ziyan Wang
Hao He
Yuzhe Yang
Paven Kumar Velaga
Zhiqing Hong
Song Wen
Yunhe Gao
Monaal Sanghvi
Chengzhi Mao
Guangyuan Hao
Hengguan Huang
Guang-He Lee
Shiwei Tan

Undergraduate students

Tianyi Liu
Thea Xu
Rohit Amarnath
Sarthak Jain

Collaborators

Dit-Yan Yeung
Dimitris Metaxas
Yongfeng Zhang
Ye Wang
Kun Zhang
Paul Kantor
Lazaros Gallos
Junfeng Yang
Elias Bareinboim
Tong Zhang
Eric Xing
Tommi Jaakkola
Naiyan Wang
Jie Gao
Bernie Wang
Hao Ding
Yingci Liu
Jacob Feldman
Yifei Ma
Carl Vondrick
Zachary Lipton
Anoop Deoras
Dina Katabi
... many more!

Family and friends ...

The Next Decade of Recommender Systems

From task-specific recommenders to **general** recommenders
(e.g., foundation models and agentic models)

- **Zero-shot** and few-shot recommendation to become the norm
    - Early work: From CDL [Wang et al., 2015] to ZESRec [Ding et al., 2021]
- Tighter integration between recommendation and **generation**
    - Early work: DVBPR [Kang et al., 2017], LMRecSys [Zhang et al., 2021]
- Generation introduces **new risks** (hallucination, toxicity, etc.)
    - Survey: [Deldjoo et al., 2025]

Home page: www.wanghao.in