AutoRec: Autoencoders Meet Collaborative Filtering

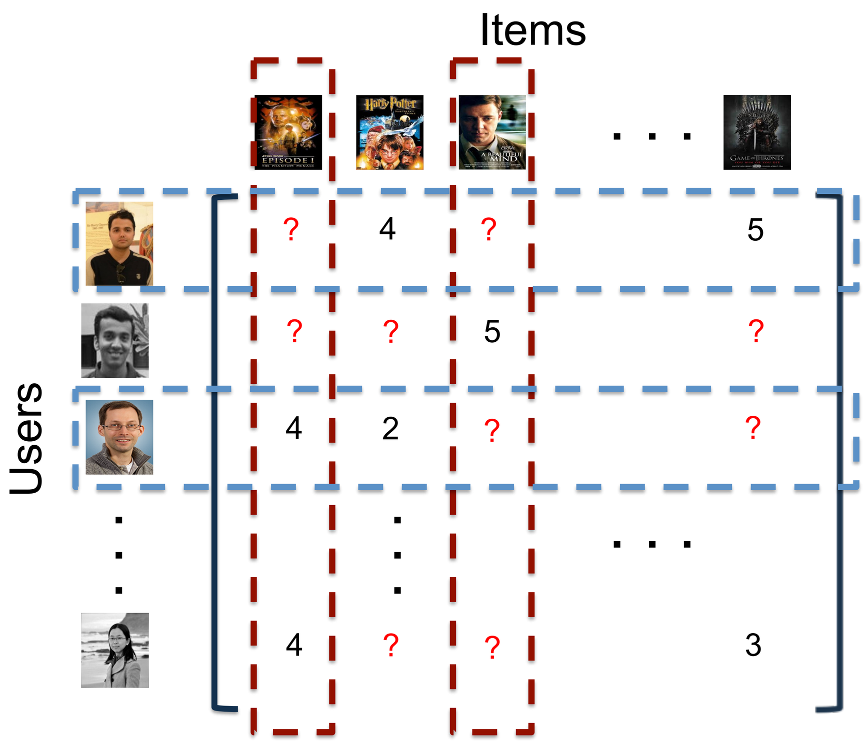
Suvash Sedhain[1,2], Aditya Krishna Menon[2,1], Scott Sanner[2,1], Lexing Xie[1,2]
ANU[1], NICTA[2]

Motivation

Autoencoders have been proven very successful in various vision and speech problems. Can we do the same for collaborative filtering?

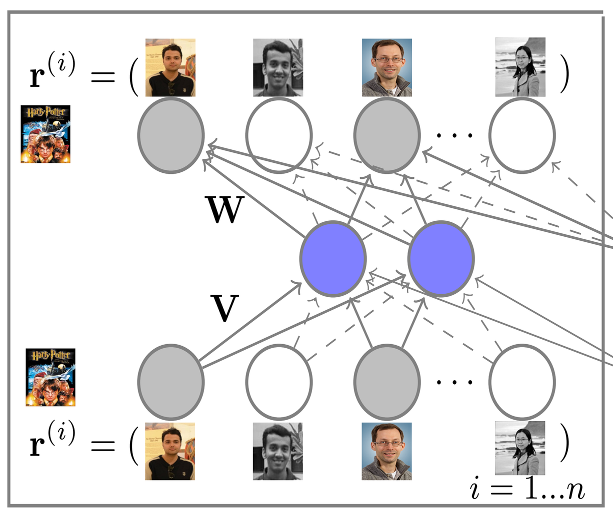
A: Yes

Rating prediction problem



Given a partially observed user-item rating matrix, $R^{m \times n}$, fill in the missing entries

AutoRec model



For each item, construct (partially observed) vector of ratings $\mathbf{r}^{(i)}$

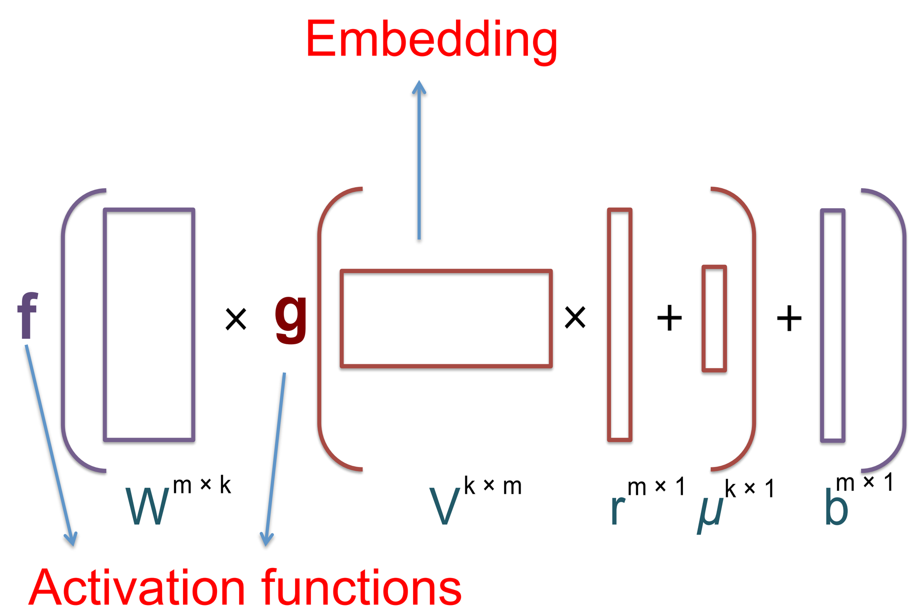
Perform autoencoding on result, where

- Weights are tied across items
- Only observed ratings are used to update model

Prediction:

$$\hat{R}_{ui} = (h(\mathbf{r}^{(i)}; \hat{\theta}))_u.$$

$$h(\mathbf{r}; \theta) = f\left(\mathbf{W} \cdot g(\mathbf{V}\mathbf{r} + \boldsymbol{\mu}) + \mathbf{b}\right)$$



Training objective:

$$\min_{\theta} \sum_{i=1}^{n} ||\mathbf{r}^{(i)} - h(\mathbf{r}^{(i)}; \theta))||^2_{\mathcal{O}} + \frac{\lambda}{2} \cdot (||\mathbf{W}||^2_F + ||\mathbf{V}||^2_F),$$

Comparisons with existing methods

	AutoRec	RBM-CF
Model type	Discriminative	Generative
Objective	RMSE	Log-likelihood
Optimisation	Gradient-based (fast)	Contrastive divergence (slow)
Ratings	Real-valued	Discrete

	AutoRec	Matrix Factorization
Embedding	Users only	Users and items (more parameters)
Representation	Non-linear	Linear

Experiments

Data Description

	#users	#items
Ml-1M	6,040	3,706
Ml-10M	69,878	10,677
Netflix	480,189	17,770

Q: Is user- or item-based modelling better?

A: Item-based is superior.

	ML-1M	ML-10M
U-RBM	0.881	0.823
I-RBM	0.854	0.825
U-AutoRec	0.874	0.867
I-AutoRec	**0.831**	**0.782**

Q: What are good choices of activations *f*(.), *g*(.)?

A: Nonlinearity in hidden unit is key for performance

$f(\cdot)$	$g(\cdot)$	RMSE
Identity	Identity	0.872
Sigmoid	Identity	0.852
Identity	Sigmoid	**0.831**
Sigmoid	Sigmoid	0.836

Q: How many hidden units are needed for AutoRec?

A: Good performance with ~400 hidden units

Q: How does AutoRec perform against all baselines?

A: Systematically outperforms state-of-the-art methods

	ML-1M	ML-10M	Netflix
BiasedMF	0.845	0.803	0.844
I-RBM	0.854	0.825	-
U-RBM	0.881	0.823	0.845
LLORMA	0.833	**0.782**	0.834
I-AutoRec	**0.831**	**0.782**	**0.823**

Q: Do deep extensions of AutoRec help?

A: Deep I-AutoRec with three hidden layers (500-200-500) reduced RMSE from 0.831 to 0.827 on ML-1M dataset.

Try it now: https://github.com/mesuvash/NNRec

Future work

Further exploration of deep autoencoders, and applications to implicit feedback datasets.

References

[1] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. Computer, 42, 2009.
[2] J. Lee, S. Kim, G. Lebanon, and Y. Singer. Local low-rank matrix approximation. In ICML, 2013.
[3] M. Riedmiller and H. Braun. A direct adaptive method for faster backpropagation learning: the rprop algorithm. In IEEE International Conference on Neural Networks, 1993.
[4] R. Salakhutdinov, A. Mnih, and G. Hinton. Restricted Boltzmann machines for collaborative filtering. In ICML, 2007.
[5] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In WWW, 2001.