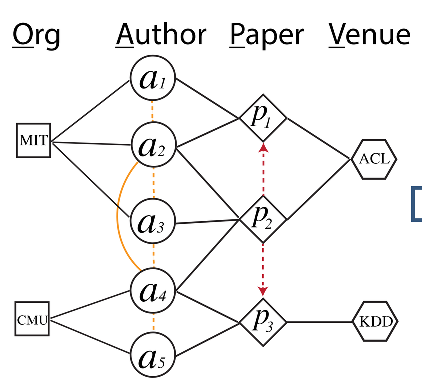
metapath2vec: Scalable Representation Learning for Heterogeneous Networks

Yuxiao Dong (yuxdong@microsoft.com) University of Notre Dame & Microsoft Research
Nitesh V. Chawla (nchawla@nd.edu) University of Notre Dame
Ananthram Swami (ananthram.swami.civ@mail.mil) Army Research Laboratory

Heterogeneous Network Embedding

- **Input:** A heterogeneous network *G = (V, E, T)* in which $T = \{T_V, T_E\}$ denotes the node and edge types.
- **Output**: *d*-dimensional latent representations $\boldsymbol{X} \in \mathbb{R}^{|V| \times d}, d \ll |V|$
- **Goal:** $\boldsymbol{X}$ is able to capture the structural and semantic relations among different types of nodes.



metapath2vec metapath2vec++ ⇒ X embedding ⇒

Network Mining Tasks
- node label inference
- community detection
- similarity search
- link prediction
- …

- **Challenges:**
  - How do we effectively preserve the concept of "node-context" among multiple types of nodes, e.g., authors, papers, & venues in academic heterogeneous networks?
  - Can we directly apply homogeneous network embedding architectures to heterogeneous networks?
  - It is difficult for conventional meta-path based methods to model similarities between nodes without connected meta-paths.
- **Solutions:**

metapath2vec / metapath2vec++ ⇒ meta-path-based random walks + skip-gram / heterogeneous skip-gram

metapath2vec++

Input: The heterogeneous information network $G = (V, E, T)$, a meta-path scheme $\mathcal{P}$, #walks per node w, walk length l, embedding dimension d, neighborhood size k
Output: The latent node embeddings $\mathbf{X} \in \mathbb{R}^{|V| \times d}$

initialize $\mathbf{X}$;
for $i = 1 \rightarrow w$ **do**
  for $v \in V$ **do**
    MP = MetaPathRandomWalk($G, \mathcal{P}, v, l$) ;
    $\mathbf{X}$ = HeterogeneousSkipGram($\mathbf{X}, k, MP$) ;
  end
end
return $\mathbf{X}$;

MetaPathRandomWalk($G, \mathcal{P}, v, l$)
$MP[1] = v$;
for $i = 1 \rightarrow l{-}1$ **do**
  draw u according to Eq. 3 ;
  $MP[i+1] = u$;
end
return MP ;

HeterogeneousSkipGram($\mathbf{X}, k, MP$)
for $i = 1 \rightarrow l$ **do**
  $v = MP[i]$;
  for $j = max(0, i{-}k) \rightarrow min(i{+}k, l)$ & $j \neq i$ **do**
    $c_t = MP[j]$;
    $X^{new} = X^{old} - \eta \cdot \frac{\partial O(\mathbf{X})}{\partial X}$ (Eq. 7) ;
  end
end

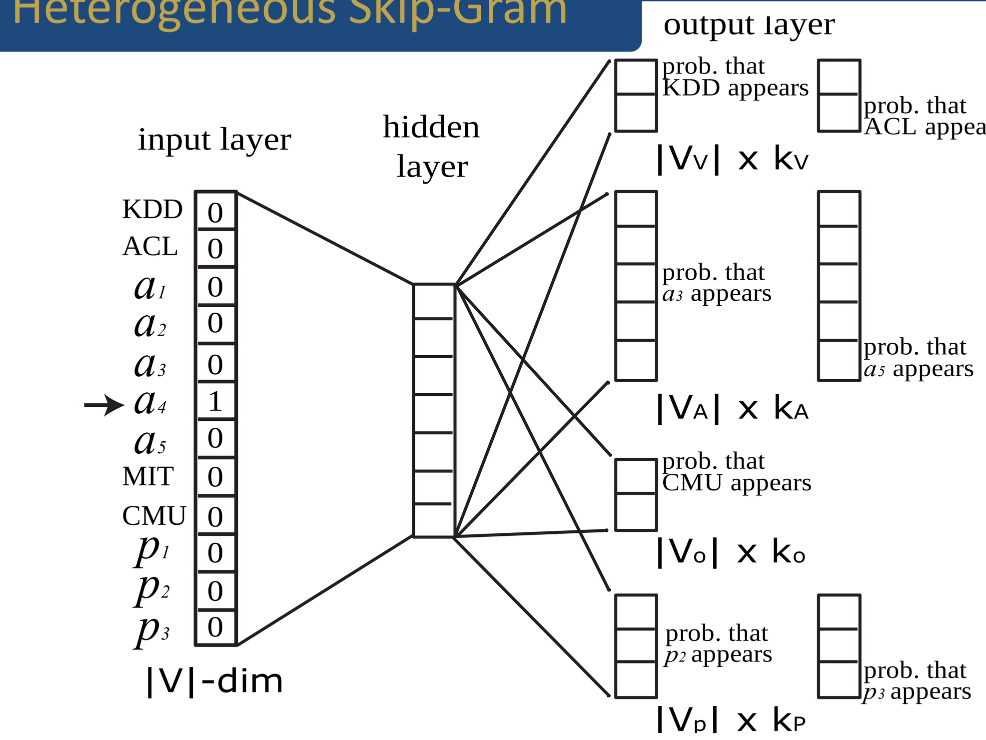
Meta-Path-Based Random Walks

We design meta-path-based random walks to generate paths that are able to capture both the semantic and structural correlations between different types of nodes, facilitating the transformation of heterogeneous network structures into metapath2vec's skip-gram.

- Given a heterogeneous network *G = (V, E, T)* and a meta-path scheme
$$\mathcal{P}: V_1 \xrightarrow{R_1} V_2 \xrightarrow{R_2} \cdots V_t \xrightarrow{R_t} V_{t+1} \cdots \xrightarrow{R_{l-1}} V_l$$
- The transition probability at step *i* is defined as
$$p(v^{i+1}|v_t^i, \mathcal{P}) = \begin{cases} \frac{1}{|N_{t+1}(v_t^i)|} & (v^{i+1}, v_t^i) \in E, \phi(v^{i+1}) = t{+}1 \\ 0 & (v^{i+1}, v_t^i) \in E, \phi(v^{i+1}) \neq t{+}1 \\ 0 & (v^{i+1}, v_t^i) \notin E \end{cases}$$
- In addition, meta-paths are commonly used in a symmetric way, that is, its first node type is the same with the last one [5], facilitating its recursive guidance for random walkers, i.e.,
$$p(v^{i+1}|v_t^i) = p(v^{i+1}|v_1^i), \text{ if } t = l$$

Example: In a traditional random walk procedure, in the toy example, the next step of a walker on node a_4 transitioned from node *CMU* can be all types of nodes surrounding it—a_2, a_3, a_5, p_2, p_3, and *CMU*. However, under the meta-path scheme '*OAPVPAO*', for example, the walker is biased towards paper nodes (*P*) given its previous step on an organization node *CMU* (*O*), following the semantics of this meta-path.

Heterogeneous Skip-Gram



- The heterogeneous skip-gram model used in metapath2vec++ when predicting for a_4. Instead of one set of multinomial distributions for all types of neighborhood nodes in the output layer, it specifies one set of multinomial distributions for each type of nodes in a_4's neighborhood.
- Network maximization in both
$$\arg\max_{\theta} \sum_{v \in V} \sum_{t \in T_V} \sum_{c_t \in N_t(v)} \log p(c_t|v;\theta)$$
- Softmax in metapath2vec
$$p(c_t|v;\theta) = \frac{e^{X_{c_t} \cdot X_v}}{\sum_{u \in V} e^{X_u \cdot X_v}}$$
- Softmax in metapath2vec++
$$p(c_t|v;\theta) = \frac{e^{X_{c_t} \cdot X_v}}{\sum_{u_t \in V_t} e^{X_{u_t} \cdot X_v}}$$
- Objective function in metapath2vec++ (heterogeneous negative sampling)
$$O(\mathbf{X}) = \log \sigma(X_{c_t} \cdot X_v) + \sum_{m=1}^{M} \mathbb{E}_{u_t^m \sim P_t(u_t)}[\log \sigma(-X_{u_t^m} \cdot X_v)]$$

Heterogeneous Network Data

- **AMiner [6]:** ~~9~~ 1.7 million authors, 3 million papers, 3800+ venues, & 8 categories of venues for labeling venues & authors.
  - Computer Linguistics
  - Computer Graphics
  - Computer Networks
  - Computer Vision
  - Computing Systems
  - Databases & Info
  - Human Computer Interaction
  - Theoretical Computer Science
- **meta-path:** *APVPA*
- #walks per node ***w***: 1000
- walk length ***l***: 100
- vector dimension ***d***: 128
- neighborhood size ***k***: 7
- #negative-samples: 5
- **DBIS [5]**: 5 thousand authors, 72 thousand papers, 464 venues.

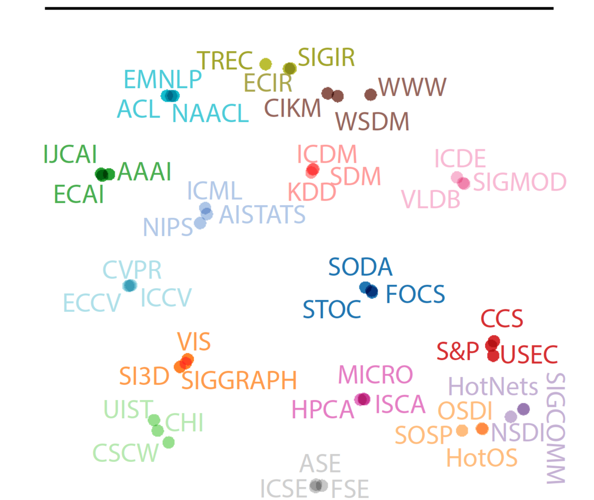
Experiments: Label Prediction

Multi-class venue node classification results in AMiner data

Metric	Method	5%	10%	20%	30%	40%	50%
Macro-F1	DeepWalk/node2vec	0.0723	0.1396	0.1905	0.2795	0.3427	0.3911
	LINE (1st+2nd)	0.2245	0.4629	0.7011	0.8473	0.8953	0.9203
	PTE	0.1702	0.3388	0.6535	0.8304	0.8936	0.9210
	metapath2vec	0.3033	0.5247	0.8033	0.8971	0.9406	0.9532
	metapath2vec++	0.3090	0.5444	0.8049	0.8995	0.9468	0.9580
Micro-F1	DeepWalk/node2vec	0.1701	0.2142	0.2486	0.3266	0.3788	0.4090
	LINE (1st+2nd)	0.3000	0.5167	0.7159	0.8457	0.8950	0.9209
	PTE	0.2512	0.4267	0.6879	0.8372	0.8950	0.9239
	metapath2vec	0.4173	0.5975	0.8327	0.9011	0.9400	0.9522
	metapath2vec++	0.4331	0.6192	0.8336	0.9032	0.9463	0.9582

Experiments: Clustering

methods	venue	author
DeepWalk/node2vec	0.1952	0.2941
LINE (1st+2nd)	0.8967	0.6423
PTE	0.9060	0.6483
metapath2vec	0.9274	0.7470
metapath2vec++	0.9261	0.7354

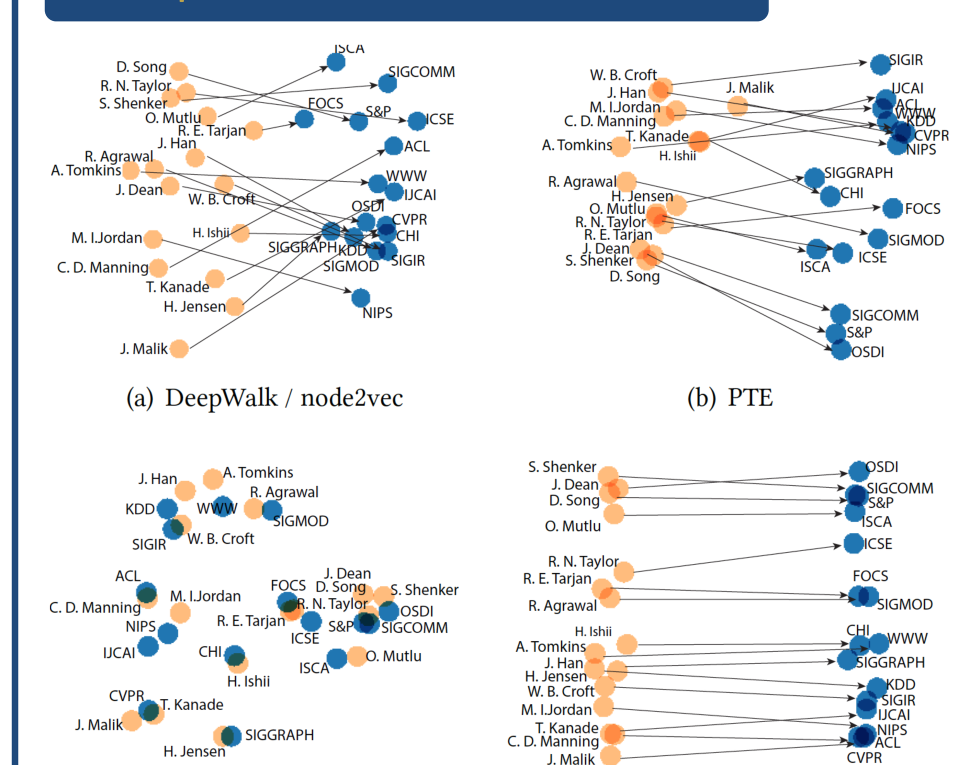


Experiments: Similarity Search

ACL	NIPS	IJCAI	CVPR	FOCS	SOSP	ISCA	S&P
ACL	NIPS	IJCAI	CVPR	FOCS	SOSP	ISCA	S&P
EMNLP	ICML	AAAI	ECCV	STOC	TOCS	HPCA	CCS
NAACL	AISTATS	AI	ICCV	SICOMP	OSDI	MICRO	NDSS
CL	JMLR	JAIR	IJCV	SODA	HotOS	ASPLOS	USENIX S
CoNLL	NC	ECAI	ACCV	A-R	SIGOPS E	PACT	ACSAC
COLING	MLJ	KR	CVIU	TALG	ATC	ICS	JCS
IJCNLP	COLT	AI Mag	BMVC	ICALP	NSDI	HiPEAC	ESORICS
NLE	UAI	ICAPS	ICPR	ECCC	OSR	PPOPP	TISS
ANLP	KDD	CI	EMMCVPR	TOC	ASPLOS	ICCD	ASIACCS
LREC	CVPR	AIPS	T on IP	JAlG	EuroSys	CGO	RAID
EACL	ECML	UAI	WACV	ITCS	SIGCOMM	ISLPED	CSFW

ICSE	SIGGRAPH	SIGCOMM	CHI	KDD	SIGMOD	SIGIR	WWW
ICSE	SIGGRAPH	SIGCOMM	CHI	KDD	SIGMOD	SIGIR	WWW
TOSEM	TOG	CCR	CSCW	SDM	PVLDB	ECIR	WSDM
FSE	SI3D	HotNets	TOCHI	TKDD	ICDE	CIKM	CIKM
ASE	RT	NSDI	UIST	ICDM	DE Bull	IR J	TWEB
ISSTA	CGF	CoNEXT	DIS	DMKD	VLDBJ	TREC	ICWSM
E SE	NPAR	IMC	HCI	KDD E	EDBT	SIGIR F	HT
MSR	Vis	TON	MobileHCI	WSDM	TODS	ICTIR	SIGIR
ESEM	JGT	INFOCOM	INTERACT	CIKM	CIDR	WSDM	KDD
A SE	VisComp	PAM	GROUP	PKDD	SIGMOD R	TOIS	TIT
ICPC	GI	MobiCom	NordiCHI	ICML	WebDB	IPM	WISE
WICSA	CG	IPTPS	UbiComp	PAKDD	PODS	AIRS	WebSci

Experiments: Visualization



(a) DeepWalk / node2vec (b) PTE (c) *metapath2vec* (d) *metapath2vec++*

References

1. T. Mikolov, et al. Distributed Representations of Words and Phrases and Their Compositionality. In *NIPS 2013*.
2. B. Perozzi, et al. DeepWalk: Online Learning of Social Representations. In *KDD 2014.*
3. J. Tang, et al. LINE: Large-scale Information Network Embedding. In *WWW* 2017.
4. Y Bengio, et al. 2013. Representation learning: A review and new perspectives. *IEEE TPAMI 2013.*
5. Y. Sun & J. Han. Mining Heterogeneous Information Networks: Principles and Methodologies. *Morgan & Claypool Publishers.*
6. J. Tang, et al. ArnetMiner: Extraction and Mining of Academic Social Networks. In *KDD 2008.*

Acknowledgements

Paper Info: (Poster Slot #41)

KDD2017

Yuxiao Dong, Nitesh V. Chawla, Ananthram Swami.
metapath2vec: Scalable Representation Learning for Heterogeneous Networks.
In *Proc. of 23rd ACM SIGKDD Conf. on Knowledge Discovery and Data Mining (KDD'17).*

Data & Code