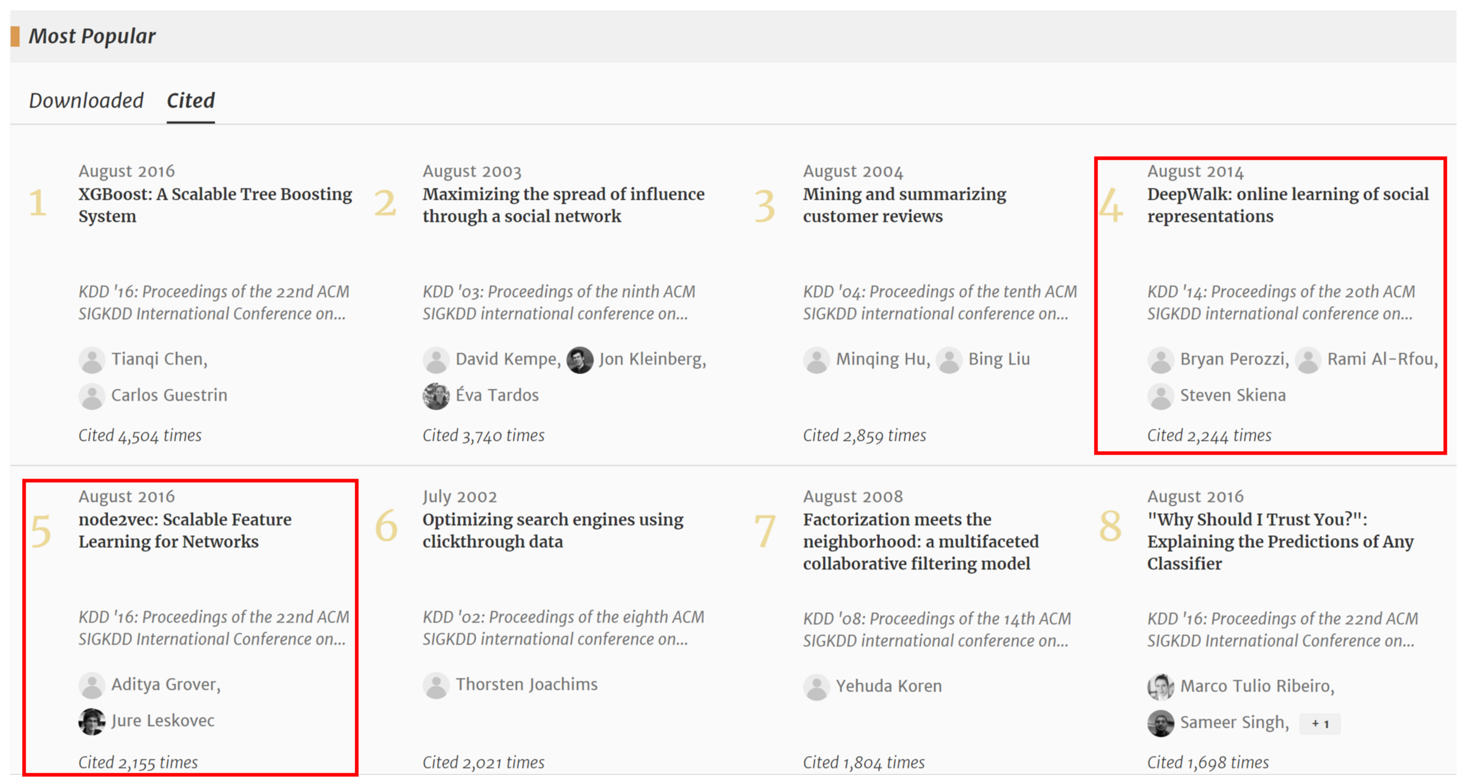
A Broader Picture of Random-walk Based Graph Embedding

Zexi Huang[1], Arlei Silva[2], Ambuj Singh[1]

[1]UC Santa Barbara, [2]Rice University

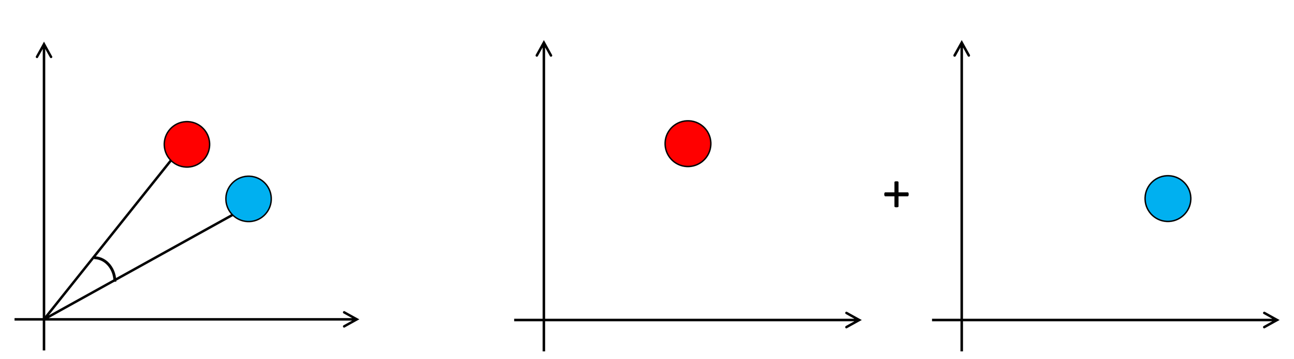
Popularity of Random-walk Based Embedding



- (Perrozi et al., 2014) DeepWalk: online learning of social representations
- (Grover and Leskovec, 2016) node2vec: scalable feature learning for networks
- ... (with **many more** items omitted)
- (Chanpuriya and Musco, 2020) InfiniteWalk: deep network embeddings as ...
- (Zhu et al., 2021) Node proximity is all you need: ...

How can one compare existing methods and to design novel ones?

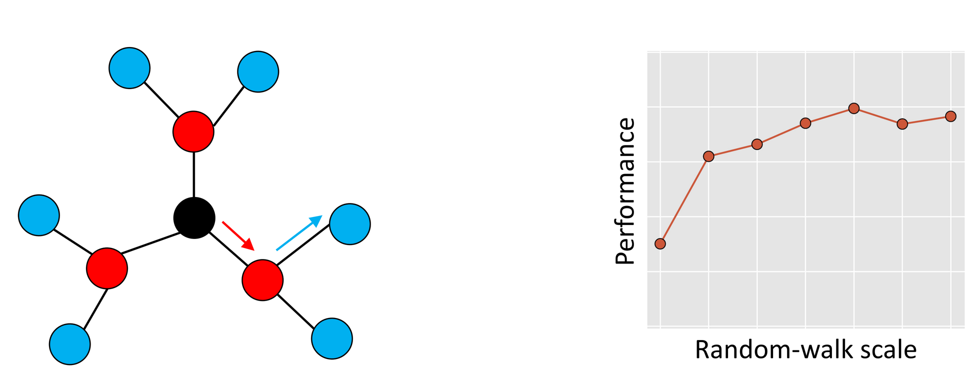
How to Use Embeddings for Link Prediction



(a) Dot product [1] (b) Classification based on combined embeddings [2]

How should embeddings be used for link prediction?

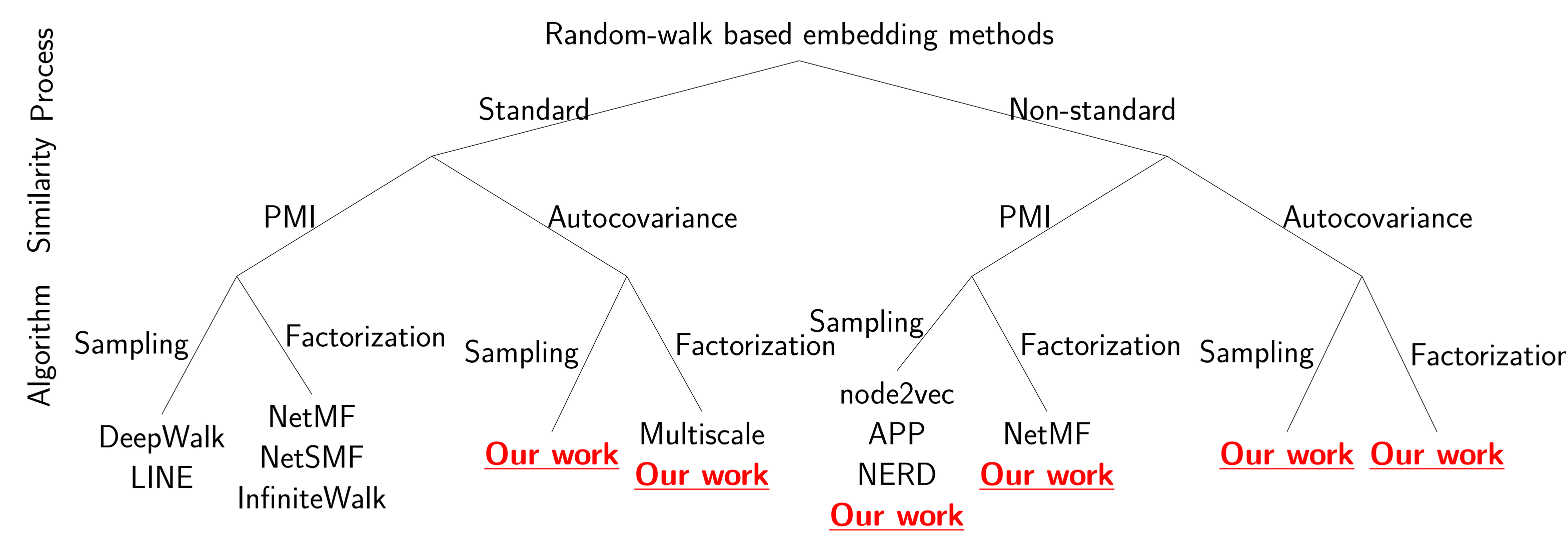
How to Embed Graphs with Multiple Scales



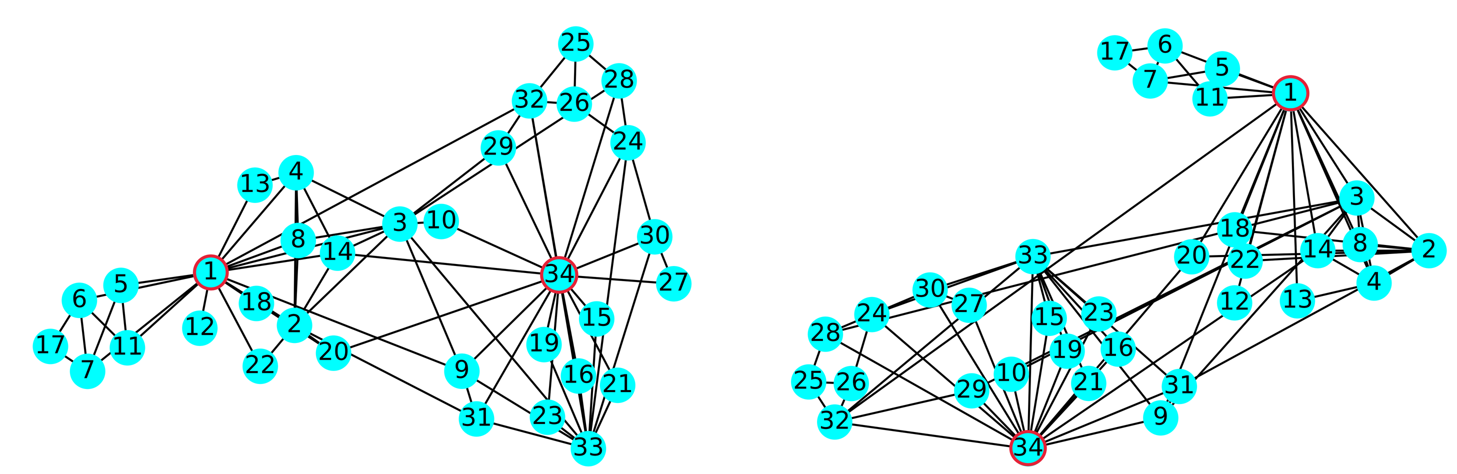
(a) Random-walks capture multiple structural scales [3] (b) Embedding performance [2] insensitve to random-walk scales

How do embeddings capture different structural scales?

Our Analytical Framework



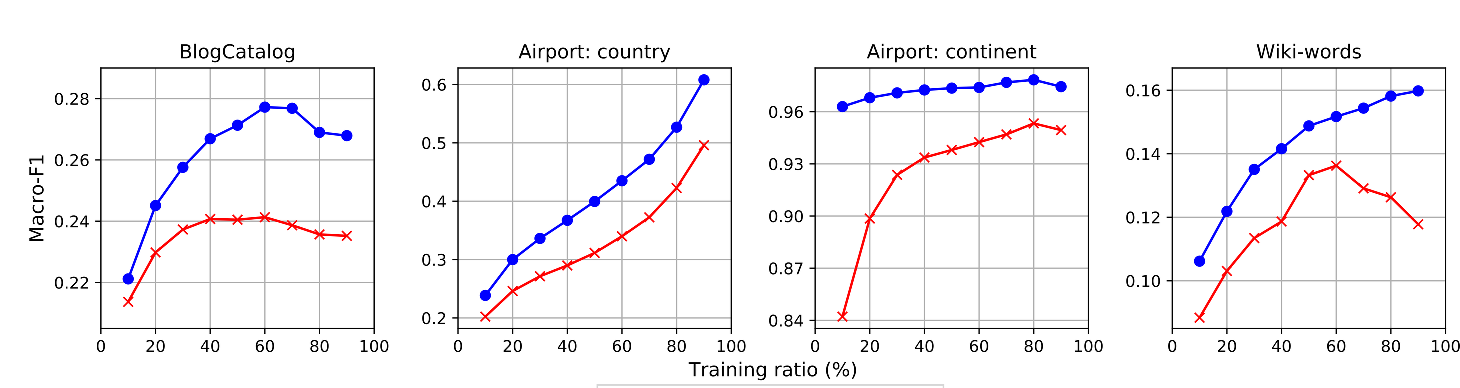
Random-walk process:
Standard random-walk: $M = D^{-1}A$; $\Pi = D/2m$
Non-standard random-walk: BFS/DFS, PageRank, Ruelle-Bowen
Similarity metric:
Pointwise Mutual Information [4]: $R = \log(\Pi M^\tau) - \log(\pi\pi^T)$
Autocovariance [3]: $R = \Pi M^\tau - \pi\pi^T$



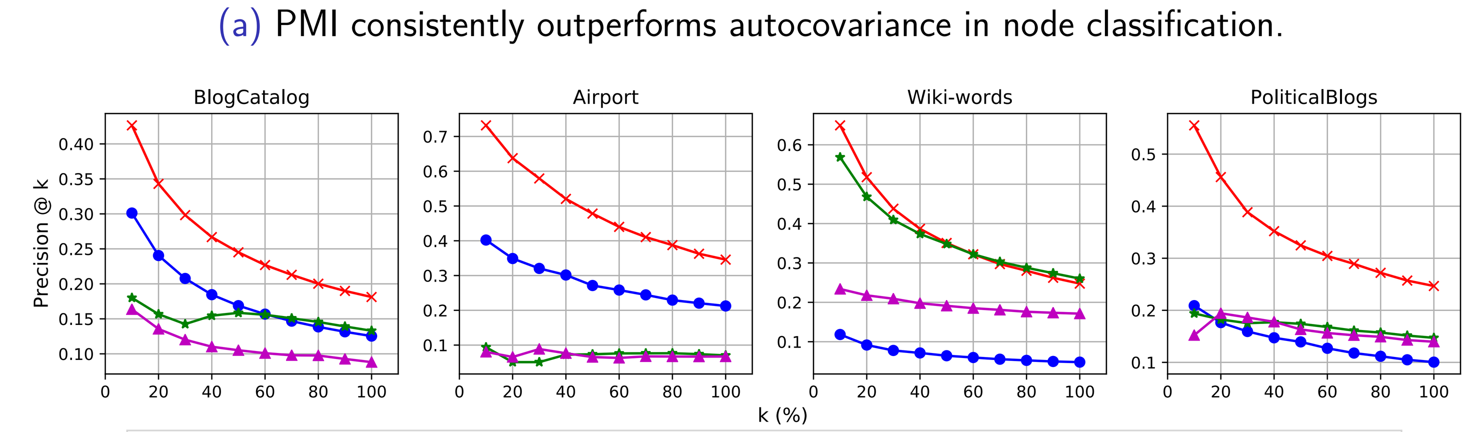
(a) PMI (b) Autocovariance

Embedding algorithm:
Factorization (SVD): $\min \|UU^T - R\|_F^2$
Sampling (SGD): $\max \sum_{u,v} \log \Pr((u,v) \in \mathcal{D} | \mathbf{u}_u, \mathbf{v}_v; R)$

PMI vs Autocovariance



(a) PMI consistently outperforms autocovariance in node classification.



(b) Autocovariance with dot product ranking consistently outperforms PMI (with either ranking scheme) in link prediction, more than doubling the precision for PoliticalBlogs.

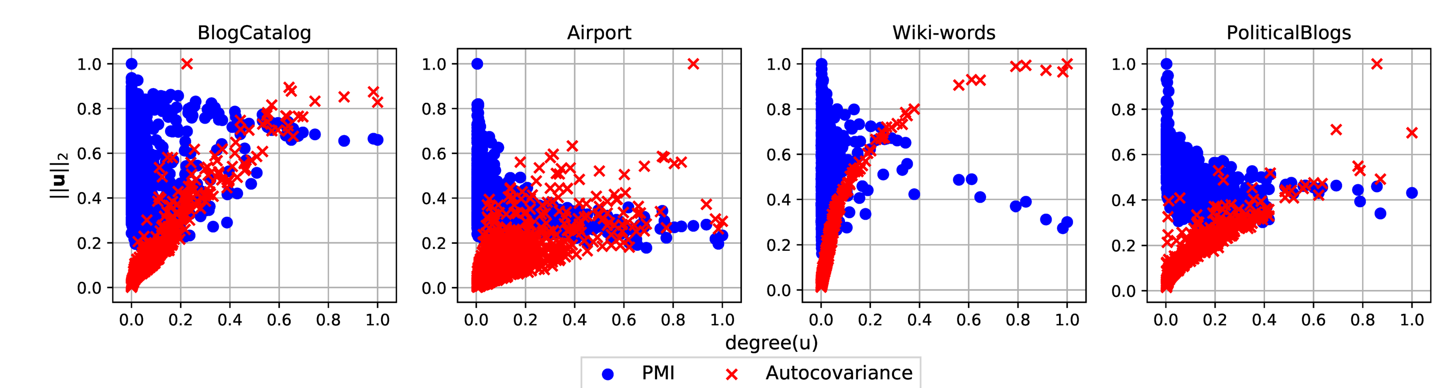
Why Autocovariance Shines in Link Prediction

Observation 1. *Link prediction based on dot products correlates predicted node degrees and the 2-norms of embedding vectors:*

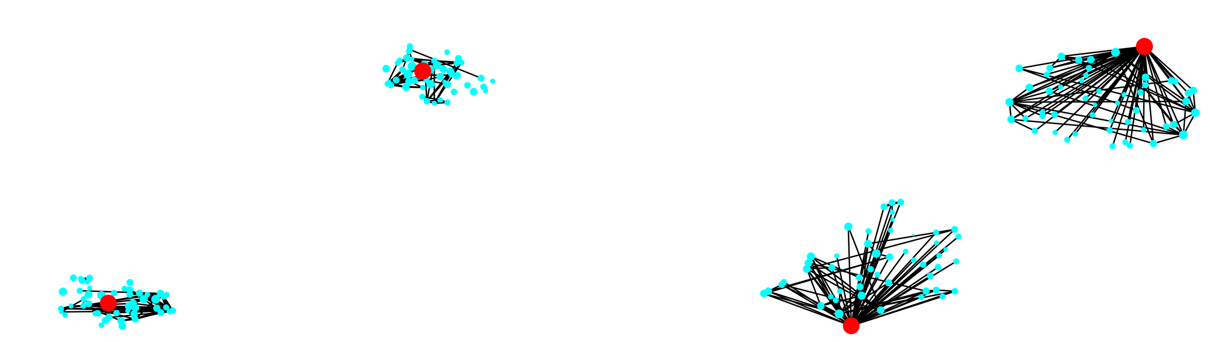
$$\widetilde{\deg}(u) \approx \sum_{v \in C(u)-\{u\}} \mathbf{u}^T\mathbf{v} = \|\mathbf{u}\|_2 \sum_{v \in C(u)-\{u\}} \|\mathbf{v}\|_2 \cos(\theta_{\mathbf{u},\mathbf{v}}) \propto \|\mathbf{u}\|_2$$

Observation 2. *Norms of autocovariance embedding vectors are correlated with actual node degrees, while norms of PMI embedding vectors are not:*

$$\|\mathbf{u}\|_2^2 = \pi(u)([M^\tau]_{u,u} - \pi(u)) = \frac{\deg(u)}{2m}\left([M^\tau]_{u,u} - \frac{\deg(u)}{2m}\right) \propto \deg^2(u)$$

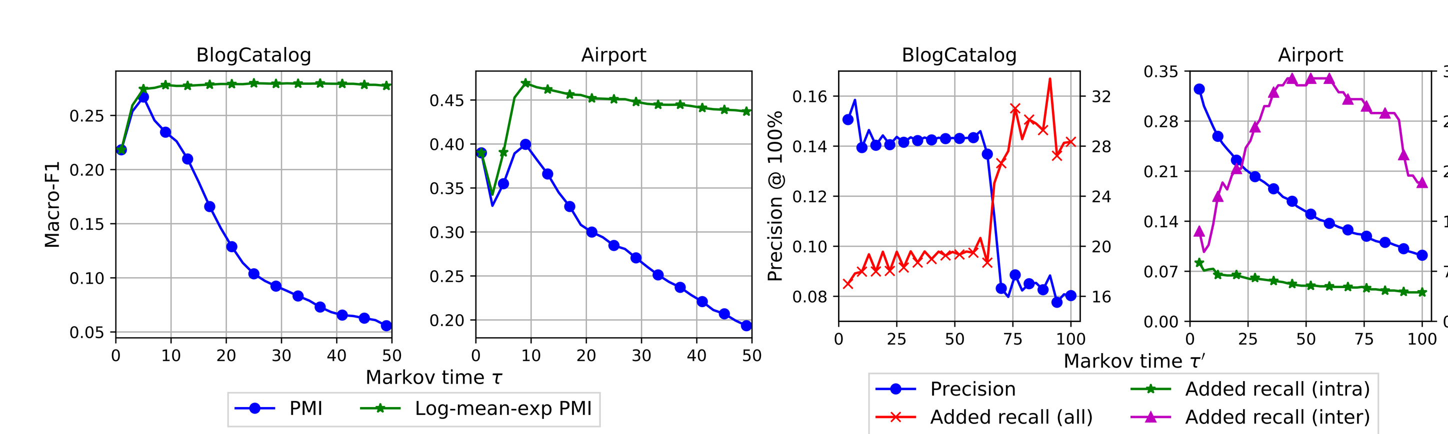


Observation 3. *Autocovariance enables state-of-the-art link prediction by capturing heterogeneous degree distribution in real graphs:* $\widetilde{\deg}(u) \propto \deg(u)$.



(a) PMI (b) Autocovariance

Embedding at Different Markov Times



(a) Node classification performance for PMI can be improved by smooth-averaging with *log-mean-exp* across multiple Markov times.
$\tilde{R} = \log(\Pi \frac{1}{\tau}\sum_{t=1}^{\tau} M^t) - \log(\pi\pi^T)$

(b) Prediction of edges of specific structural scales (e.g., inter-community) can be improved with different Markov times for autocovariance.
$R = \Pi M^\tau - \pi\pi^T$

References

[1] Mingdong Ou, Peng Cui, Jian Pei, Ziwei Zhang, and Wenwu Zhu. Asymmetric transitivity preserving graph embedding. In *SIGKDD*, 2016.

[2] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *SIGKDD*, 2016.

[3] J-C Delvenne, Sophia N Yaliraki, and Mauricio Barahona. Stability of graph communities across time scales. *PNAS*, 107(29):12755–12760, 2010.

[4] Omer Levy and Yoav Goldberg. Neural word embedding as implicit matrix factorization. In *NeurIPS*, 2014.

This work is funded by NSF via grant IIS 1817046 and DTRA via grant HDTRA1-19-1-0017.