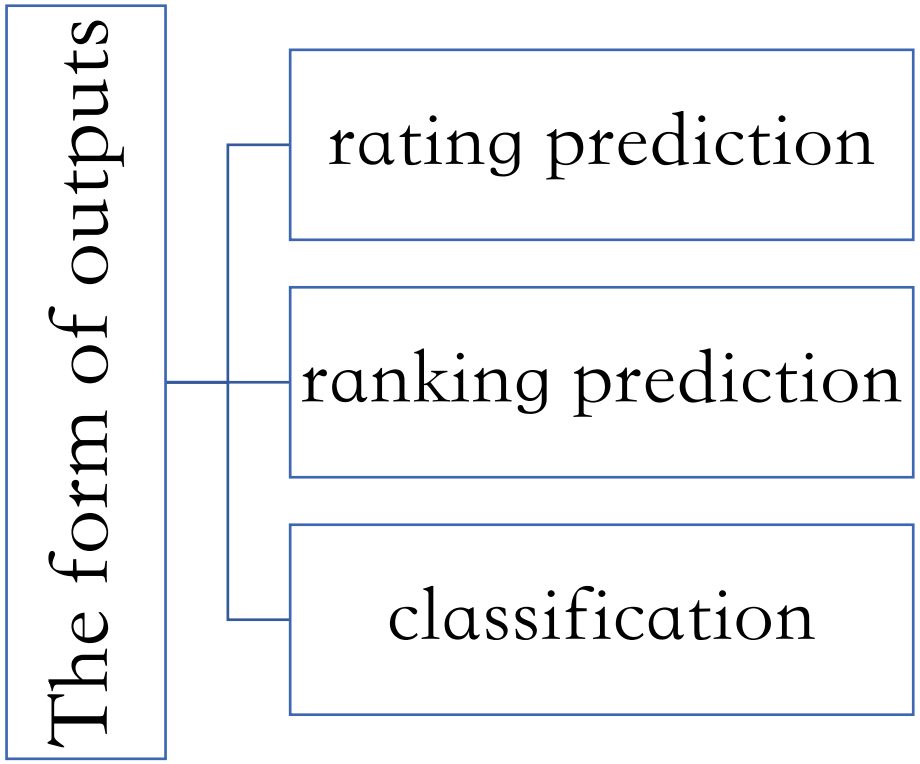
Deep Learning based Collaborative Filtering

——The Survey of recommender system

Backgrounds

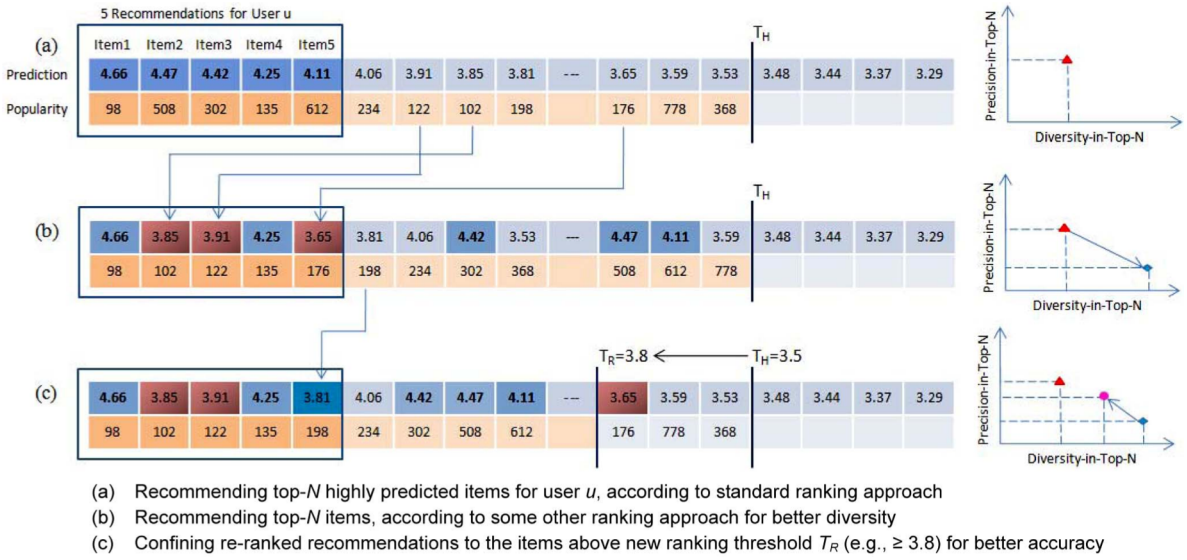
Recommender system

Recommender system is used for estimating users' preference in items they have not seen. There mainly exis
types of recommendation tasks based on the forms of outputs.

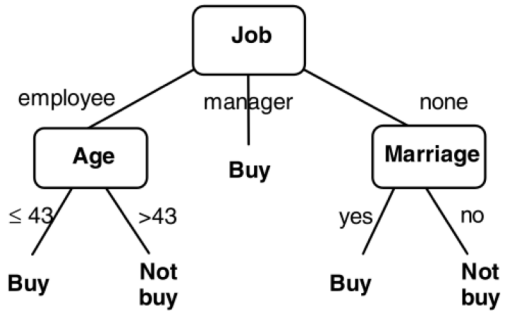


	K-PAX	Life of Brian	Memento	Notorious
Alice	4	3	2	4
Bob	∅	4	5	5
Cindy	2	2	4	∅
David	3	∅	5	2

a fragment of rating matrix for a movie recommender system



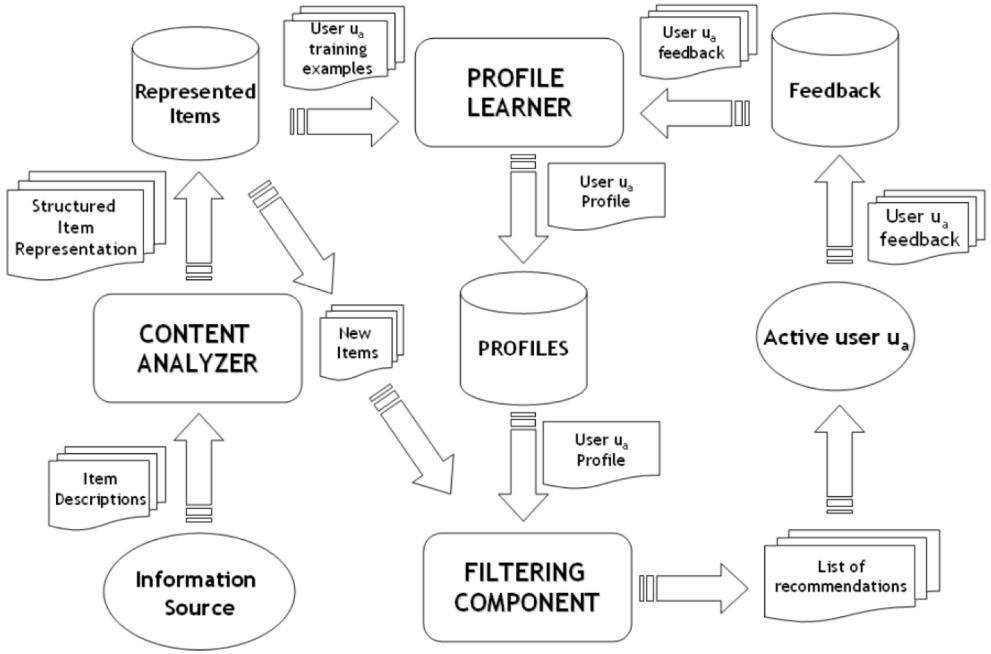
(a) Recommending top-*N* highly predicted items for user *u*, according to standard ranking approach
(b) Recommending top-*N* items, according to some other ranking approach for better diversity
(c) Confining re-ranked recommendations to the items above new ranking threshold T_R (e.g., ≥ 3.8) for better accuracy



a decision tree for purchase assessmen

The category of recommendation models

Collaborative filtering



Content-based recommender system

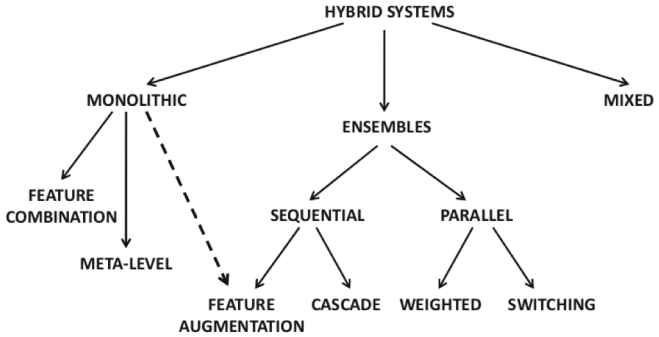
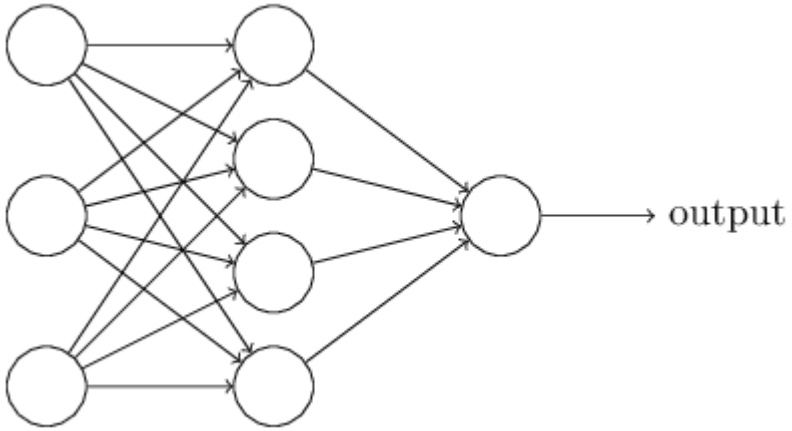


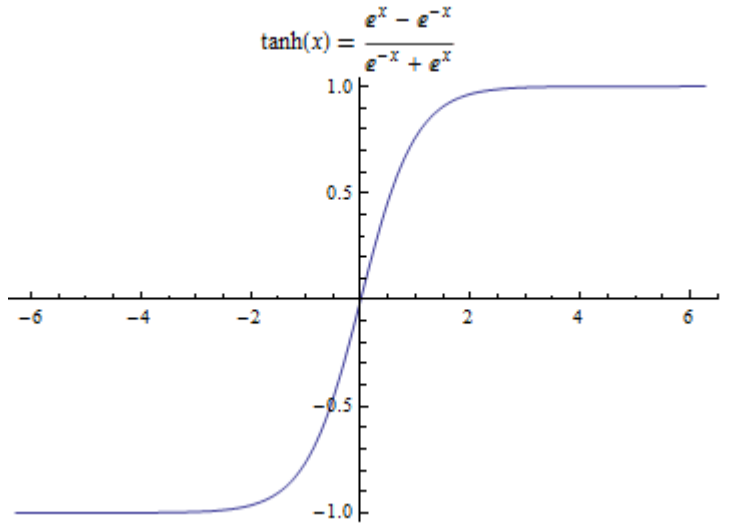
Figure 6.1: The taxonomy of hybrid systems

Hybrid recommender system based on the type of input data

proposed algorithm

Deep Learning based Recommender System

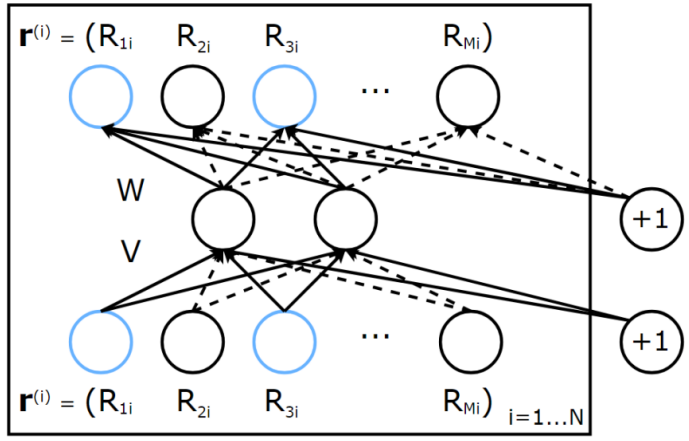





```
if w*x > 0:
    output = 1
else:
    output = -1
```

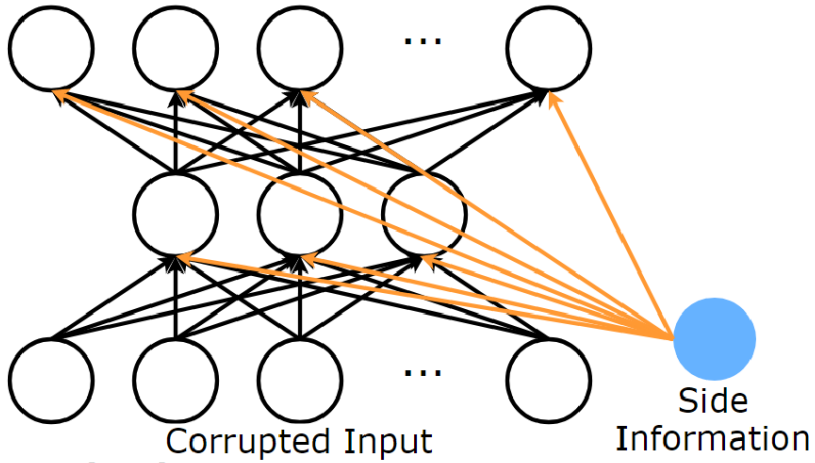
Multilayer Perception based recommender system

Deep Learning based Recommender System



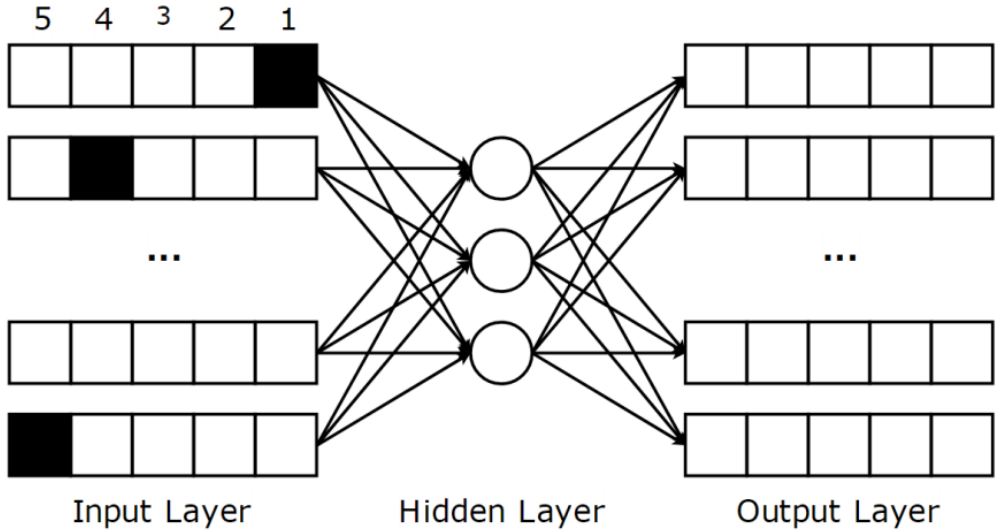
I-AutoRec

$$\arg\min_{\theta} \sum_{i=1}^{N} \| \mathbf{r}^{(i)} - h(\mathbf{r}^{(i)}; \theta) \|_{\mathcal{O}}^2 + \lambda \cdot Regularization$$

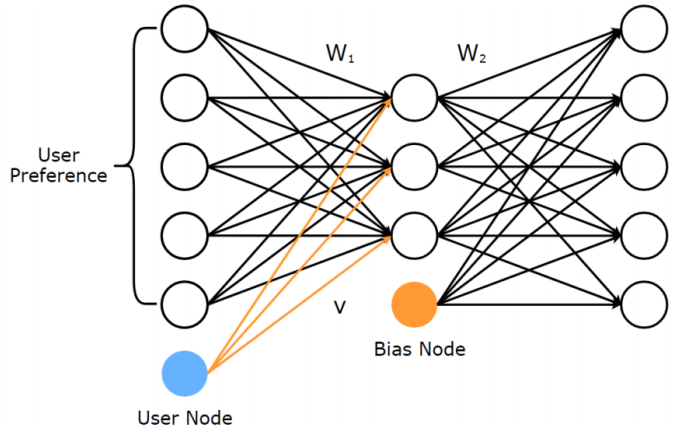


CFN

$$\mathcal{L} = \alpha \left(\sum_{i \in I(\mathcal{O}) \cap I(\mathcal{C})} [h(\tilde{\mathbf{r}}^{(i)}) - \mathbf{r}^{(i)}]^2 \right) + \beta \left(\sum_{i \in I(\mathcal{O}) \setminus I(\mathcal{C})} [h(\tilde{\mathbf{r}}^{(i)}) - \mathbf{r}^{(i)}]^2 \right) + \lambda \cdot Regularization$$



AFN

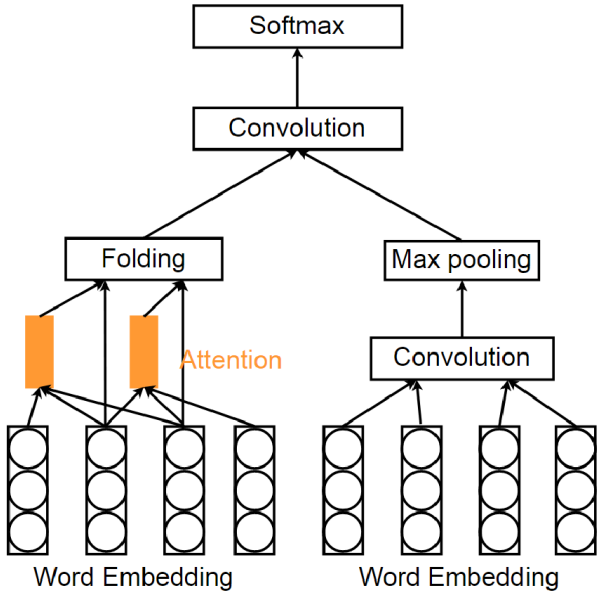


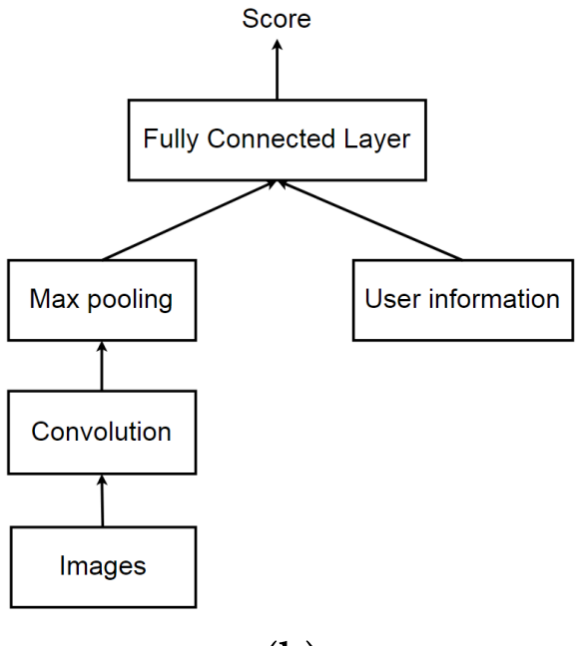
CDFN

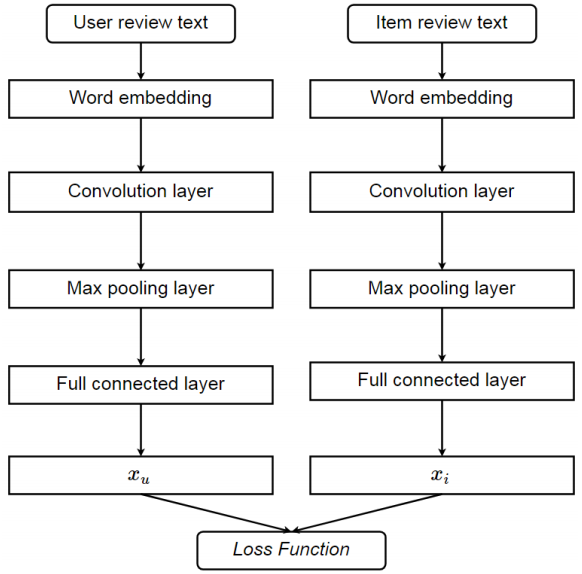
$$\arg\min_{W_1, W_2, V, b_1, b_2} \frac{1}{M} \sum_{u=1}^{M} \mathbf{E}_{p(\tilde{\mathbf{r}}_{pref}^{(u)} | \mathbf{r}_{pref}^{(u)})} [\ell(\tilde{\mathbf{r}}_{pref}^{(u)}, h(\tilde{\mathbf{r}}_{pref}^{(u)}))] + \lambda \cdot Regularization$$

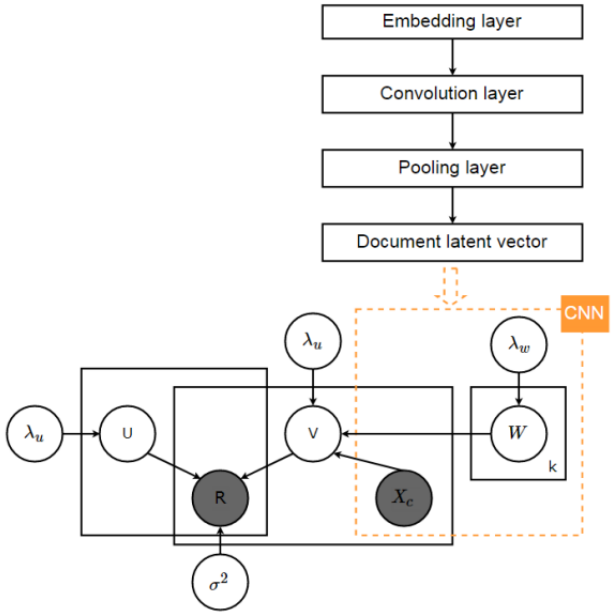
Auto-encoder based recommender system

Deep Learning based Recommender System









Attention based CNN

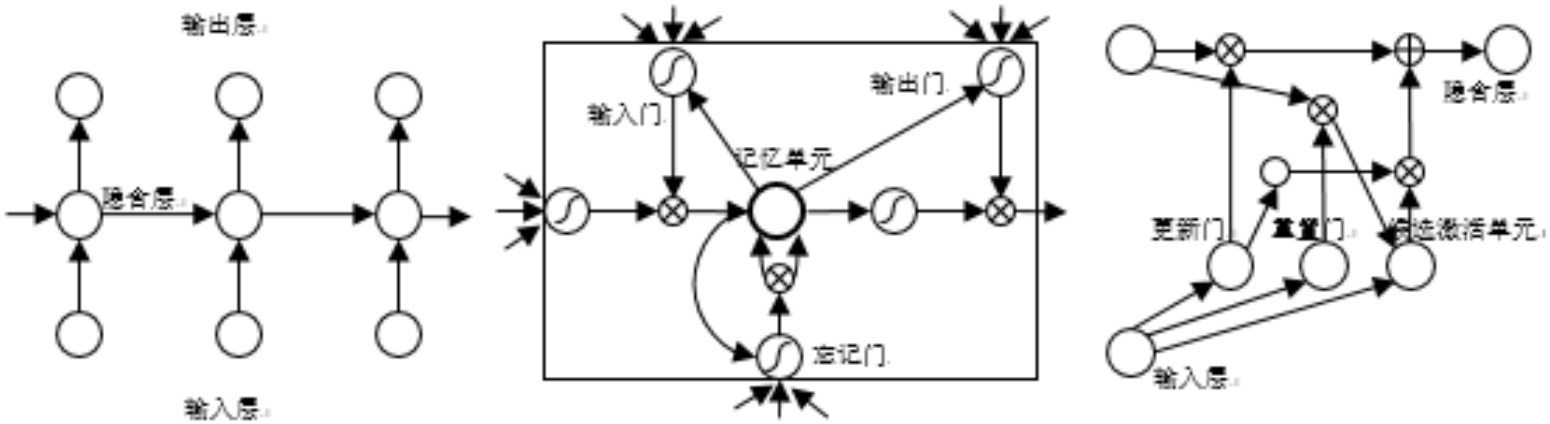
Personalized CNN tag recommendation

DeepCoNN

ConvMF

Convolutional Neural Network based recommender system

Deep Learning based Recommender System



(a) 传统 RNN

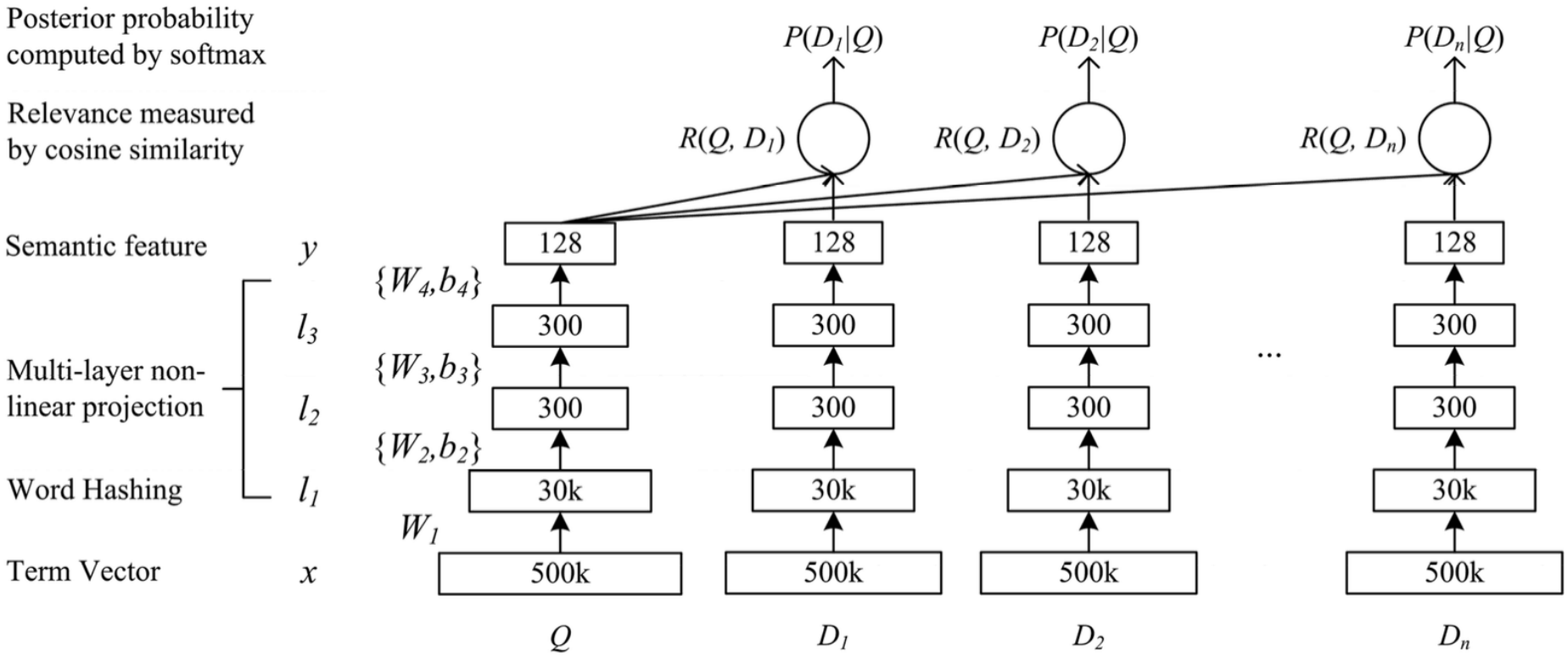
(b) LSTM 网络[Hochreiter97]

(c) GRU 单元[Cho14]

典型的递归神经网络模型

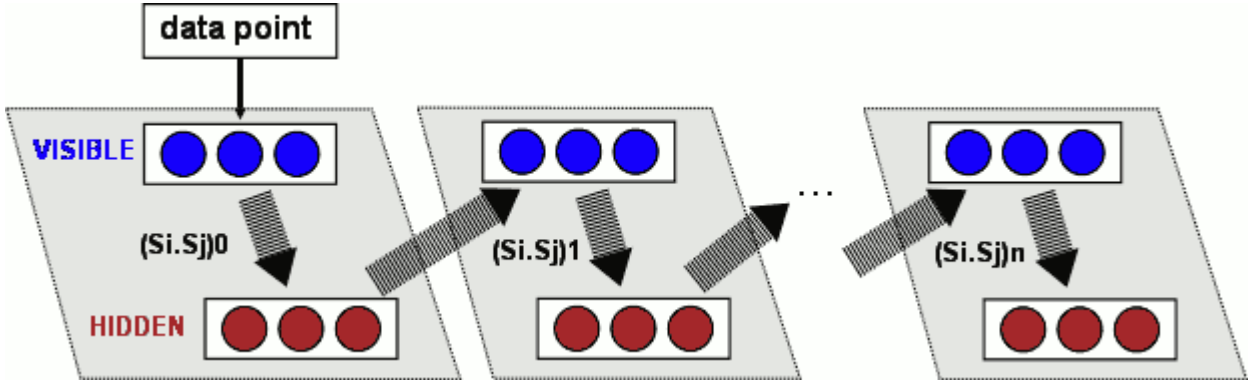
Recurrent Neural Network based recommender system

Deep Learning based Recommender System

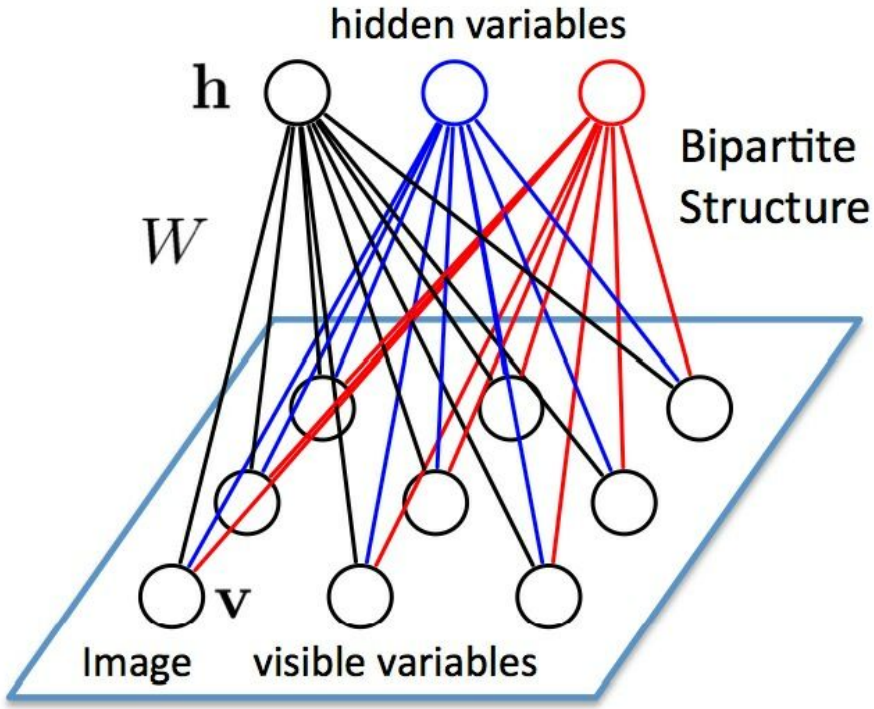


Deep Semantic Similarity Model based recommender system

Deep Learning based Recommender System



$$P(x) = \frac{1}{Z} exp(-E(x))$$

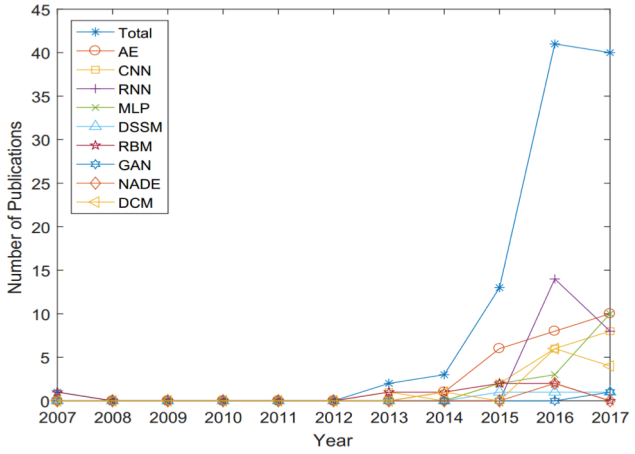


Restricted Boltzmann Machine based recommender system

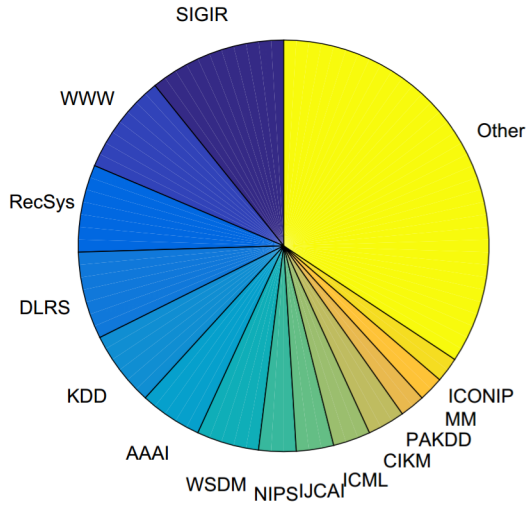
Emerging Method: NADE and GAN

Approach and experiment

Database and Evaluation metrics

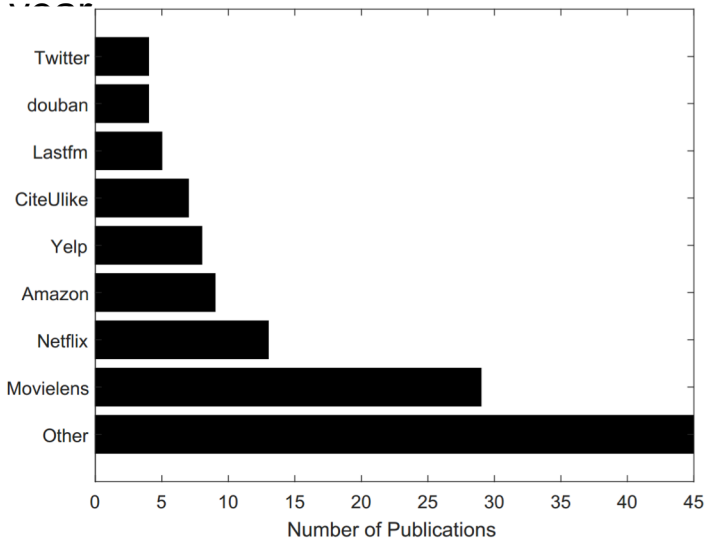


(a) Number of publications in each year

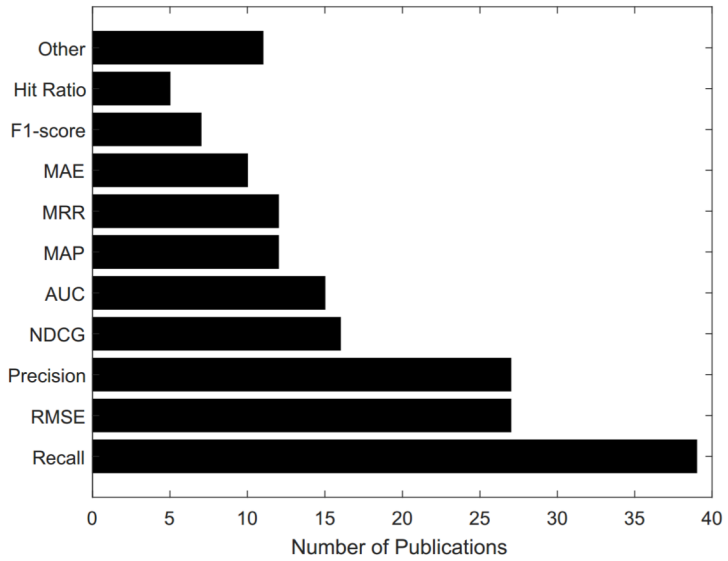


(b) Venue of publications

Most influential works with yearly citation greater than 10

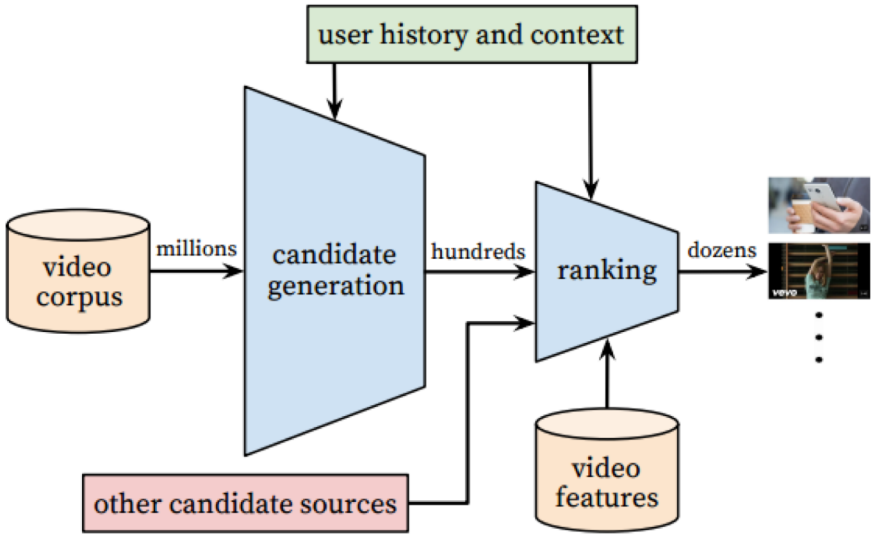


(c) Datasets in use

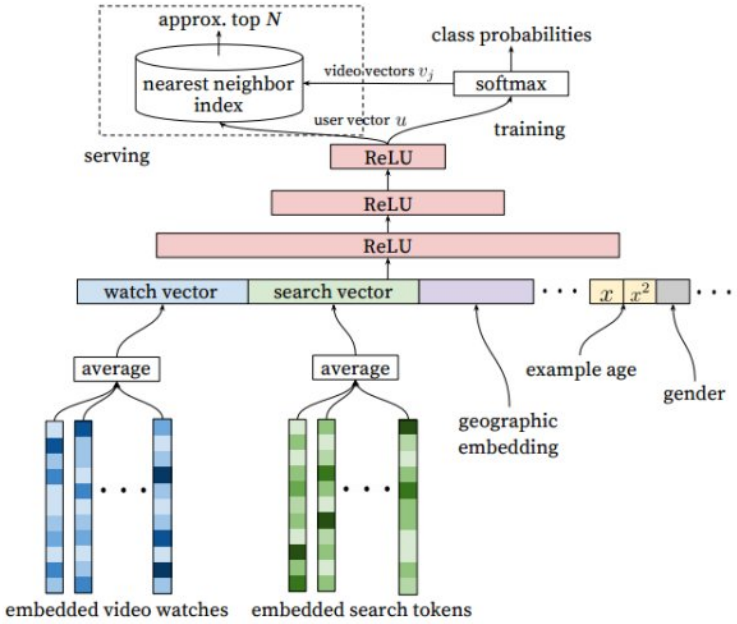


(d) Evaluation metrics in use

Recommendation System Design, Model Design



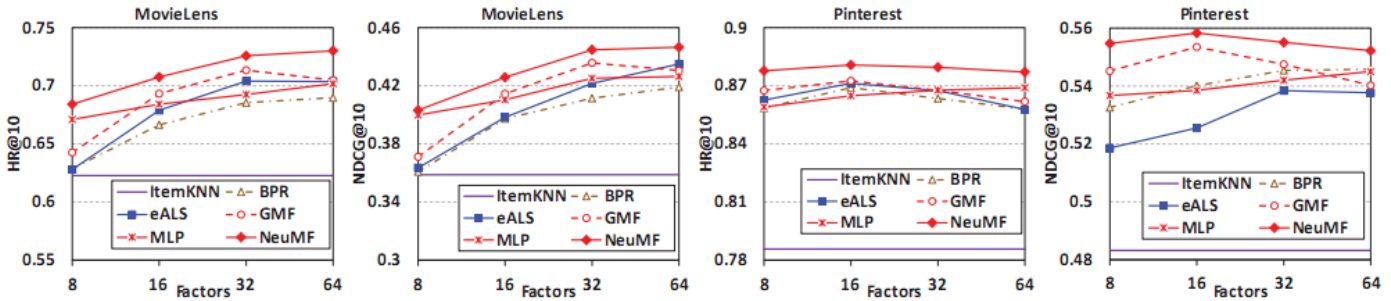
System Design



Model Design

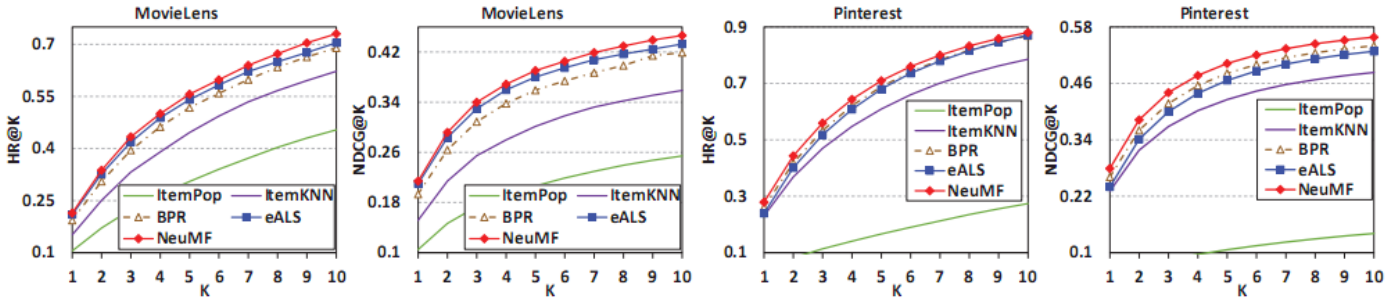
Experiment Validation

	Model	Yelp RMSE (SD)	Yelp MAE (SD)
(a) Mean-centering	User average rating	2.313 (0.008)	1.656 (0.008)
	Item average rating	1.215 (0.003)	0.932 (0.001)
	Overall average rating	1.280 (0.005)	1.030 (0.004)
	HyPER (all mean-centering rules)	**1.199** (0.003)	0.952 (0.002)
(b) User-based	Similar users (Pearson)	2.313 (0.008)	1.656 (0.008)
	Similar users (cosine)	2.313 (0.008)	1.657 (0.008)
	Similar users (latent, cosine)	2.227 (0.007)	1.597 (0.007)
	Similar users (latent, Euclidean)	2.226 (0.009)	1.596 (0.008)
	HyPER (all user-based rules)	**2.194** (0.008)	**1.573** (0.008)
(c) Item-based	Similar items (Pearson)	1.213 (0.004)	0.931 (0.002)
	Similar items (cosine)	1.211 (0.003)	0.928 (0.001)
	Similar items (adjusted cosine)	1.210 (0.004)	0.924 (0.002)
	Similar items (latent, cosine)	1.212 (0.003)	0.923 (0.001)
	Similar items (latent, Euclidean)	1.212 (0.003)	0.931 (0.001)
	HyPER (all item-based rules)	**1.208** (0.004)	0.923 (0.002)
(d) Content &Social	Similar items (content)	1.200 (0.003)	0.939 (0.002)
	Friends	1.199 (0.003)	0.932 (0.002)
	HyPER (content + social rules)	**1.195** (0.003)	**0.927** (0.002)
(e) Base models	Item-based	1.216 (0.004)	0.932 (0.001)
	MF	1.251 (0.006)	0.944 (0.005)
	BPMF	1.191 (0.003)	0.954 (0.003)
	HyPER (baseline rules)	**1.179** (0.003)	**0.926** (0.002)
	HyPER (all rules)	**1.173** (0.003)	**0.917** (0.002)



(a) MovieLens — HR@10 (b) MovieLens — NDCG@10 (c) Pinterest — HR@10 (d) Pinterest — NDCG@10

Figure 4: Performance of HR@10 and NDCG@10 *w.r.t.* the number of predictive factors on the two datasets.



(a) MovieLens — HR@K (b) MovieLens — NDCG@K (c) Pinterest — HR@K (d) Pinterest — NDCG@K

Figure 5: Evaluation of Top-K item recommendation where K ranges from 1 to 10 on the two datasets.

Thanks for listening!