metapath2vec
Scalable Representation Learning for Heterogeneous Networks

Yuxiao Dong
Microsoft Research & Notre Dame

Nitesh V. Chawla
University of Notre Dame

Ananthram Swami
Army Research Lab

Interdisciplinary Center for Network Science and Applications (iCeNSA)
University of Notre Dame
Conventional Network Mining and Learning

Network Mining Tasks
- node attribute inference
- community detection
- similarity search
- link prediction
- social recommendation
- …

feature engineering → hand-crafted feature matrix → machine learning models
Network Embedding for Mining and Learning

Network Mining Tasks
- node attribute inference
- community detection
- similarity search
- link prediction
- social recommendation
- …

latent representation matrix

feature learning

machine learning models

Word Embedding in NLP

- **Input**: a text corpus \( D = \{W\} \)
- **Output**: \( X \in R^{|W| \times d}, d \ll |W| \), \( d \)-dim vector \( X_w \) for each word \( w \).


Geographically close words---a word and its context words---in a sentence or document exhibit interrelations in human natural language.
Network Embedding

- **Input:** a network $G = (V, E)$
- **Output:** $X \in R^{|V| \times d}$, $d \ll |V|$, $d$-dim vector $X_v$ for each node $v$.

DeepWalk [Perozzi et al., KDD14]

Heterogeneous Network Embedding: Problem

- **Input:** a heterogeneous information network \( G = (V, E, T) \)
- **Output:** \( X \in \mathbb{R}^{|V| \times d}, d \ll |V| \), \( d \)-dim vector \( X_v \) for each node \( v \).

Latent representation vector

\( \text{Org, Author, Paper, Venue} \)
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 also difficult for conventional meta-path based methods to model similarities between nodes without connected meta-paths.
Heterogeneous Network Embedding: Solutions

- metapath2vec
- metapath2vec++
- meta-path-based random walks
- skip-gram
- heterogeneous skip-gram
Goal: 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 skip-gram.
metapath2vec: Meta-Path-Based Random Walks

Given 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

\[
p(v^{i+1} | v^i, \mathcal{P}) = \begin{cases} 
\frac{1}{|N_{t+1}(v^i_t)|} & (v^{i+1}, v^i_t) \in E, \phi(v^{i+1}) = t+1 \\
0 & (v^{i+1}, v^i_t) \in E, \phi(v^{i+1}) \neq t+1 \\
0 & (v^{i+1}, v^i_t) \notin E 
\end{cases}
\]

Recursive guidance for random walkers, i.e.,

\[ p(v^{i+1} | v^i_t) = p(v^{i+1} | v^i_1), \text{ if } t = l \]
metapath2vec: Meta-Path-Based Random Walks

- Given a meta-path scheme (Example)
  
  \textit{OAPVPAO}

- In a traditional random walk procedure, in the toy example, the next step of a walker on node a4 transitioned from node CMU can be all types of nodes surrounding it—\textit{a2, a3, a5, p2, p3, and CMU}.

- 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.
The potential issue of skip-gram for heterogeneous network embedding:

To predict the context node $c_t$ (type $t$) given a node $v$, metapath2vec encourages all types of nodes to appear in this context position.
metapath2vec++

meta-path-based random walks

meta paths

- APA
- APVPA
- OAPVPA

heterogeneous skip-gram

input layer

hidden layer

output layer

prob. that KDD appears
prob. that ACL appears
prob. that $a_7$ appears
prob. that $a_9$ appears
prob. that CMU appears
prob. that $p_2$ appears
prob. that $p_3$ appears

$|V|\times\text{dim}$

$|V_p|\times k_p$

$|V_o|\times k_o$

$|V_A|\times k_A$

$|V_v|\times k_V$
metapath2vec++: Heterogeneous Skip-Gram

- Objective function (heterogeneous negative sampling)

\[ \mathcal{O}(X) = \log \sigma(X_{ct} \cdot X_v) + \sum_{k=1}^{K} \mathbb{E}_{u_t^k \sim P(u_t)} [\log \sigma(-X_{u_t^k} \cdot X_v)] \]

- Softmax in metapath2vec

\[ p(c_t|v; \theta) = \frac{e^{X_{ct}} \cdot e^{X_v}}{\sum_{u \in V} e^{X_u} \cdot e^{X_v}} \]

- Softmax in metapath2vec++

\[ p(c_t|v; \theta) = \frac{e^{X_{ct}} \cdot e^{X_v}}{\sum_{u_t \in V_t} e^{X_{u_t}} \cdot e^{X_v}} \]

- Stochastic gradient descent

\[ \frac{\partial \mathcal{O}(X)}{\partial X_{u_t^k}} = (\sigma(X_{u_t^k} \cdot X_v - I_{ct}[u_t^k]))X_v \]

\[ \frac{\partial \mathcal{O}(X)}{\partial X_v} = \sum_{k=0}^{K} (\sigma(X_{u_t^k} \cdot X_v - I_{ct}[u_t^k]))X_{u_t^k} \]

**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 $X \in \mathbb{R}^{|V| \times d}$

Initialize $X$;

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

return $X$;

$\text{MetaPathRandomWalk}(G, \mathcal{P}, v, l)$

$MP[1] = v$;

for $i = 1 \rightarrow l-1$ do
  draw $u$ according to Eq. 3;
  $MP[i+1] = u$;
end

return $MP$;

$\text{HeterogeneousSkipGram}(X, k, MP)$

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

- every sub-procedure is easy to parallelize
- 24-32X speedup by using 40 cores
Network Mining and Learning Paradigm

Network Applications
- node attribute inference
- community detection
- similarity search
- link prediction
- social recommendation
- ...

Latent representation vector
Experiments

Heterogeneous Data
- AMiner Academic Network
  - 9-1.7 million authors
  - 3 million papers
  - 3800+ venues
  - 8 research areas

Baselines
- DeepWalk [KDD ’14]
- node2vec [KDD ’16]
- LINE [WWW ’15]
- PTE [KDD ’15]

Parameters
- #walks: 1000
- walk-length: 100
- #dimensions: 128
- neighborhood size: 7

Mining Tasks
- node classification
  - logistic regression
- node clustering
  - k-means
- similarity search
  - cosine similarity

https://aminer.org/aminernetwork
## Application 1: Multi-Class Node Classification

### Table 2: Multi-class node classification results in AMiner data.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Method</th>
<th>5%</th>
<th>10%</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
<th>60%</th>
<th>70%</th>
<th>80%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Macro-F1</strong></td>
<td>DeepWalk/node2vec</td>
<td>0.0723</td>
<td>0.1396</td>
<td>0.1905</td>
<td>0.2795</td>
<td>0.3427</td>
<td>0.3911</td>
<td>0.4424</td>
<td>0.4774</td>
<td>0.4955</td>
<td>0.4457</td>
</tr>
<tr>
<td></td>
<td>LINE (1st+2nd)</td>
<td>0.2245</td>
<td>0.4629</td>
<td>0.7011</td>
<td>0.8473</td>
<td>0.8953</td>
<td>0.9203</td>
<td>0.9308</td>
<td>0.9466</td>
<td>0.9410</td>
<td>0.9466</td>
</tr>
<tr>
<td></td>
<td>PTE</td>
<td>0.1702</td>
<td>0.3388</td>
<td>0.6535</td>
<td>0.8304</td>
<td>0.8936</td>
<td>0.9210</td>
<td>0.9352</td>
<td>0.9505</td>
<td>0.9525</td>
<td>0.9489</td>
</tr>
<tr>
<td></td>
<td>metapath2vec</td>
<td>0.3033</td>
<td>0.5247</td>
<td>0.8033</td>
<td>0.8971</td>
<td>0.9406</td>
<td>0.9532</td>
<td>0.9529</td>
<td>0.9701</td>
<td>0.9683</td>
<td>0.9670</td>
</tr>
<tr>
<td></td>
<td>metapath2vec++</td>
<td>0.3090</td>
<td>0.5444</td>
<td>0.8049</td>
<td>0.8995</td>
<td>0.9468</td>
<td>0.9580</td>
<td>0.9561</td>
<td>0.9675</td>
<td>0.9533</td>
<td>0.9503</td>
</tr>
<tr>
<td><strong>Micro-F1</strong></td>
<td>DeepWalk/node2vec</td>
<td>0.1701</td>
<td>0.2142</td>
<td>0.2486</td>
<td>0.3266</td>
<td>0.3788</td>
<td>0.4090</td>
<td>0.4630</td>
<td>0.4975</td>
<td>0.5259</td>
<td>0.5286</td>
</tr>
<tr>
<td></td>
<td>LINE (1st+2nd)</td>
<td>0.3000</td>
<td>0.5167</td>
<td>0.7159</td>
<td>0.8457</td>
<td>0.8950</td>
<td>0.9209</td>
<td>0.9333</td>
<td>0.9500</td>
<td>0.9556</td>
<td>0.9571</td>
</tr>
<tr>
<td></td>
<td>PTE</td>
<td>0.2512</td>
<td>0.4267</td>
<td>0.6879</td>
<td>0.8372</td>
<td>0.8950</td>
<td>0.9239</td>
<td>0.9352</td>
<td>0.9550</td>
<td>0.9667</td>
<td>0.9571</td>
</tr>
<tr>
<td></td>
<td>metapath2vec</td>
<td>0.4173</td>
<td>0.5975</td>
<td>0.8327</td>
<td>0.9011</td>
<td>0.9400</td>
<td>0.9522</td>
<td>0.9537</td>
<td>0.9725</td>
<td>0.9815</td>
<td>0.9857</td>
</tr>
<tr>
<td></td>
<td>metapath2vec++</td>
<td>0.4331</td>
<td>0.6192</td>
<td>0.8336</td>
<td>0.9032</td>
<td>0.9463</td>
<td>0.9582</td>
<td>0.9574</td>
<td>0.9700</td>
<td>0.9741</td>
<td>0.9786</td>
</tr>
</tbody>
</table>
### Application 1: Multi-Class Node Classification

Table 3: Multi-class **author** node classification results in AMiner data.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Method</th>
<th>5%</th>
<th>10%</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
<th>60%</th>
<th>70%</th>
<th>80%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DeepWalk/node2vec</td>
<td>0.7153</td>
<td>0.7222</td>
<td>0.7256</td>
<td>0.7270</td>
<td>0.7273</td>
<td>0.7274</td>
<td>0.7273</td>
<td>0.7271</td>
<td>0.7275</td>
<td>0.7275</td>
</tr>
<tr>
<td></td>
<td>LINE (1st+2nd)</td>
<td>0.8849</td>
<td>0.8886</td>
<td>0.8911</td>
<td>0.8921</td>
<td>0.8926</td>
<td>0.8929</td>
<td>0.8934</td>
<td>0.8936</td>
<td>0.8938</td>
<td>0.8934</td>
</tr>
<tr>
<td></td>
<td>PTE</td>
<td>0.8898</td>
<td>0.8940</td>
<td>0.897</td>
<td>0.8982</td>
<td>0.8987</td>
<td>0.8990</td>
<td>0.8997</td>
<td>0.9002</td>
<td>0.9005</td>
<td></td>
</tr>
<tr>
<td></td>
<td>metapath2vec</td>
<td>0.9216</td>
<td>0.9262</td>
<td>0.9292</td>
<td>0.9303</td>
<td>0.9309</td>
<td>0.9314</td>
<td>0.9315</td>
<td>0.9316</td>
<td>0.9319</td>
<td>0.9320</td>
</tr>
<tr>
<td></td>
<td>metapath2vec++</td>
<td>0.9107</td>
<td>0.9156</td>
<td>0.9186</td>
<td>0.9199</td>
<td>0.9204</td>
<td>0.9207</td>
<td>0.9208</td>
<td>0.9211</td>
<td>0.9212</td>
<td></td>
</tr>
<tr>
<td></td>
<td>DeepWalk/node2vec</td>
<td>0.7312</td>
<td>0.7372</td>
<td>0.7402</td>
<td>0.7414</td>
<td>0.7418</td>
<td>0.7420</td>
<td>0.7419</td>
<td>0.7420</td>
<td>0.7425</td>
<td>0.7425</td>
</tr>
<tr>
<td></td>
<td>LINE (1st+2nd)</td>
<td>0.8936</td>
<td>0.8969</td>
<td>0.8993</td>
<td>0.9002</td>
<td>0.9007</td>
<td>0.9010</td>
<td>0.9015</td>
<td>0.9016</td>
<td>0.9018</td>
<td>0.9017</td>
</tr>
<tr>
<td></td>
<td>PTE</td>
<td>0.8986</td>
<td>0.9023</td>
<td>0.9051</td>
<td>0.9061</td>
<td>0.9066</td>
<td>0.9068</td>
<td>0.9075</td>
<td>0.9077</td>
<td>0.9079</td>
<td>0.9082</td>
</tr>
<tr>
<td></td>
<td>metapath2vec</td>
<td>0.9279</td>
<td>0.9319</td>
<td>0.9346</td>
<td>0.9356</td>
<td>0.9361</td>
<td>0.9365</td>
<td>0.9365</td>
<td>0.9365</td>
<td>0.9367</td>
<td>0.9369</td>
</tr>
<tr>
<td></td>
<td>metapath2vec++</td>
<td>0.9173</td>
<td>0.9217</td>
<td>0.9243</td>
<td>0.9254</td>
<td>0.9259</td>
<td>0.9261</td>
<td>0.9261</td>
<td>0.9262</td>
<td>0.9264</td>
<td>0.9266</td>
</tr>
</tbody>
</table>
### Application 2: Node Clustering

Node clustering results (NMI) in AMiner

<table>
<thead>
<tr>
<th>methods</th>
<th>venue</th>
<th>author</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeepWalk/node2vec</td>
<td>0.1952</td>
<td>0.2941</td>
</tr>
<tr>
<td>LINE (1st+2nd)</td>
<td>0.8967</td>
<td>0.6423</td>
</tr>
<tr>
<td>PTE</td>
<td>0.9060</td>
<td>0.6483</td>
</tr>
<tr>
<td>metapath2vec</td>
<td>0.9274</td>
<td>0.7470</td>
</tr>
<tr>
<td>metapath2vec++</td>
<td>0.9261</td>
<td>0.7354</td>
</tr>
</tbody>
</table>

http://projector.tensorflow.org/
# Application 3: Similarity Search

## Table 5: Case study of similarity search in AMiner Data

<table>
<thead>
<tr>
<th>Rank</th>
<th>ACL</th>
<th>NIPS</th>
<th>IJCAI</th>
<th>CVPR</th>
<th>FOCS</th>
<th>SOSP</th>
<th>ISCA</th>
<th>S&amp;P</th>
<th>ICSE</th>
<th>SIGGRAPH</th>
<th>SIGCOMM</th>
<th>CHI</th>
<th>KDD</th>
<th>SIGMOD</th>
<th>SIGIR</th>
<th>WWW</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>ACL</td>
<td>NIPS</td>
<td>IJCAI</td>
<td>CVPR</td>
<td>FOCS</td>
<td>SOSP</td>
<td>ISCA</td>
<td>S&amp;P</td>
<td>ICSE</td>
<td>SIGGRAPH</td>
<td>SIGCOMM</td>
<td>CHI</td>
<td>KDD</td>
<td>SIGMOD</td>
<td>SIGIR</td>
<td>WWW</td>
</tr>
<tr>
<td>1</td>
<td>EMNLP</td>
<td>ICML</td>
<td>AAAI</td>
<td>ECCV</td>
<td>TOCS</td>
<td>TOCS</td>
<td>HPCC</td>
<td>CCS</td>
<td>TOSEM</td>
<td>TOG</td>
<td>CCR</td>
<td>CCR</td>
<td>SDM</td>
<td>PVLDB</td>
<td>ECIR</td>
<td>WSDM</td>
</tr>
<tr>
<td>2</td>
<td>NAACL</td>
<td>AI</td>
<td>IJCAI</td>
<td>ICCV</td>
<td>SICOMP</td>
<td>OSFI</td>
<td>MICRO</td>
<td>NDSS</td>
<td>FSF</td>
<td>S3D</td>
<td>HotNets</td>
<td>TOCHI</td>
<td>TKDD</td>
<td>ICDE</td>
<td>CIKM</td>
<td>CIKM</td>
</tr>
<tr>
<td>3</td>
<td>CL</td>
<td>JMLR</td>
<td>JAIR</td>
<td>IIJCV</td>
<td>SODA</td>
<td>HotOS</td>
<td>ASPLOS</td>
<td>USENIXS</td>
<td>ASE</td>
<td>RT</td>
<td>NSDI</td>
<td>UIST</td>
<td>ICDM</td>
<td>DE Bull</td>
<td>IRJ</td>
<td>TWEB</td>
</tr>
<tr>
<td>4</td>
<td>CoNLL</td>
<td>NC</td>
<td>ECAI</td>
<td>ACCV</td>
<td>A-R</td>
<td>SIGOPS</td>
<td>E</td>
<td>PACT</td>
<td>ACSAC</td>
<td>ISSTA</td>
<td>CGF</td>
<td>CoNEXT</td>
<td>DIS</td>
<td>DMKD</td>
<td>VLDB</td>
<td>TREC</td>
</tr>
<tr>
<td>5</td>
<td>COLING</td>
<td>MIJ</td>
<td>KR</td>
<td>CVIU</td>
<td>TALG</td>
<td>ATC</td>
<td>ICS</td>
<td>JCS</td>
<td>ESE</td>
<td>NPAR</td>
<td>IMC</td>
<td>HCI</td>
<td>KDD E</td>
<td>EDBT</td>
<td>SIGIR F</td>
<td>HT</td>
</tr>
<tr>
<td>6</td>
<td>IJCNLP</td>
<td>COLT</td>
<td>AI Mag</td>
<td>BMVC</td>
<td>ICALP</td>
<td>NSDI</td>
<td>HiPEAC</td>
<td>ESORICS</td>
<td>MSR</td>
<td>Vis</td>
<td>TON</td>
<td>MobileHCI</td>
<td>WSDM</td>
<td>TOSD</td>
<td>ICTIR</td>
<td>SIGIR</td>
</tr>
<tr>
<td>7</td>
<td>NLE</td>
<td>UAI</td>
<td>ICAI</td>
<td>IPR</td>
<td>ECCC</td>
<td>OSR</td>
<td>PPOPP</td>
<td>TISS</td>
<td>ESEM</td>
<td>JGT</td>
<td>INFOCOM</td>
<td>INTERACT</td>
<td>CIKM</td>
<td>CIDR</td>
<td>WSDM</td>
<td>KDD</td>
</tr>
<tr>
<td>8</td>
<td>ANLP</td>
<td>KDD</td>
<td>CI</td>
<td>EMMCVPR</td>
<td>TOC</td>
<td>ASPLOS</td>
<td>ICCD</td>
<td>ASIAACS</td>
<td>A SE</td>
<td>VisComp</td>
<td>PAM</td>
<td>GROUP</td>
<td>PKDD</td>
<td>SIGMOD R</td>
<td>TOIS</td>
<td>TIT</td>
</tr>
<tr>
<td>9</td>
<td>LREC</td>
<td>CVPR</td>
<td>ALS</td>
<td>T on IP</td>
<td>JAG</td>
<td>EuroSys</td>
<td>CGO</td>
<td>RAID</td>
<td>ICPC</td>
<td>GI</td>
<td>MobiCom</td>
<td>NordicHI</td>
<td>ICML</td>
<td>WebDB</td>
<td>IPM</td>
<td>WISE</td>
</tr>
<tr>
<td>10</td>
<td>EACL</td>
<td>ECML</td>
<td>UAI</td>
<td>WACV</td>
<td>ITCS</td>
<td>SIGCOMM</td>
<td>ISLPED</td>
<td>CSFW</td>
<td>WICSA</td>
<td>CG</td>
<td>IPTPS</td>
<td>UbiComp</td>
<td>PAKDD</td>
<td>PODS</td>
<td>AIRS</td>
<td>WebSci</td>
</tr>
</tbody>
</table>
Visualization

(a) DeepWalk/node2vec

(b) PTE

(c) metapath2vec

(d) metapath2vec++

word2vec [Mikolov, 2013]

http://projector.tensorflow.org/
Problem: Heterogeneous Network Embedding

Models: metapath2vec & metapath2vec++

- The automatic discovery of internal semantic relationships between different types of nodes in heterogeneous networks

Applications: classification, clustering, & similarity search
Thank you!

Data & Code

https://ericdongyx.github.io/metapath2vec/m2v.html