SVD-AE: Simple Autoencoders for Collaborative Filtering

Seoyoung Hong[1], Jeongwhan Choi[2], Yeon-Chang Lee[3], Srijan Kumar[4], Noseong Park[5]

[1]Boeing Korea Engineering & Technology Center [2]Yonsei University
[3]UNIST [4]Georgia Institute of Technology [5]KAIST [2]jeongwhan.choi@yonsei.ac.kr

Motivation: Achieving Optimal Balance in Recommendation Systems

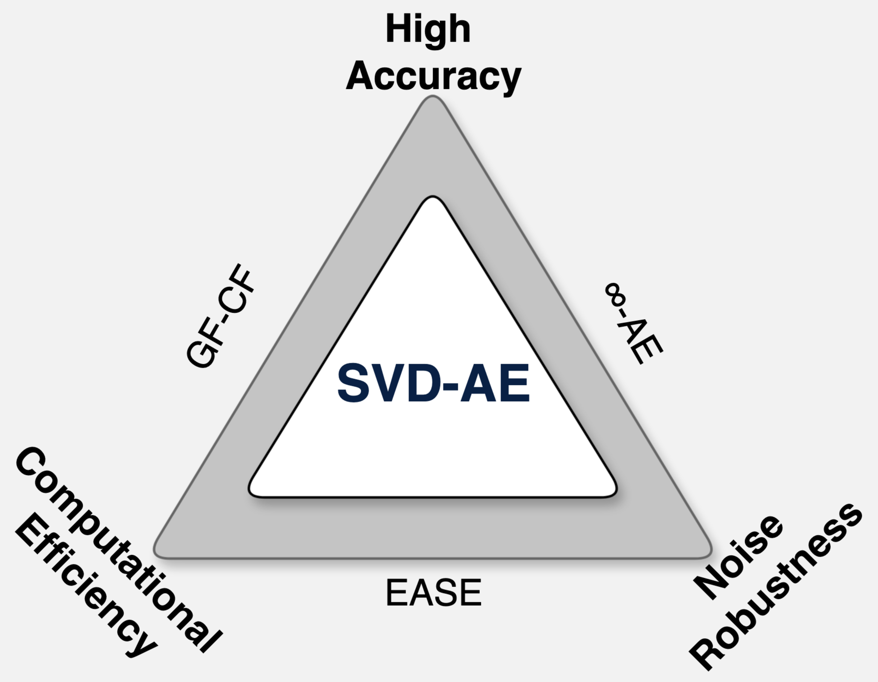


Figure 1:The best overall balance between 3 goals.

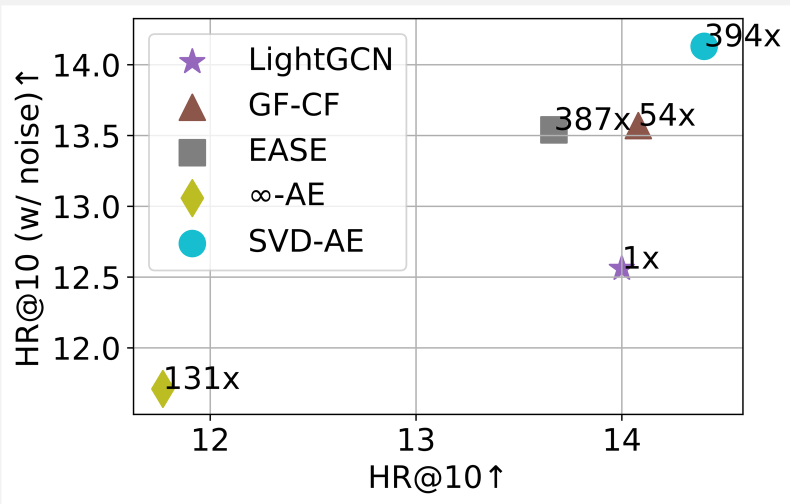


Figure 2:The accuracy, robustness, and computation time of various methods on Gowalla.

	GF-CF [1]	EASE [2]	∞-AE [3]	**SVD-AE**
Closed-form Solution	✓	✓	✓	✓
Autoencoder-based	✗	✓	✓	✓
Using SVD	✓	✗	✗	✓
Using Neural Networks	✗	✗	✓	✗

Table 1:Comparison of existing lightweight methods and our SVD-AE.

Generalized Linear Autoencoder for Recommender Systems

- The objective function of linear autoencoder is:

$$\min_{\hat{\mathbf{R}}} \quad \|\mathbf{R} - \hat{\mathbf{R}}\|_2^2, \quad \text{s.t.} \quad \mathcal{C}, \tag{1}$$

  - $\mathbf{R} \in \{0,1\}^{|U|\times|I|}$ is the given user-item interaction matrix
  - $\hat{\mathbf{R}} \in \{0,1\}^{|U|\times|I|}$ is the reconstructed interaction matrix
- EASE uses ridge regression with a regularization term:

$$\min_{\mathbf{B}} \quad \|\mathbf{R} - \mathbf{R}\mathbf{B}\|_F^2 + \lambda \cdot \|\mathbf{B}\|_F^2, \quad \text{s.t.diag}(\mathbf{B}) = 0, \tag{2}$$

- ∞-AE uses Kernelized Ridge Regression:

$$\underset{[\alpha_j]_{j=1}^{|U|}}{\text{argmin}} \quad \sum_{u\in U} \|\mathbf{R}_u - f(\mathbf{R}_u|\alpha)\|_2^2 + \lambda \cdot \|f\|_{\mathcal{H}}^2. \tag{3}$$

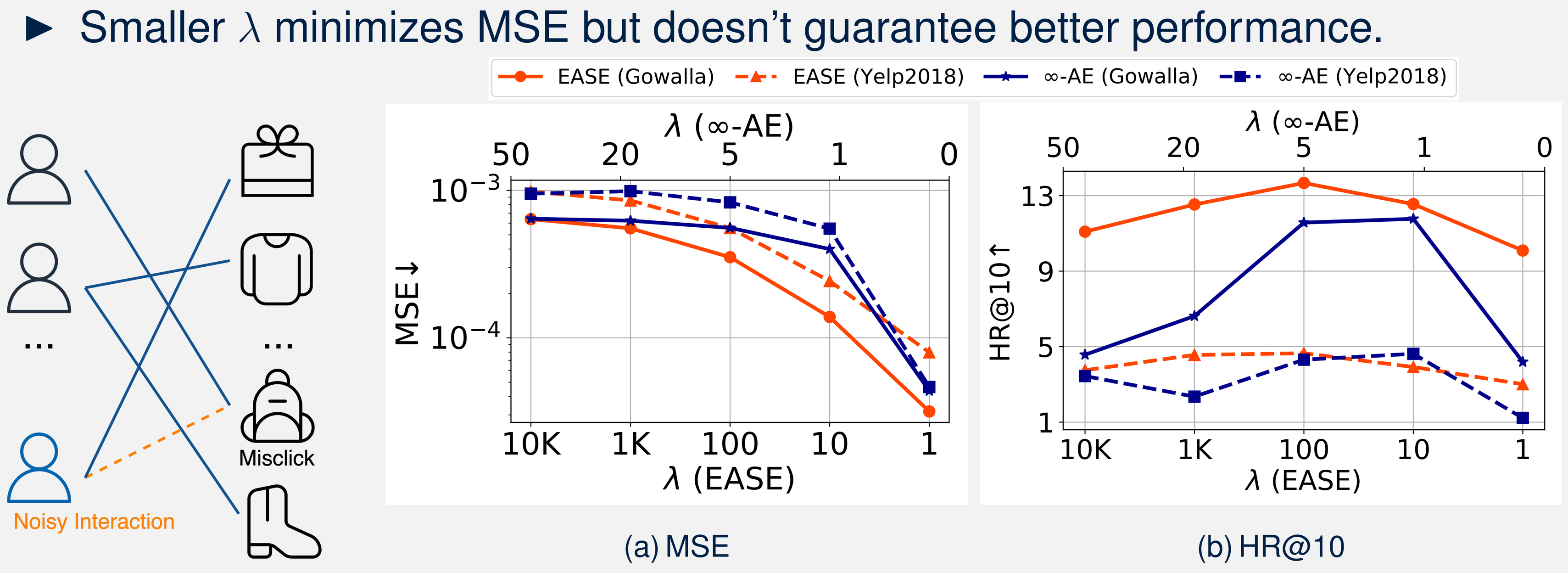
- Closed-form solutions for optimal $\hat{\mathbf{R}}$ in different methods:

$$\hat{\mathbf{R}} = \begin{cases} \mathbf{R} \cdot (\mathbf{I} - \hat{\mathbf{P}} \cdot \text{diagMat}(\vec{1} \oslash \text{diag}(\hat{\mathbf{P}}))) & \text{(EASE)}, \\ \mathbf{K} \cdot (\mathbf{K} + \lambda\mathbf{I})^{-1} \cdot \mathbf{R} & (\infty\text{-AE}), \\ \tilde{\mathbf{R}} \cdot \mathbf{V}\tilde{\Sigma}^{+}\mathbf{Q}^T\mathbf{R} & \text{(SVD-AE)}, \end{cases} \tag{4}$$

  - $\hat{\mathbf{P}} = (\mathbf{R}^T\mathbf{R} + \lambda\mathbf{I})^{-1}$.
  - $\tilde{\mathbf{R}} = \mathbf{D}_U^{-\frac{1}{2}}\mathbf{R}\mathbf{D}_I^{-\frac{1}{2}}$ is a normalized adjacency matrix.

The Presence of Noise

- $\mathbf{R}$ often contains noisy interactions that don't reflect true user preferences.
- EASE and ∞-AE use λ to prevent overfitting to noisy rating matrix.
- Smaller λ minimizes MSE but doesn't guarantee better performance.



(a) MSE (b) HR@10

Figure 3:The performance comparison with different regularization parameters.

SVD-AE Method

- **SVD-AE** solves a ridge regression-like problem:

$$\min_{\mathbf{B}} \|\mathbf{R} - \tilde{\mathbf{R}}\mathbf{B}\|^2 \tag{5}$$

  - The regularization term is implicitly handled by truncated SVD.
- Novel closed-form solution:

$$\mathbf{B} = \tilde{\mathbf{R}}^{+}\mathbf{R} = \mathbf{V}\Sigma^{+}\mathbf{Q}^T\mathbf{R}, \tag{6}$$

  - Let $\tilde{\mathbf{R}} = \mathbf{Q}\Sigma\mathbf{V}^T$ be the SVD of $\tilde{\mathbf{R}}$, then we can get the pseudo-inverse of $\tilde{\mathbf{R}}$, $\tilde{\mathbf{R}}^{+}$.
  - $\mathbf{Q} \in \mathbb{R}^{|U|\times m}$ and $\mathbf{V} \in \mathbb{R}^{|I|\times m}$ are top-m singular vectors
  - $\tilde{\Sigma}^{+}$ contains inverse of top-m singular values of $\tilde{\mathbf{R}}$
  - $m = \lfloor \gamma \times \min(|U|, |I|) \rfloor$, where $\gamma = 0.04$ is optimal for all datasets
- Low-rank Inductive Bias in **SVD-AE**:
  - Reduces noise (smaller singular values).
  - Speeds up calculations for large, sparse matrices.

Performance Comparison

Dataset	Measure	LightGCN	GF-CF	MultVAE	EASE	∞-AE	SVD-AE
Gowalla	HR@10	14.00	14.08	11.88	13.67	11.77	**14.40**
	HR@100	37.40	**38.84**	33.56	35.74	34.20	37.34
	NDCG@10	13.77	13.50	11.30	13.15	10.84	**13.94**
	NDCG@100	21.04	**21.25**	18.11	20.08	17.97	21.15
	PSP@10	2.26	2.47	2.09	2.31	2.02	**2.48**
Yelp2018	HR@10	4.32	4.87	4.31	4.65	4.62	**4.90**
	HR@100	19.01	**20.86**	18.75	17.74	18.33	19.79
	NDCG@10	4.19	4.66	4.10	4.55	4.48	**4.74**
	NDCG@100	9.57	**10.53**	9.37	9.37	9.54	10.22
	PSP@10	0.39	0.44	0.43	0.42	0.43	**0.45**
ML-1M	HR@10	29.07	30.81	27.86	30.43	31.15	**31.79**
	HR@100	57.62	59.10	57.67	57.74	**60.75**	59.33
	NDCG@10	30.30	32.37	28.44	31.90	32.27	**33.55**
	NDCG@100	39.95	42.00	39.34	40.95	42.54	**42.57**
	PSP@10	3.01	3.17	3.13	3.16	**3.22**	**3.22**
ML-10M	HR@10	34.79	35.10	34.20	36.30	35.83	**36.76**
	HR@100	64.11	64.23	64.55	64.78	64.48	**64.80**
	NDCG@10	35.60	36.02	34.48	37.63	36.93	**37.75**
	NDCG@100	46.14	45.71	45.23	46.74	46.27	**46.97**
	PSP@10	4.69	4.73	4.82	4.76	4.74	**4.93**

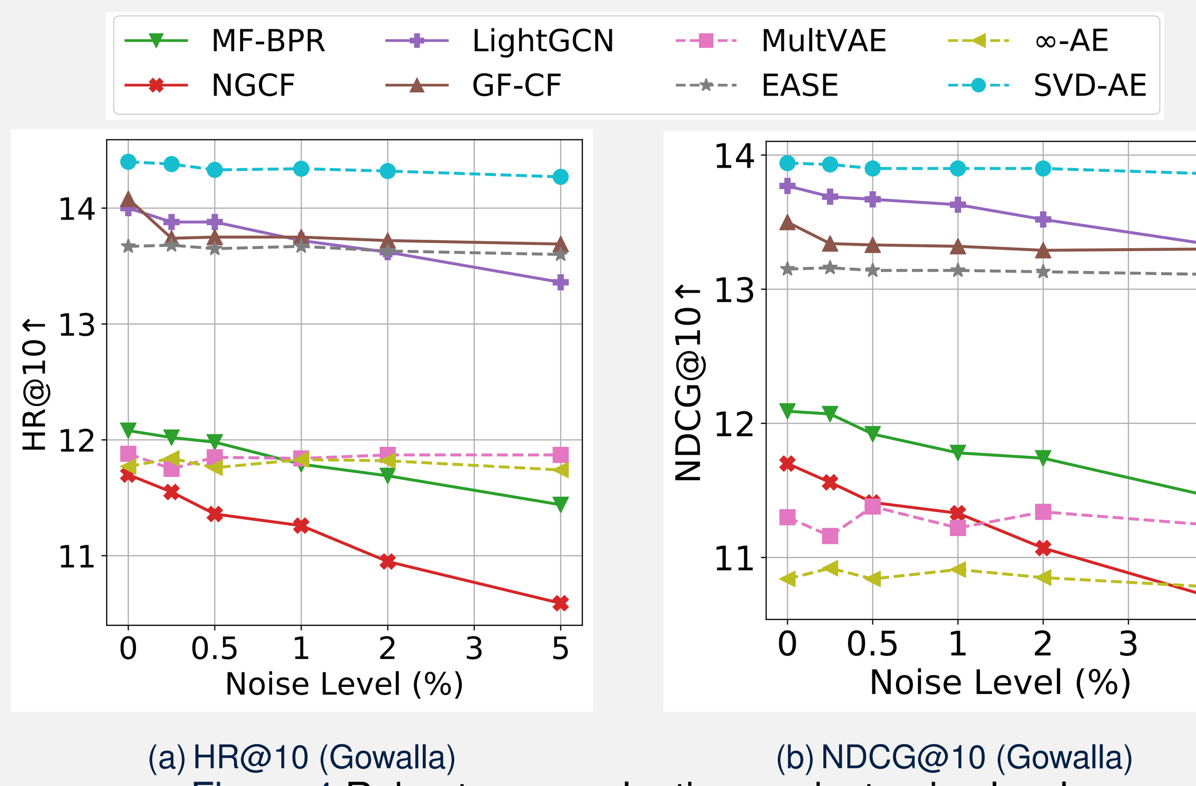
Table 2:Performance evaluation of overall performance among SVD-AE and baselines

Efficiency Comparison

Model	ML-1M		ML-10M	
	Pre-processing	Training	Pre-processing	Training
LightGCN	N/A	2.44h	N/A	132.97h
GF-CF	4.62s	6.37s	28.98s	1260.80s
EASE	4.52s	5.72s	52.63s	6.05s
∞-AE	N/A	2.24s	N/A	388.39s
SVD-AE	0.54s	2.06s	47.59s	3.06s

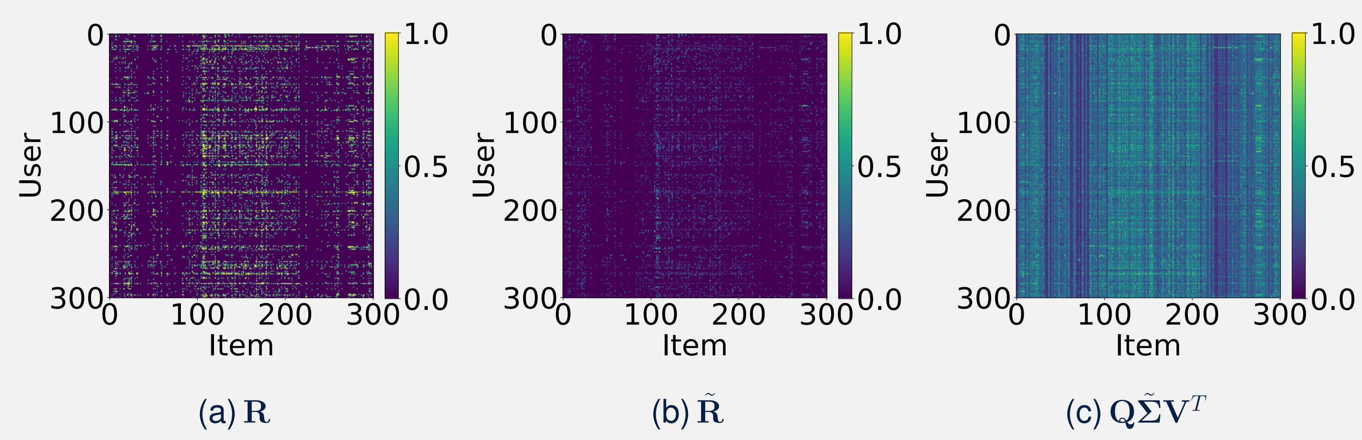
Table 3:Efficiency comparison on overall computation time.

Robustness on Noise



(a) HR@10 (Gowalla) (b) NDCG@10 (Gowalla)

Figure 4:Robustness evaluation against noise level.



(a) $\mathbf{R}$ (b) $\tilde{\mathbf{R}}$ (c) $\mathbf{Q}\Sigma\mathbf{V}^T$

Figure 5:The smoothing effect of the truncated SVD in reducing noise.

References

[1] Yifei Shen, Yongji Wu, Yao Zhang, Caihua Shan, Jun Zhang, B Khaled Letaief, and Dongsheng Li. How powerful is graph convolution for recommendation? In *CIKM*, 2021.

[2] Harald Steck. Embarrassingly shallow autoencoders for sparse data. In *TheWebConf*, 2019.

[3] Noveen Sachdeva, Mehak Preet Dhaliwal, Carole-Jean Wu, and Julian McAuley. Infinite recommendation networks: A data-centric approach. In *NeurIPS*, 2022.

Paper GitHub

jeongwhanchoi.me
jeongwhan_choi
jeongwhanchoi
jeongwhanchoi

IJCAI JEJU 2024