Data Analytics for Recommender Systems

- More detailed project description will be uploaded to Canvas

Build up your team

- At most 3 persons per team
- Each team works on a project and give a presentation together
- Project due date is 11/25 (1-month)
- Selected presentations on 12/2, 12/4, 12/9, 12/11. Presentations will have bonus points
  - See Canvas announcement for details of bonus policy
- Submission
  - Code, a project report, and presentation slides

Project Description

- Dataset
  - Amazon (24 product categories: 24 sub-datasets)
  - Only need to choose **one** sub-dataset
- Basic data format (Actually it is a sparse matrix)
  - **user-id**: which is denoted as “reviewerID” in the dataset
  - **product-id**: which is denoted as “asin” in the dataset
  - **rating**: a 1-5 integer star rating, which is the rating that the user rated on the product, it is denoted as “overall” in the dataset
  - **review**: a piece of review text, which is the review content that the user commented about the product, it is denoted as “reviewText” in the dataset
  - **title**: the title of the review, which is denoted as “summary” in the dataset
  - **timestamp**: time that the user made the rating and review
  - **image**: image of the product
  - **description**: a piece of text description of the product
  - **…....**

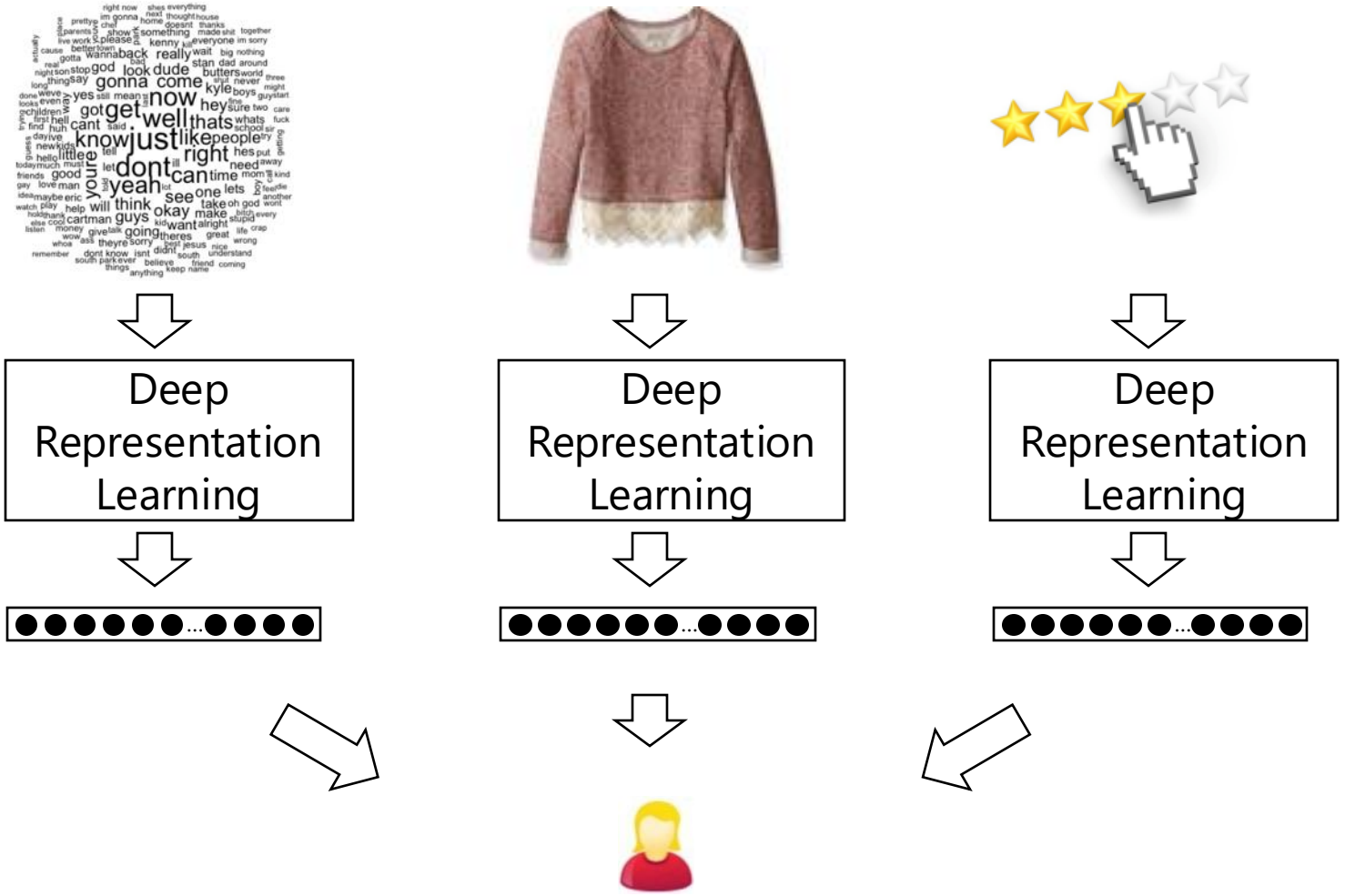
Project Tasks

- Basic Requirements
  - Step 1: Data prepressing and split, create a training dataset and a testing dataset for experiment
  - Step 2: Rating prediction, develop an algorithm to predict the ratings in the testing set based on the information (ratings and others) in the training dataset, and evaluate the predictions based on MAE and RMSE.
  - Step 3: Item Recommendation, construct a recommendation list for each user, and then evaluate the recommendation quality based on precision, recall, F-measure, and NDCG.
- Optional for bonus points
  - Responsible AI Perspectives
    - Explainability, Fairness, Controllability, etc.
  - See Canvas announcement for details of bonus policy

An example project implementation

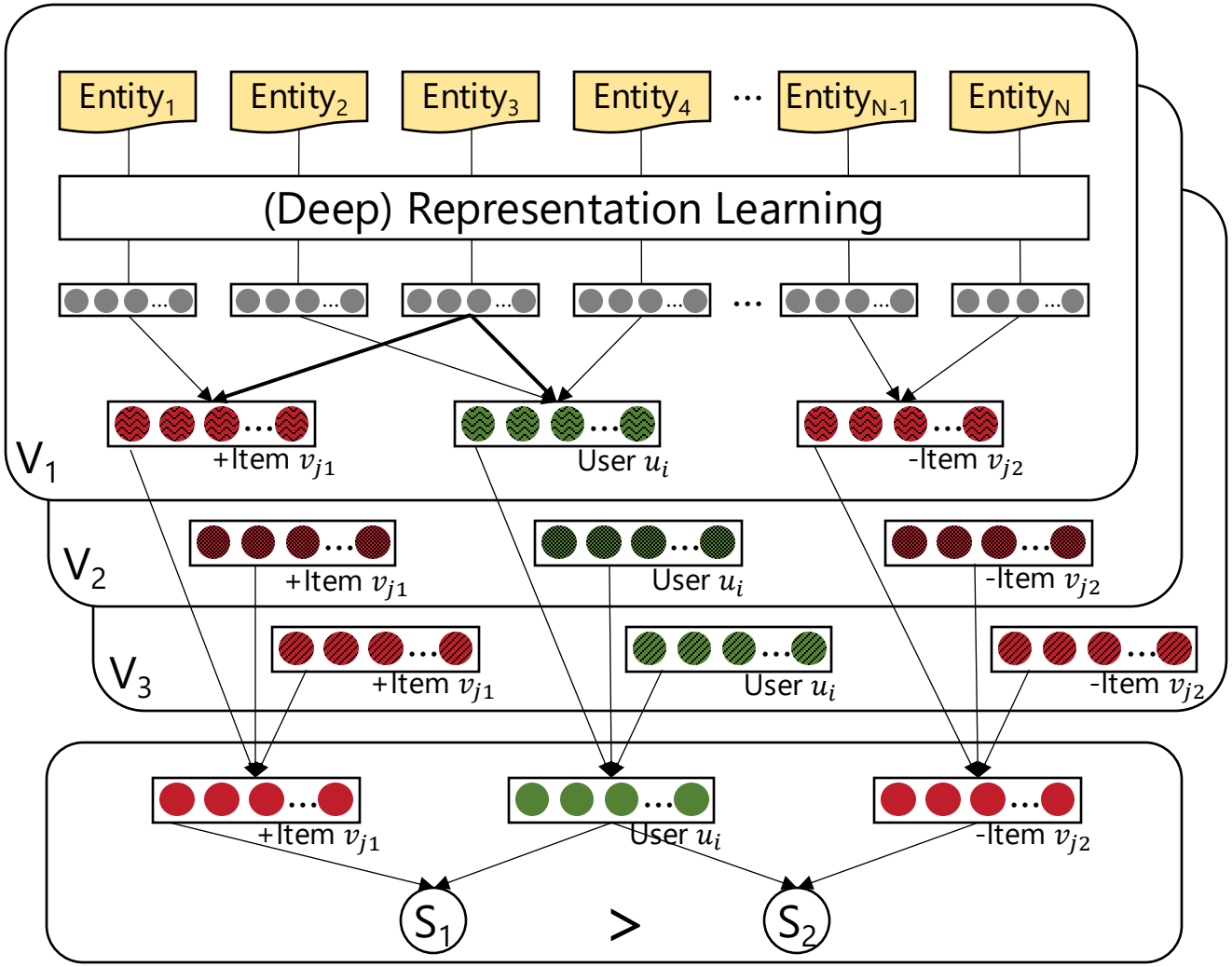
Joint Representation Learning for Recommendation with Heterogeneous Information Sources

Deep Representation Learning



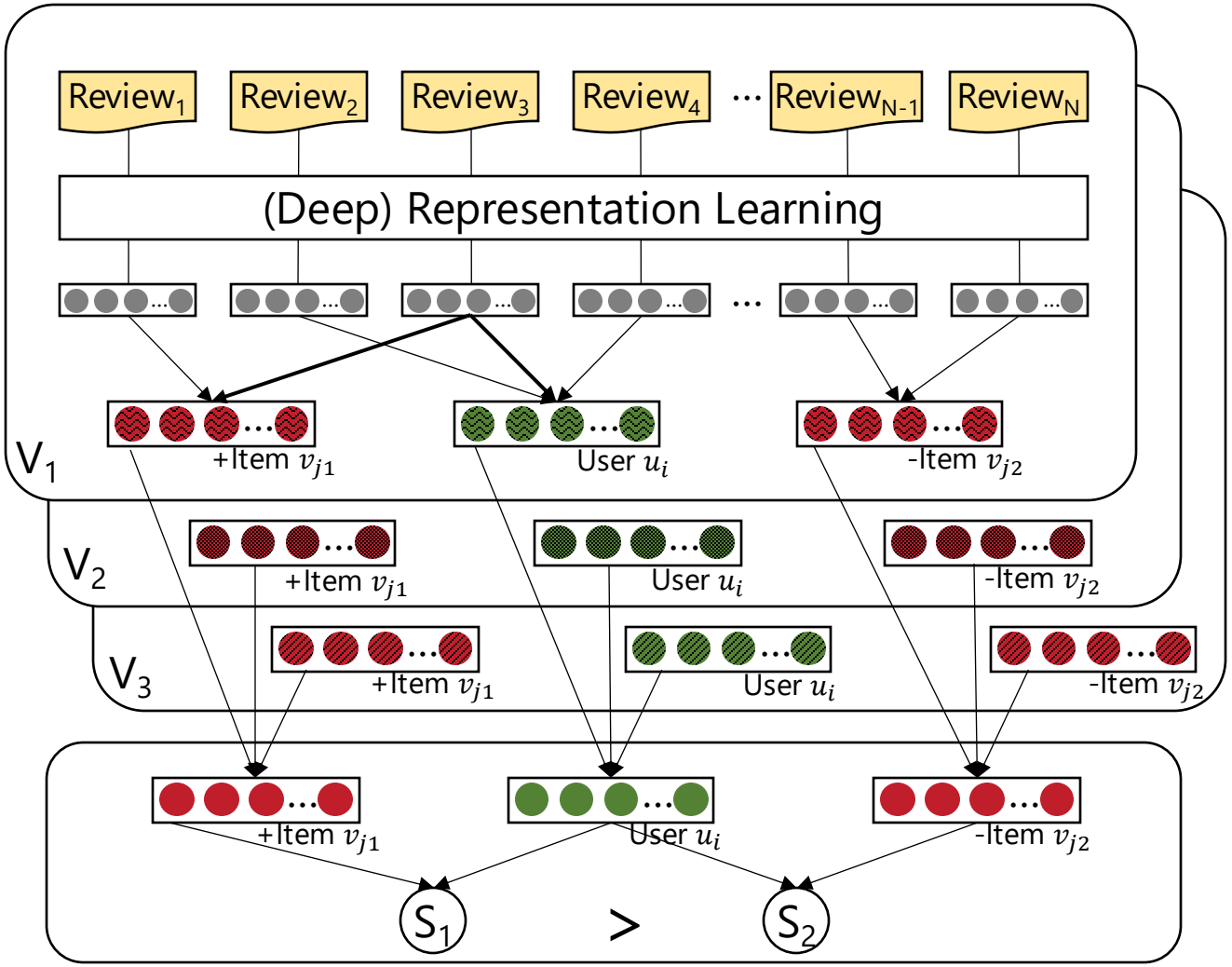
Joint Representation Learning

A Multi-View Machine Learning Framework with Heterogeneous Information Sources



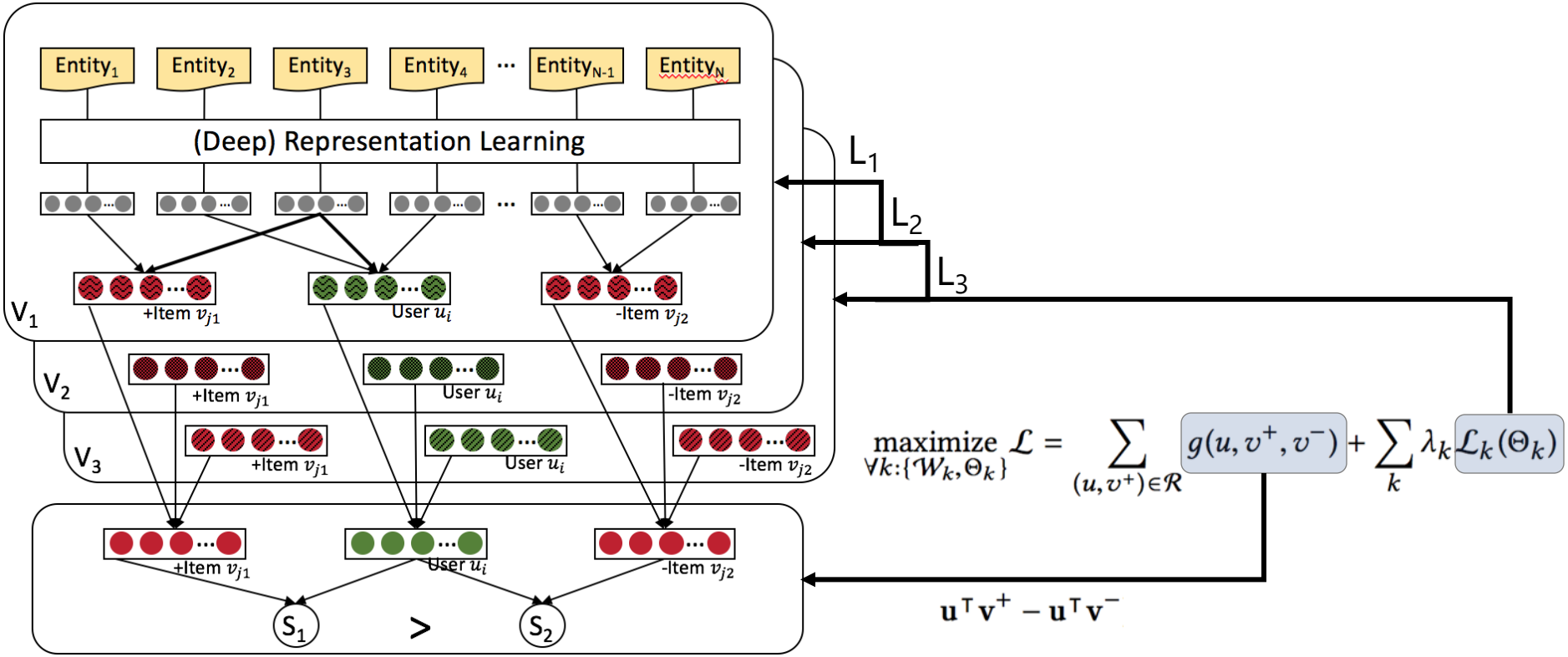
Joint Representation Learning

A Multi-View Machine Learning Framework with Heterogeneous Information Sources

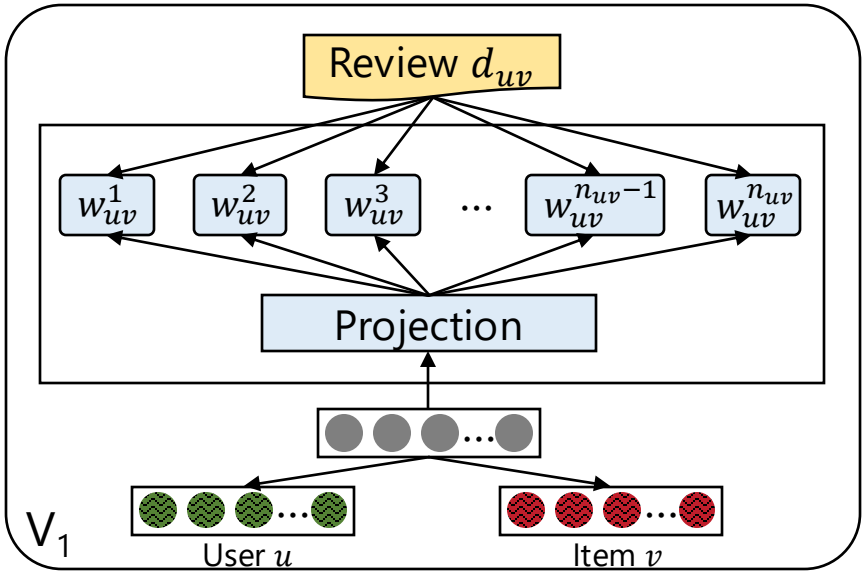


Joint Representation Learning

A Multi-View Machine Learning Framework with Heterogeneous Information Sources



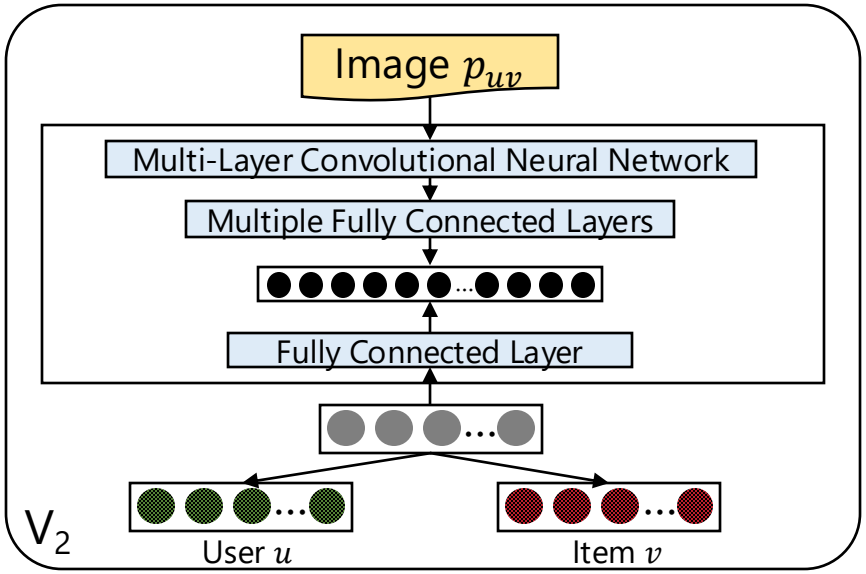
Modeling of Textual Reviews (View V_1)



$$\mathcal{L}_1(\mathbf{w}, \mathbf{d}_{uv}) = \sum_{w \in \mathcal{V}} \sum_{(u,v) \in \mathcal{R}} \mathrm{f}_{w, d_{uv}} \log \sigma(\mathbf{w}^\intercal \mathbf{d}_{uv}) + \sum_{w \in \mathcal{V}} \sum_{(u,v) \in \mathcal{R}} \mathrm{f}_{w, d_{uv}} \left(t \cdot \mathbb{E}_{w_N \sim P_\mathcal{V}} \log \sigma(-\mathbf{w}_N^\intercal \mathbf{d}_{uv}) \right)$$

Word Embedding by Paragraph Vector Learning

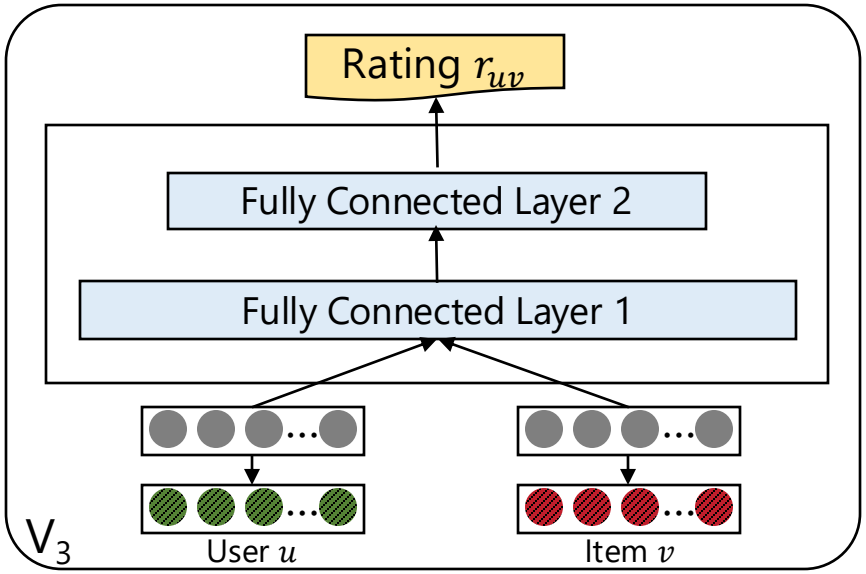
Modeling of Visual Images (View V_2)



$$\mathcal{L}_2(A, \mathbf{b}, \mathbf{p}_{uv}) = \sum_{(u,v)\in\mathcal{R}} \left(\phi(A \cdot \mathbf{p}_{uv} + \mathbf{b}) - \vec{p}_{uv}\right)^2$$

Image Embedding by Convolutional Neural Network

Modeling of Numerical Ratings (View V_3)

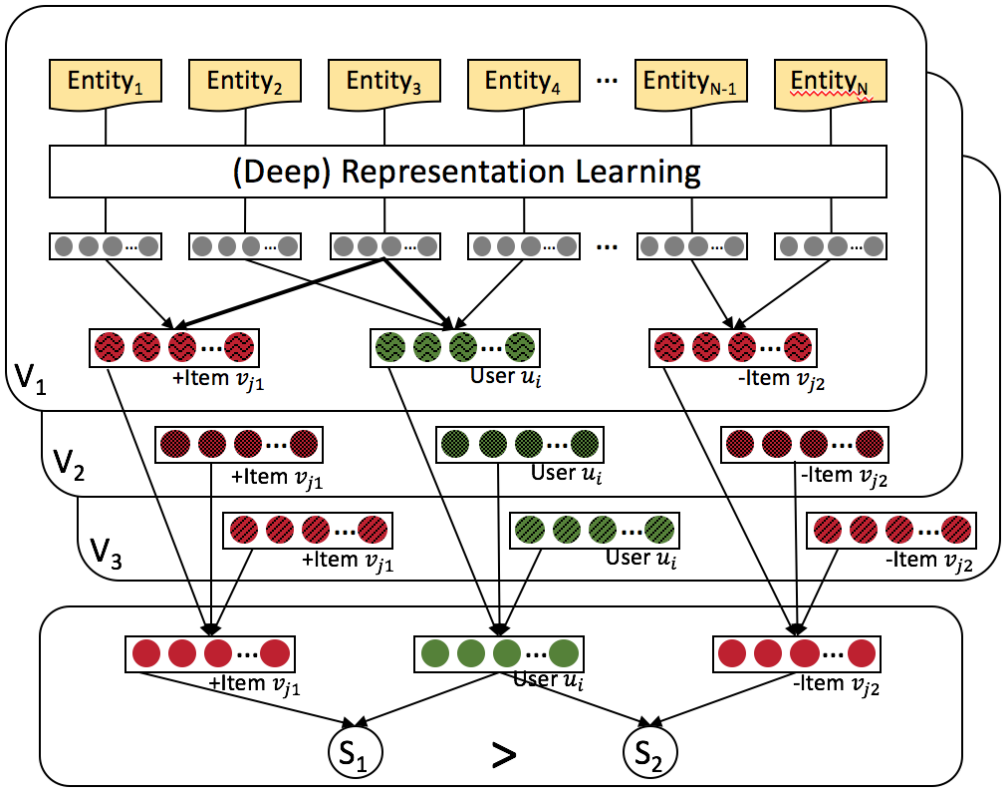


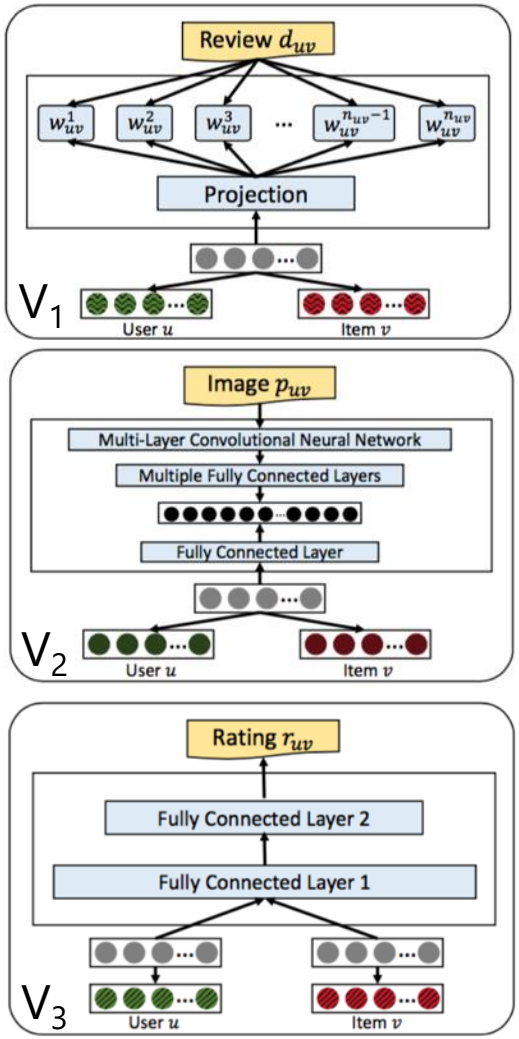
$$\hat{r}_{uv} = \phi\left(U_2 \cdot \phi\left(U_1(\mathbf{r}_u \odot \mathbf{r}_v) + \mathbf{c}_1\right) + \mathbf{c}_2\right)$$

$$\mathcal{L}_3(U_1, U_2, \mathbf{c}_1, \mathbf{c}_2, \mathbf{r}_u, \mathbf{r}_v) = \sum_{(u,v)\in\mathcal{R}} (\hat{r}_{uv} - r_{uv})^2$$

Rating Embedding by Fully Connected Layers

Joint Representation Learning





$$\underset{\forall k:\{\mathcal{W}_k,\Theta_k\}}{\text{maximize}}\ \mathcal{L} = \sum_{(u,v^+)\in\mathcal{R}} g(u,v^+,v^-) + \sum_k \lambda_k \mathcal{L}_k(\Theta_k)$$

Extendable to New Information Sources

$$\underset{\forall k:\{\mathcal{W}_k,\Theta_k\}}{\text{maximize}}\ \mathcal{L} = \sum_{(u,v^+)\in\mathcal{R}} g(u,v^+,v^-) + \sum_k \lambda_k \mathcal{L}_k(\Theta_k)$$

$$\frac{\partial \mathcal{L}}{\partial \Theta_k} = \lambda_k \frac{\partial \mathcal{L}_k}{\Theta_k}$$

$$\frac{\partial \mathcal{L}}{\partial \mathcal{W}_k} = \sum_{(u,v)\in\mathcal{R}} \sigma'(\mathbf{u}_k^\top \mathbf{v}_k^+ - \mathbf{u}_k^\top \mathbf{v}_k^-)\left((\mathbf{v}_k^+ - \mathbf{v}_k^-)^\top \frac{\partial \mathbf{u}_k}{\partial \mathcal{W}_k} + \mathbf{u}_k^\top \frac{\partial \mathbf{v}_k^+}{\partial \mathcal{W}_k} - \mathbf{u}_k^\top \frac{\partial \mathbf{v}_k^-}{\partial \mathcal{W}_k}\right)$$

Gradient on parameters from view k

Only contains parameters of view k itself
Independent from other views

Experimental Setup

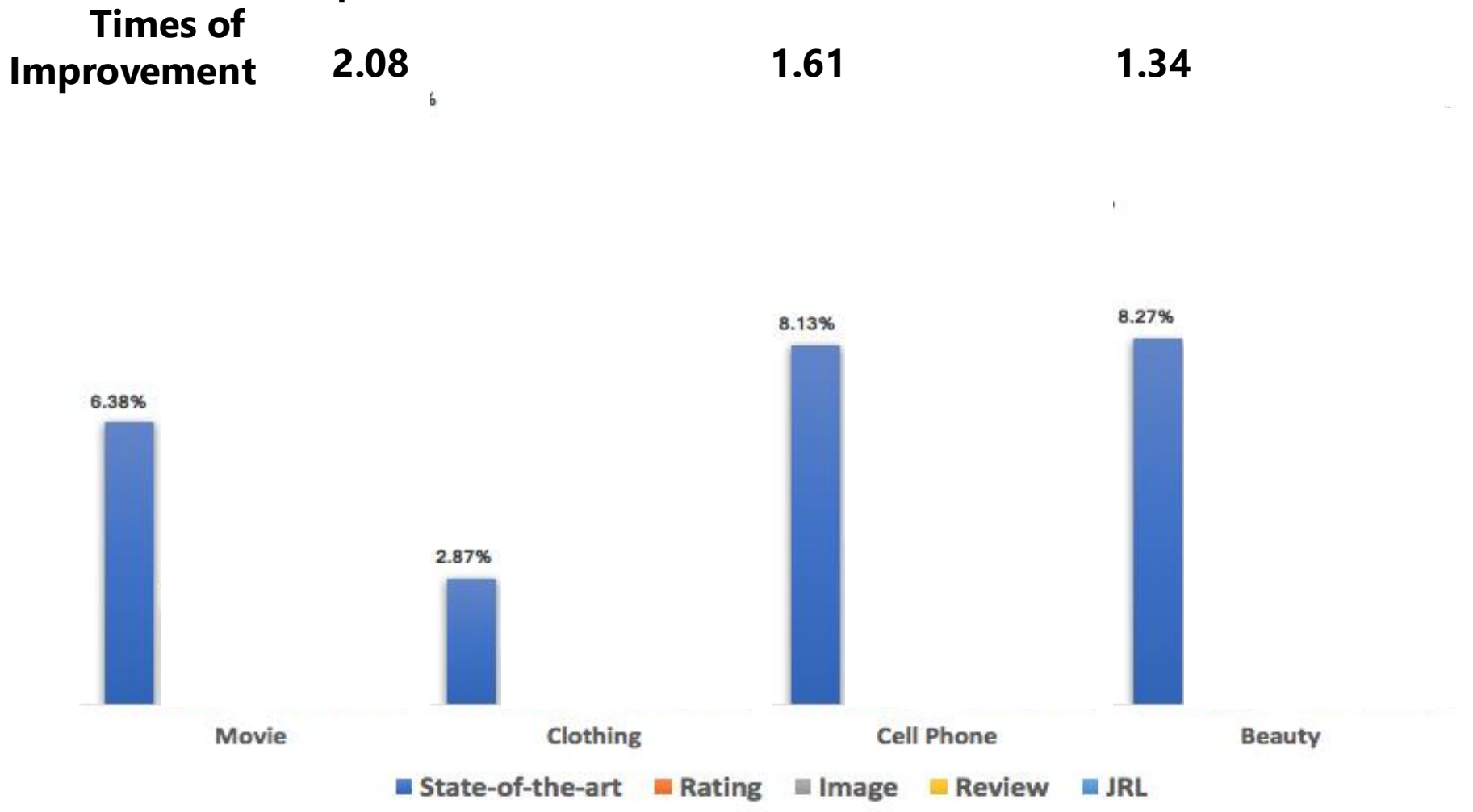
- We take the Amazon dataset for experiments

Dataset	#users	#items	#interactions	sparsity
Movies	123,960	50,052	1,697,533	0.0274%
CDs	75,258	64,421	1,097,592	0.0226%
Clothing	39,387	23,033	278,677	0.0307%
Cell Phones	27,879	10,429	194,439	0.0669%
Beauty	22,363	12,101	198,502	0.0734%

Baseline methods

- Baseline Methods
  - BPR: Bayesian Personalized Ranking with implicit feedback.
  - HFT: Hidden factor and topics model integrated into BPR, because the original model is designed for rating prediction. It relies on reviews.
  - VBPR: Visual Bayesian Personalized Ranking method based on images.

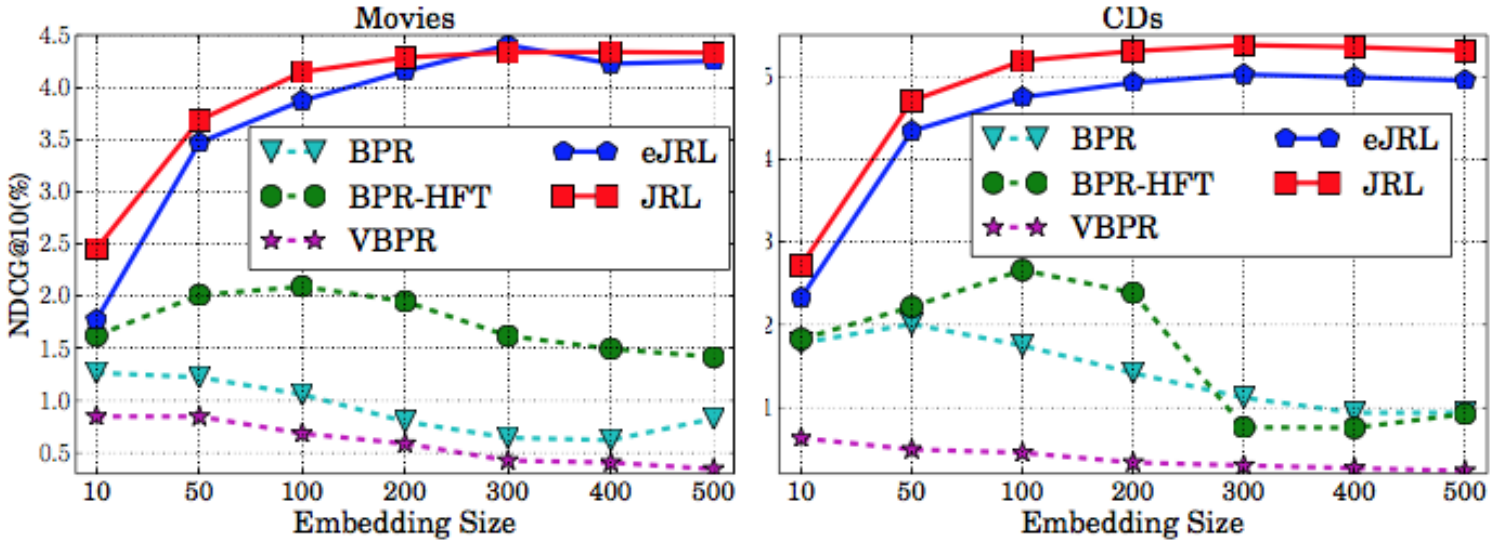
Compare with Shallow Models on NDCG



State-of-the-art = Best of {BPR (by rating), HFT (by review), VBPR (by image)}

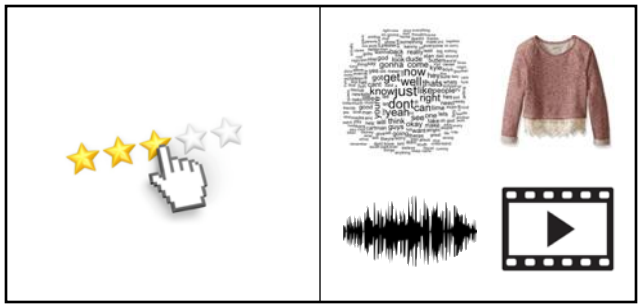
Impact of Embedding Size

- Tune embedding size from 10 to 500



- Observations
  - Our model: performance keeps increasing until about 300 dimension, and does not decrease when using more dimensions.
  - Shallow baselines: best performance when dimension less than 100, performance decreases when using more dimensions.
- What we learn
  - Deep models can capture more complex interaction from data
  - Learning ability of shallow models is limited and may over-fit if model complexity is too high.

Summary and Future Works



Use various heterogeneous data to provide **Significant personalization performance.**

- Develop a Joint Representation Learning framework for recommendation based on heterogeneous information sources.
- Not only ratings, review, and images, but also extendable to new information sources.
- Achieved significant improve for top-N recommendation.

- Future work
  - Consider other representation learning architectures for recommendation
  - Consider other information sources for recommendation


THANKS