MLG Spotlight Talks

August 20th, 2018

Growing Better Graphs with Latent-Variable Probabilistic Graph

Xinyi Wang, Salvador Aguinaga, Tim Weninger, David Chiang

Background and Problems

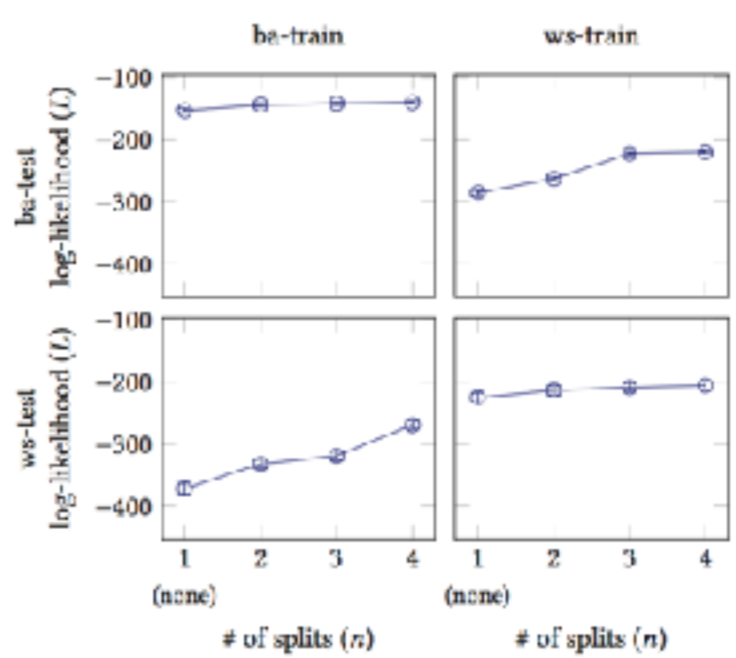
- Hyperedge Replacement Grammar (HRG)
  - Generate graphs like CFG generating strings
  - Extract from the tree decomposition of a graph
  - **Problem: Lack of context**
- Graph Generator
  - **Problem: Evaluate on training data**

Solution: latent variable HRG

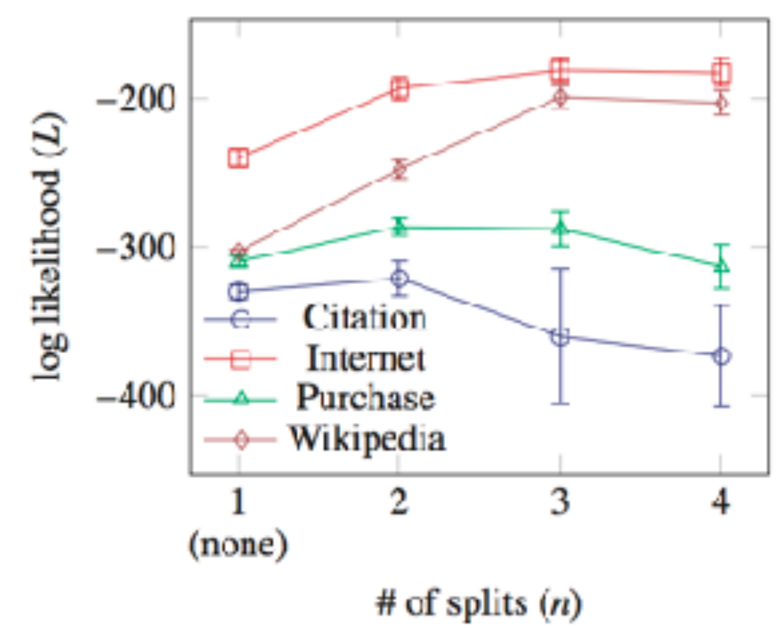
- Nonterminal Splitting **[add CONTEXT to HRG]**
  - Learning with Expectation Maximization
  - Objective: max P(training graphs)
  - Rule splits to differentiate contexts
- Evaluation **[robust and accurate]**
  - Log likelihood of **TEST** graph

Experiments and Results

- Train/test graphs
  - Two synthesized graphs
  - Four real world graphs
- Left: log likelihood is an effective metric
- Right: latent variable HRG improves over HRG
- Comparable with other graph generators in terms of GCD



- On the test graph of similar structure with the training graph, the log likelihood is higher than the test graph of different structure



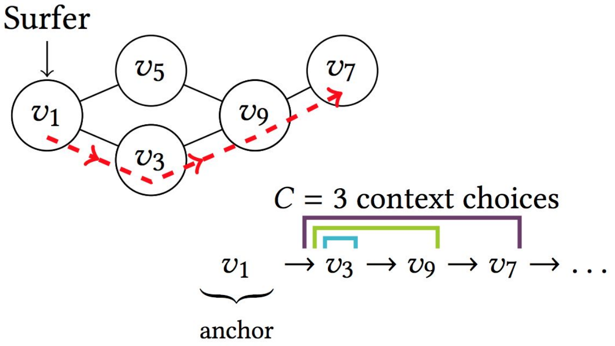
- Log Likelihood always maximize at number of split n > 1

Watch Your Step: Learning Graph Embeddings through Attention

Sami Abu-El-Haija[1,2], *Bryan Perozzi*[2], *Rami Al-Rfou*[2], *Alex Alemi*[2] *Information Sciences Institute*[1] Google AI[2]

Task: Node Embeddings

- Goal: Learn Node Embeddings. Useful for various tasks (Link Prediction & Node Classification)
- modern methods pass random walk sequences to word2vec [1], which samples context using uniform distribution:



E[statistics]

We derive analytical solution on (anchor, context) sampling:

$$\mathbb{E}\left[\mathbf{D}^{\text{DeepWalk}[C]}\right] = \tilde{\mathbf{P}}^{(0)} \sum_{k=1}^{C} \left[1 - \frac{k-1}{C}\right] (\mathcal{T})^k$$

$$\mathbb{E}\left[\mathbf{D}^{\text{GloVe}[C]}\right] = \tilde{\mathbf{P}}^{(0)} \sum_{k=1}^{C} \frac{1}{k} (\mathcal{T})^k$$

Ours:

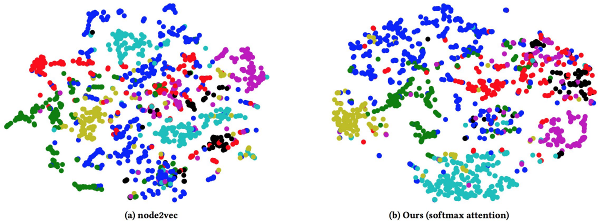
We **train the context distribution** jointly with embeddings:

$$\mathbb{E}\left[\mathbf{D} \mid Q_1, Q_2, \ldots Q_C\right] = \tilde{\mathbf{P}}^{(0)} \sum_{k=1}^{C} Q_k (\mathcal{T})^k$$

Our Objective: extends [2]

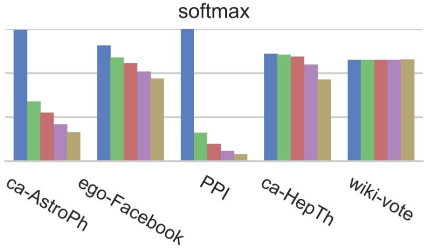
$$\min_{\mathbf{L},\mathbf{R},\mathbf{q}} \beta ||\mathbf{q}||_2^2 + \Big\| -\mathbb{E}[\mathbf{D} \mid \mathbf{q}] \circ \log\left(\sigma(\mathbf{L} \times \mathbf{R}^T)\right) - \mathbb{1}[\mathbf{A} = 0] \circ \log\left(1 - \sigma(\mathbf{L} \times \mathbf{R}^T)\right) \Big\|_1$$

t-SNE: node2vec [3] VS ours



(a) node2vec (b) Ours (softmax attention)

Learned Q: differs per net



Results: reduce errors (link prediction by 20%-40%; Node classification by up to 10%)

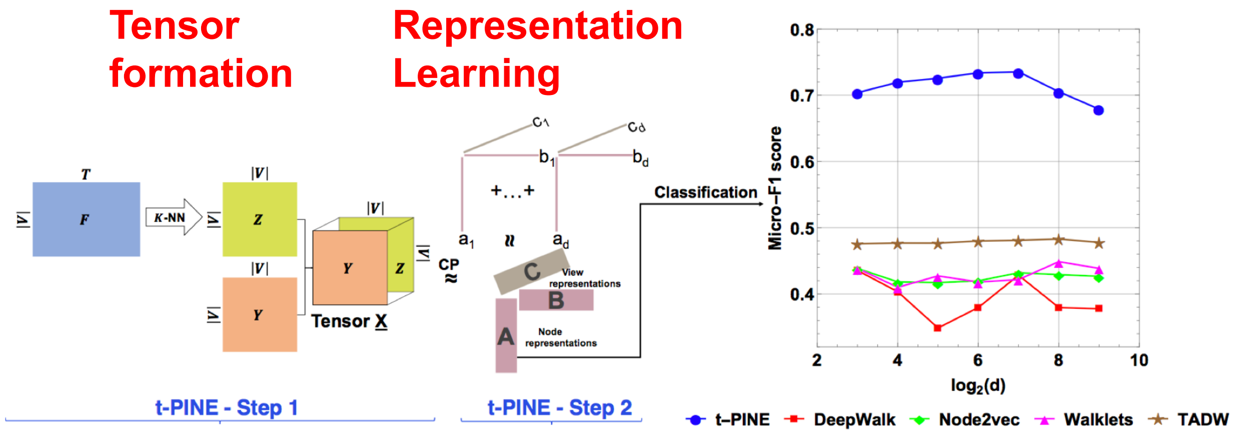
[1] Perozzi et al, *DeepWalk*, KDD'14 [2] Abu-El-Haija et al, *AsymProj*, CIKM'17 [3] Grover & Leskovec, *node2vec*, KDD'15

t-PINE: Tensor-based Predictable and Interpretable Node Embeddings

Saba Al-Sayouri, Ekta Gujral, Danai Koutra,
Evangelos E. Papalexakis, and Sarah S. Lam

BINGHAMTON UNIVERSITY STATE UNIVERSITY OF NEW YORK
UC RIVERSIDE
UNIVERSITY OF MICHIGAN

Baselines	Present Gap	t-PINE
Unsatisfactory accuracy	**Accuracy**	Better performance (Multi-view information graph)
Explicit representation learning	**Shallow models**	Explicit & Implicit representation learning
Disjoint explicit & implicit representation learning	**Representations concatenation**	Joint explicit & implicit representation learning (CP decomposition)
Uninterpretable	**Interpretability**	Interpretable



(a): t-PINE (b): Evaluation

Fully Heterogeneous Collective Regression

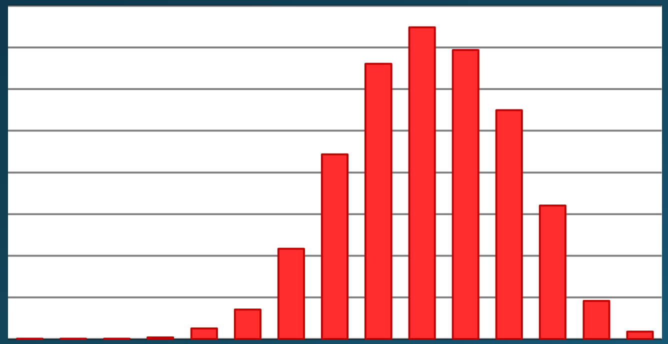
Ensign David J. Liedtka

Adviser: Professor Luke K. McDowell

United States Naval Academy

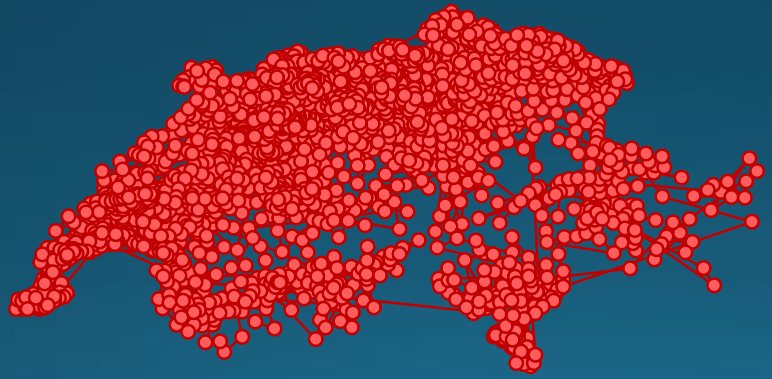
Can exploiting links in relational data lead to greater accuracy in predicting elections as they unfold in real-time?

How to "bootstrap" initial predictions to provide a baseline for inference?



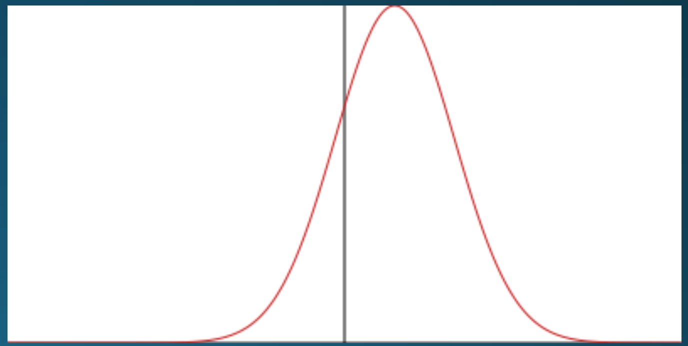
- Combine vote <u>and</u> region features

How to compute links so as to connect the regions into a useful graph?



- Leverage region-to-region <u>correlations</u>

How can we perform effective collective inference?

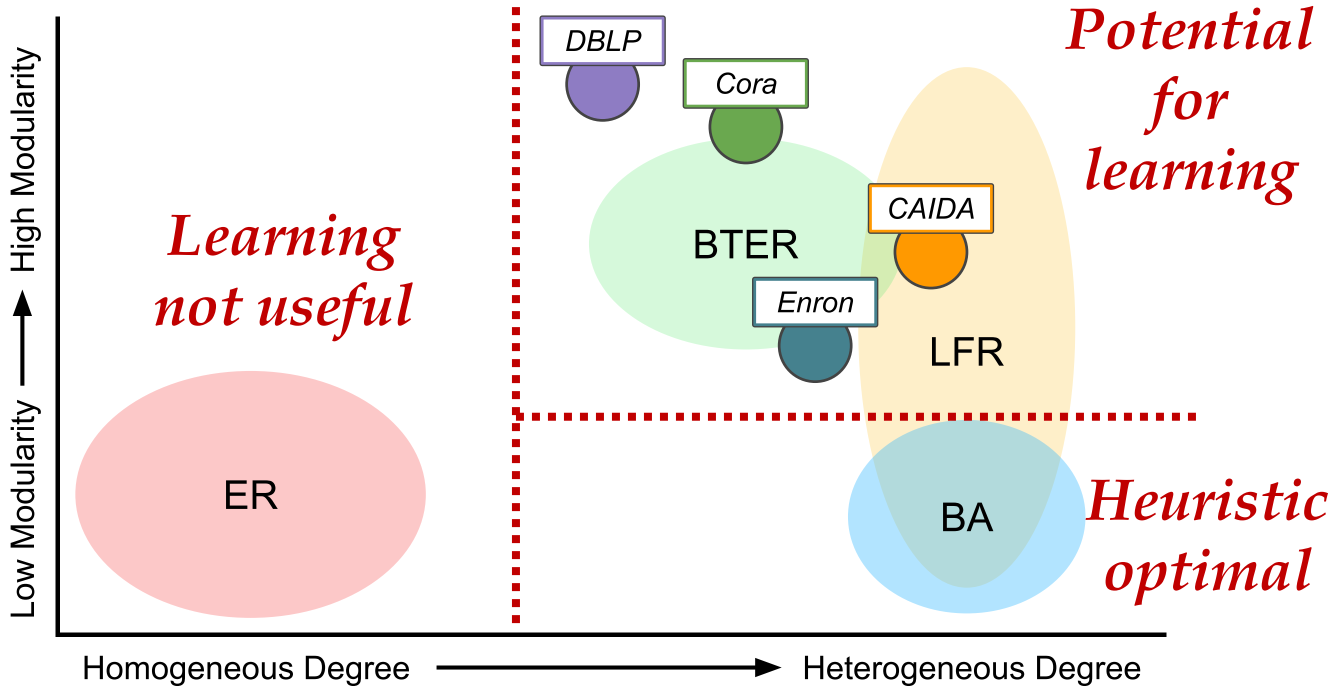


- Executes over 100x <u>faster</u>!

Reducing Network Incompleteness Through Online Learning

Timothy LaRock* Timothy Sakharov* Sahely Bhadra† Tina Eliassi-Rad*
*Northeastern University †IIT Palakkad

- Network data is often incomplete
- Acquiring more data can be expensive and/or hard
- Research question:
  - Given a network and limited resources to collect more data, how can we get the most bang for our buck?

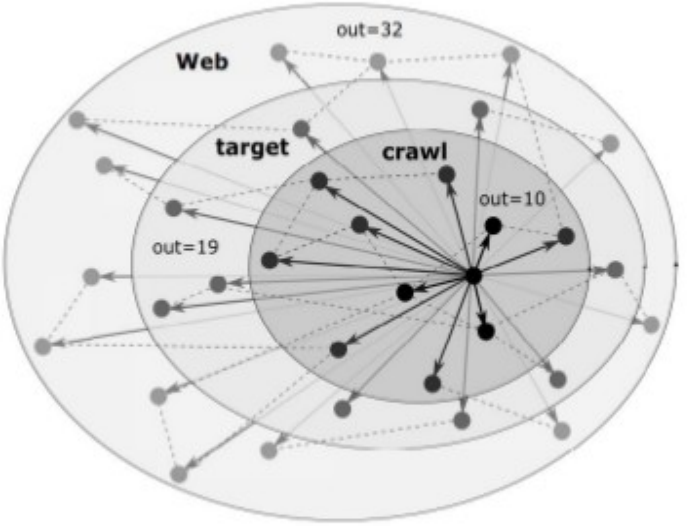


Supported by NSF 1314603

What the HAK ? Estimating Ranking Deviations in Incomplete Graphs

Helge **H**olzmann, Avishek **A**nand, Megha **K**hosla

- Graphs collected on the Web are typically incomplete
- Hypothesis: Incomplete graphs (e.g., crawls, Web archives, ...) cause deviations in random walk algorithms, such as PageRank
- Consequence: Rankings corresponding to PageRank differ from the (unavailable) complete / original graph

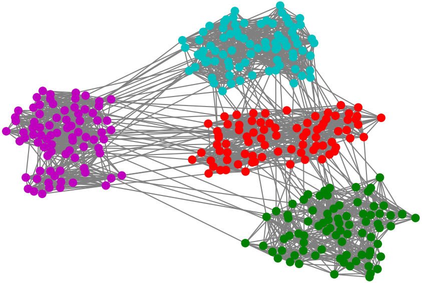


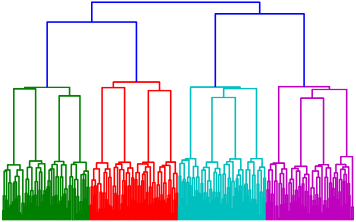
- **RQ I**: Do incomplete real-world graphs show a deviation in their PageRank ?
- **RQ II**: How can we reliably measure the extent of such ranking deviations for incomplete graphs?

Hierarchical Graph Clustering by Node Pair Sampling

Thomas Bonald, Bertrand Charpentier, Alexis Galland, Alexandre Hollocou

- Most real graphs have a **multi-scale** structure
- We propose a novel **hierarchical** graph clustering algorithm
- The algorithm is **agglomerative**, with a distance between clusters induced by **node pair sampling**





Generalized Embedding Model for Knowledge Graph Mining

Contribution

a) Propose GEN, an efficient embedding learning framework for generalized KGs

b) Consider "multi-shot" information for embedding learning simultaneously

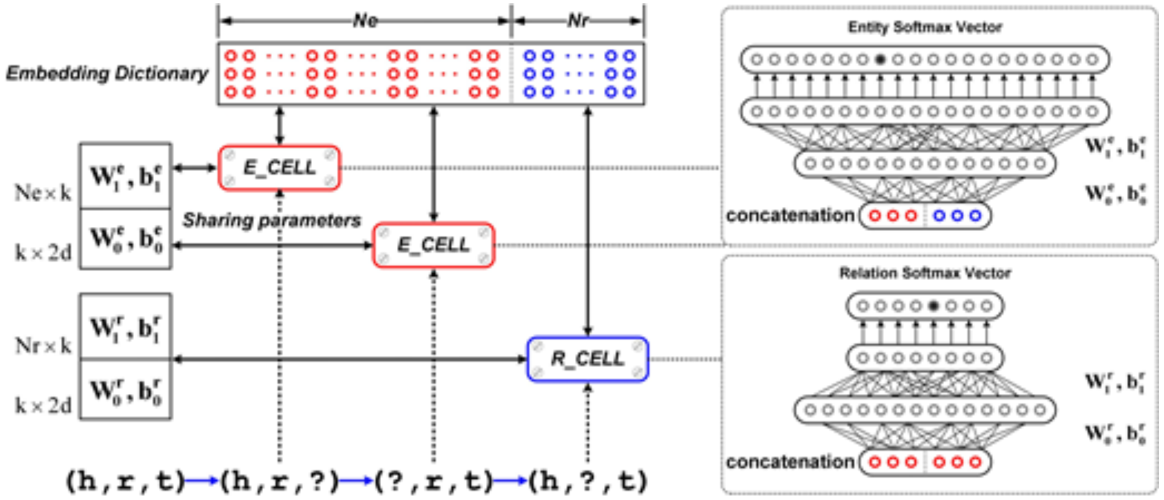
- (Subject, Predicate) ⇒ Object
- (Object, Predicate) ⇒ Subject
- (Subject, Object) ⇒ Predicate

c) We show that GEN can works on graphs in different domains

Task

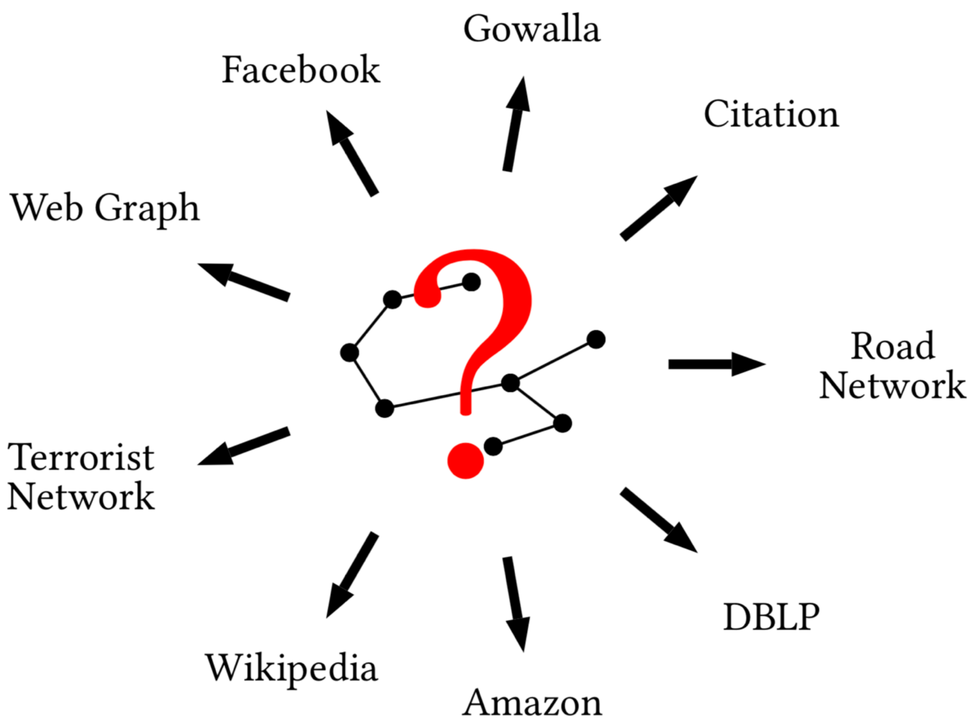
- Learning reasonable and accurate distributed representations for knowledge graph.

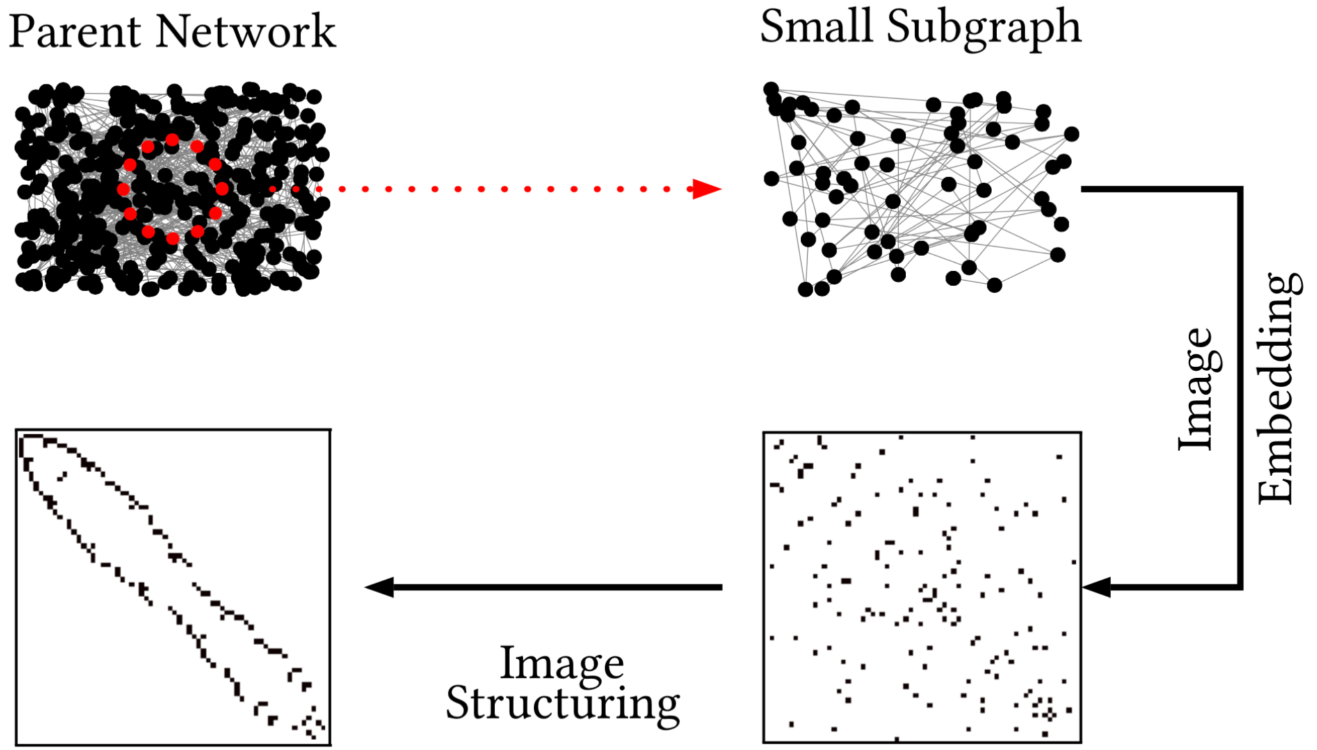
- Flexible enough to adapt to variations networks



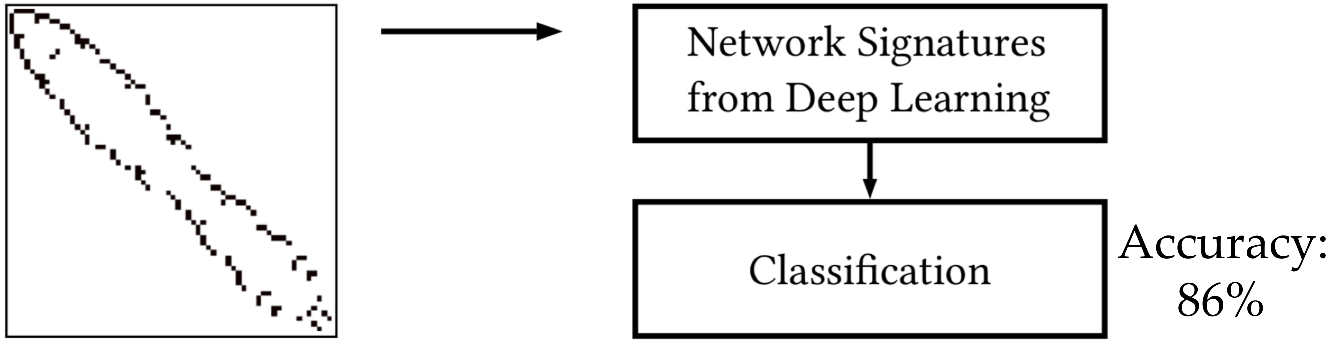
Network Signatures from Image Representation of Adjacency Matrices: Deep/Transfer Learning for Subgraph Classification

Kshiteesh Hegde, Malik Magdon-Ismail, Ram Ramanathan and Bishal Thapa





Classification	Score
Pug	0.75
Bull Mastiff	0.13
Brabancon Giffon	0.04
French Bulldog	0.02
Muzzle	0.01

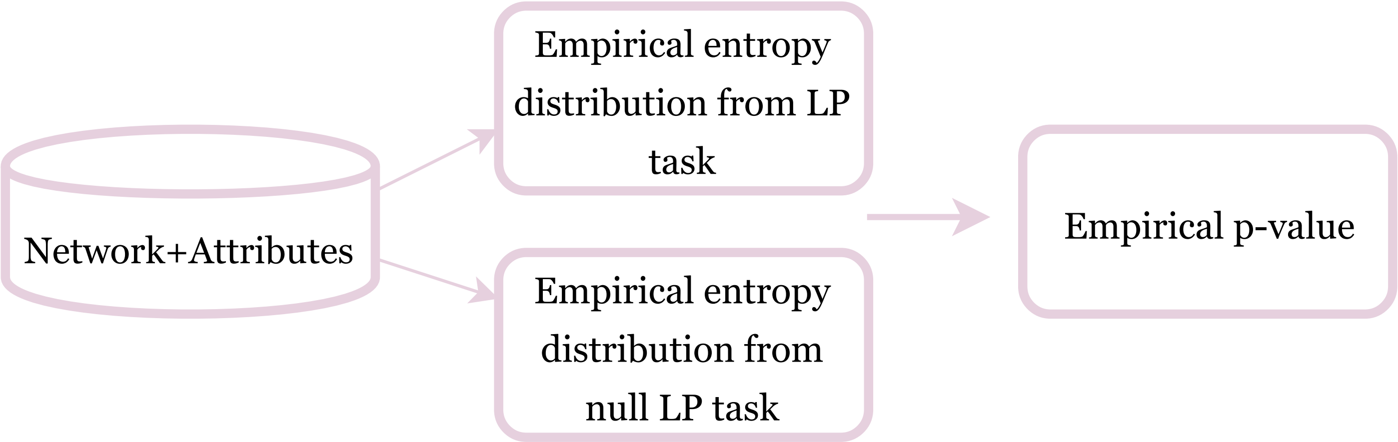


Testing Alignment of Node Attributes with Network Structure through Label Propagation

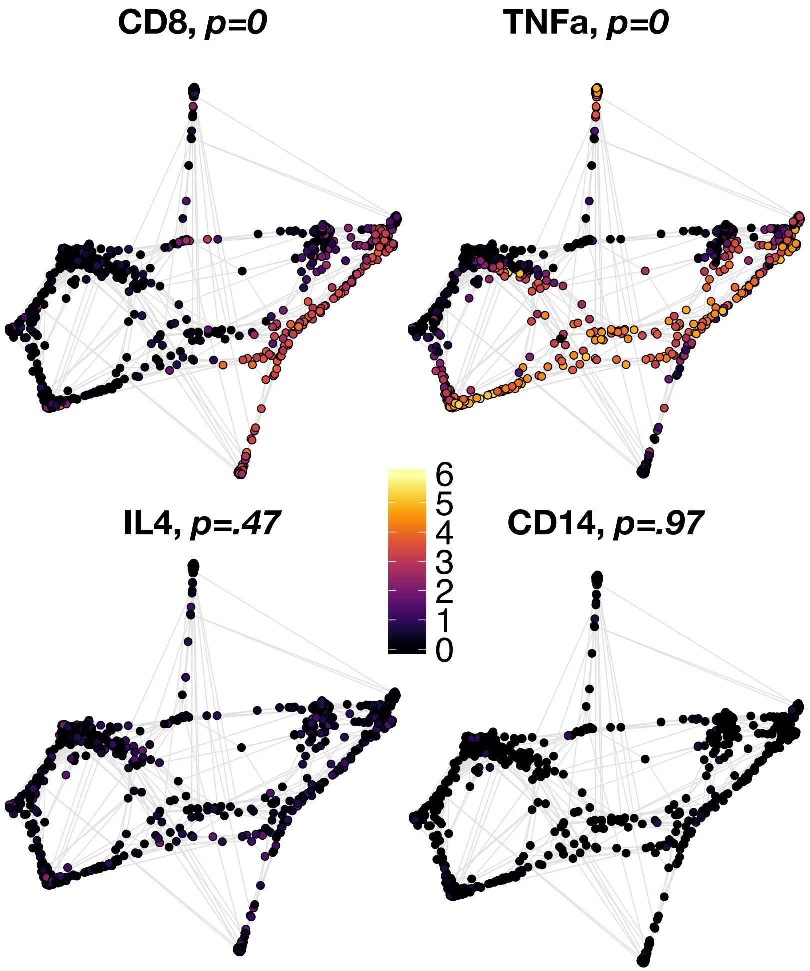
Natalie Stanley (Stanford), Marc Niethammer (UNC-CH), Peter Mucha (UNC-CH)

In this work, we developed test to measure the extent to which node attributes and network connectivity align. This relationship is reflected through an empirical p-value in a label propagation task.

Pipeline



Application: Single Cell Mass Cytometry

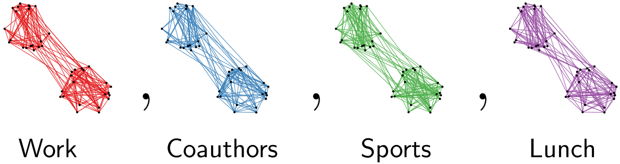


The Power Mean Laplacian for Multilayer Graph Clustering

P. Mercado, A. Gautier, F. Tudisco, M. Hein

Our Goal: Extend spectral clustering to the case where different kind of interactions are present.

$\mathbb{G} = \Big($



$\Big)$

Power Mean Laplacian: $L_p = \left(\frac{1}{T} \sum_{i=1}^{T} \left(L_{\text{sym}}^{(i)} \right)^p \right)^{1/p}$

Spread Sampling for Graphs: Theory and Application

An iterative node sampling method that

- Achieves better community diversity than state-of-the-art
- Has linear time complexity
- Is a better seeding strategy for PPR-based community detection

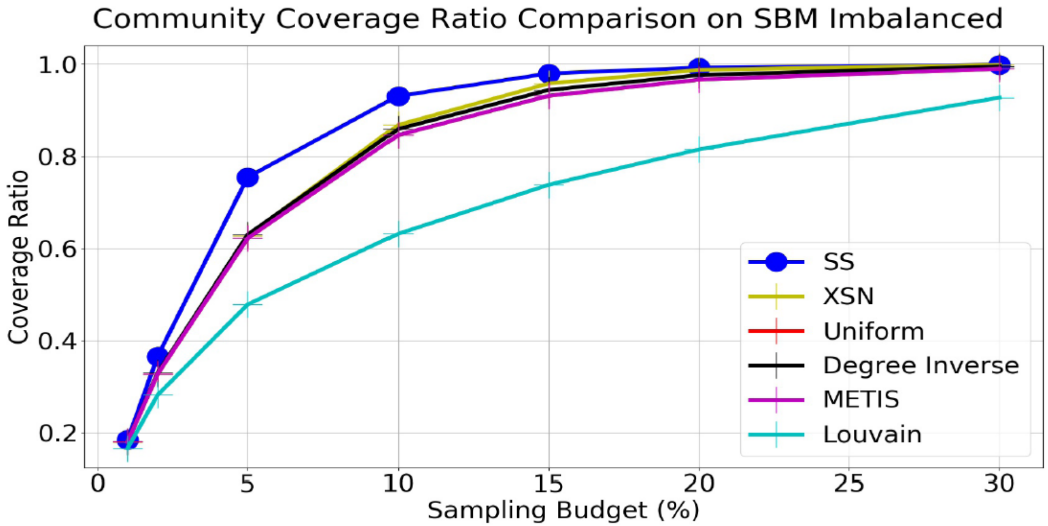


Table: Community Detection Recall with Varying Seed Strategies

Data	Spread Sampling	Uniform	Max Deg.
amazon	**.5768±.038**	.5617±.042	.5612±.043
dblp	**.2512±.004**	.2383±.004	.1479±.001
lj	**.1328±.002**	.1311±.002	.1123±.001
youtube	**.0227±.004**	.0138±.002	.0108±.001

Yu Wang, Bortik Bandyopadhyay, Aniket Chakrabarti,
David Sivakoff and Srinivasan Parthasarathy

Temporal Walk Based Centrality Metric for Graph Streams

Temporal Katz Centrality

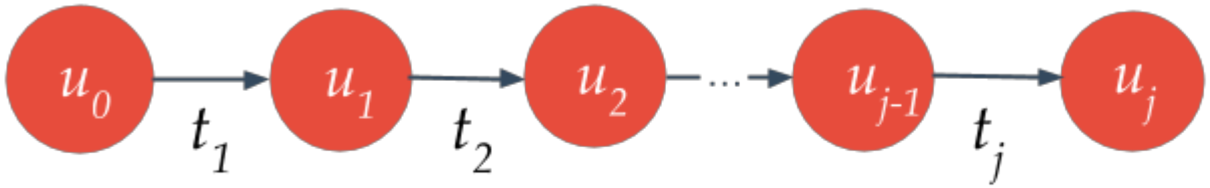
- Centrality measure for dynamic graphs
- Online updateable from the edge stream
- φ is an arbitrary time decay function

$$r_u(t) := \sum_{v} \sum_{\substack{\text{temporal paths } z \\ \text{from } v \text{ to } u}} \Phi(z,t)$$

$$\Phi(z,t) = \varphi(t_2 - t_1) \cdot \ldots \cdot \varphi(t_j - t_{j-1}) \cdot \varphi(t - t_j)$$

Supervised Evaluation

- Roland-Garros, USOpen 2017 Twitter data
- Daily tennis players are considered relevant
- Predict relevant nodes of the mention network with graph centrality



$$t_1 < t_2 < \ldots < t_j \le t$$

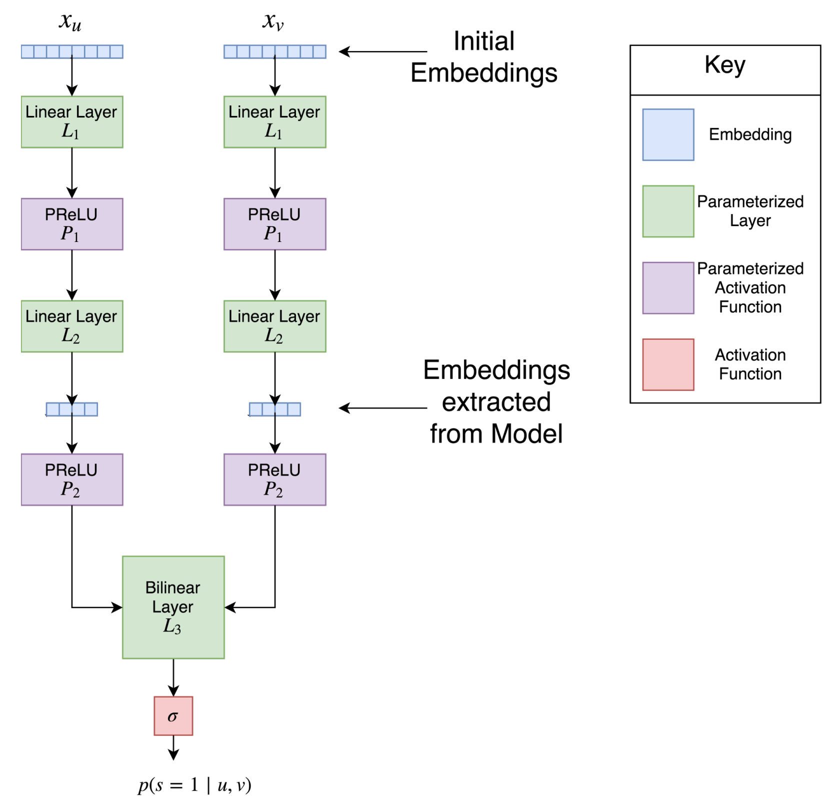
MTA **SZTAKI**
Hungarian Academy of Sciences
Institute for Computer Science and Control

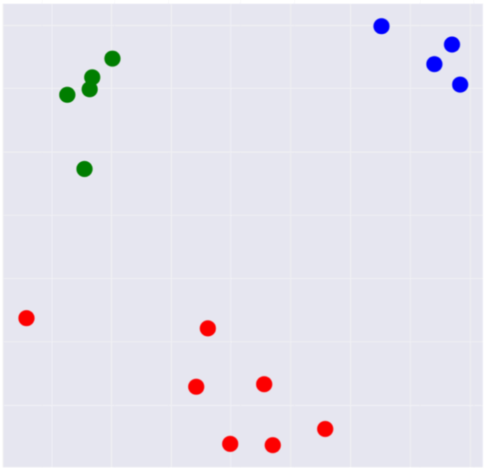
Ferenc Béres, Róbert Pálovics and András A. Benczúr

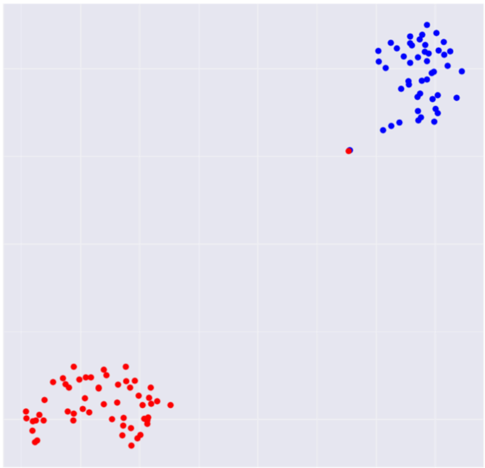
Eötvös Loránd University, Stanford University and the Hungarian Academy of Sciences

A Method for Learning Representations of Signed Networks

- Signed networks comprise +ve and -ve edges.
- Representation learning useful for downstream tasks.
- Methods for unsigned networks don't work well for signed networks.
- Present a method for learning representations using maximum likelihood estimation
- Opposing communities separated in representation space







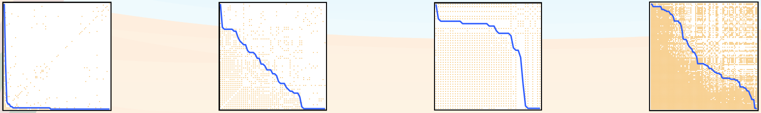
Logistic-Tropical Decompositions and Nested Subgraphs

Sanjar Karaev, Saskia Metzler, and Pauli Miettinen

{skaraev, smetzler, pmiettin}@mpi-inf.mpg.de

Model the problem as thresholded tropical matrix factorization.

Solve using stochastic gradient descent.

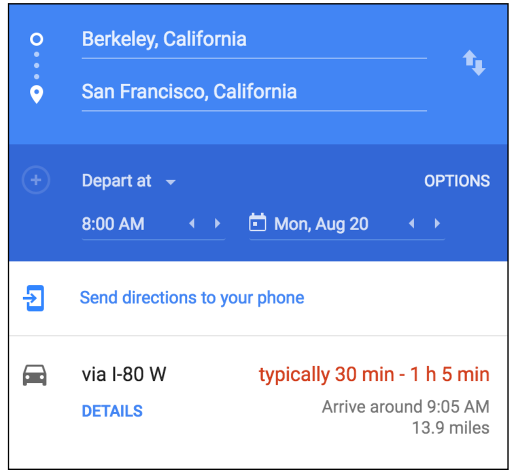


max planck institut informatik

UNIVERSITY OF EASTERN FINLAND

Dynamic Traffic Congestion Prediction Using Graph CNN + LSTM

Traffic prediction at individual level is hard



Can we solve the problem at aggregate level instead?

<u>Key questions:</u>

- How do we represent traffic congestion for a region?
- Which inputs help in predict traffic congestion?
- Can we use the underlying road network graph?
- Can we use prior knowledge of choices made by individuals?
- Can we identify the likely cause of future congestion?

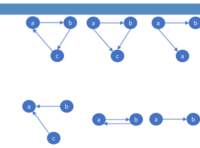
Temporal Graph Generation Based on a Distribution of Temporal Motifs

Sumit Purohit, Lawrence Holder, George Chin

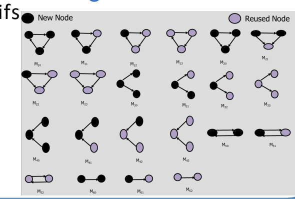
Motivations

- Generate High fidelity synthetic temporal graph
- Privacy Preservation
- Benchmarking

Approach



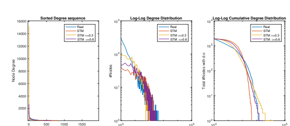
- Non-overlapping temporal motif
- Generate distribution of temporal motifs
  - Up to 3-edges, 3-vertices motifs
  - No self-loop, Non-overlapping
  - Model motif formation time
- Distributed algorithms using:
  - Apache Spark , GraphFrame, Python

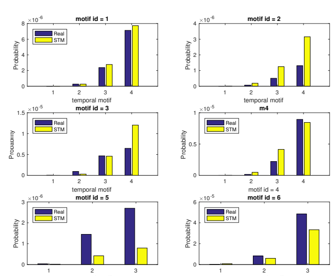


Result

	$\|V\|$	$\|E_{temporal}\|$	$\|E_{static}\|$	Time
CollegeMsg	1899	59,835	20,296	193 days
Bitcoin Alpha	3,783	24,186	24,186	1901 days
PhoneEmail	986	20,001	14,613	365 days

Table 1: Temporal Graphs Properties





Next Step

- Scalability Analysis
- Define Temporal Metrics to measure fidelity
- Deep Autoregressive models to generate graphs
- Code availability
  - Generator code: https://github.com/lbholder/graphstream-generator

Relevance Measurements in Online Signed Social Networks

Tyler Derr[1], Chenxing Wang[1], Suhang Wang[2], and Jiliang Tang[1]

Recently accepted papers on signed network modeling and applications!

Please see my homepage for details!

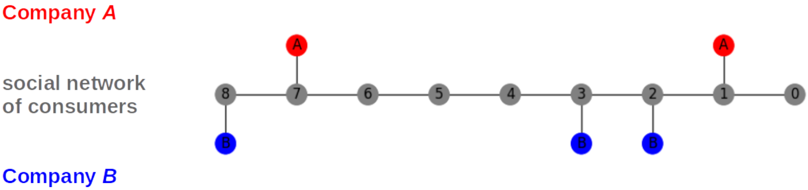
Thank you to the following:

1: Data Science and Engineering Lab, Michigan State University
2: Data Mining and Machine Learning Lab, Arizona State University

A Marketing Game:

a rigorous model for strategic resource allocation

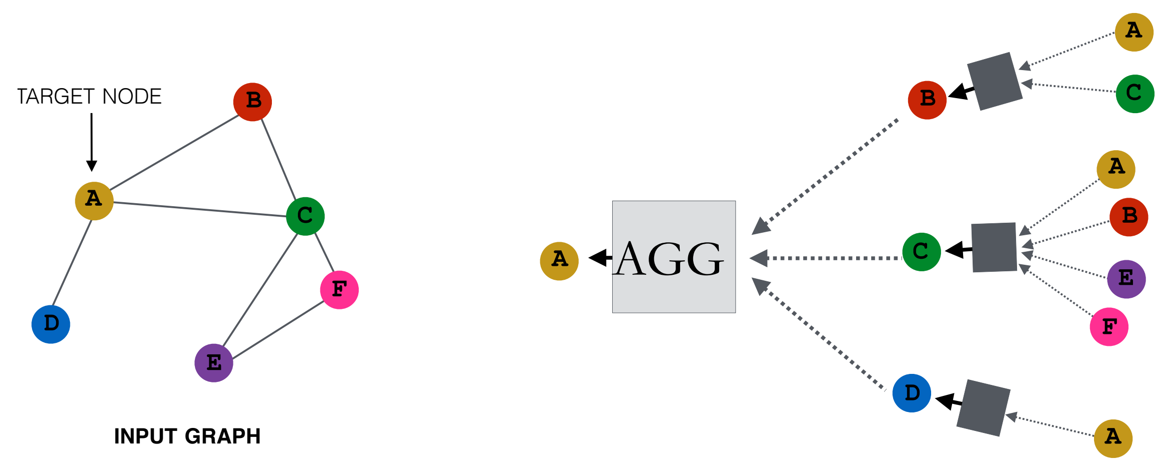
Matthew G. Reyes



Features / Contributions:

- stochastic choice updates rather than best-response
- including marketers in the model
- optimize allocation based on expected market share

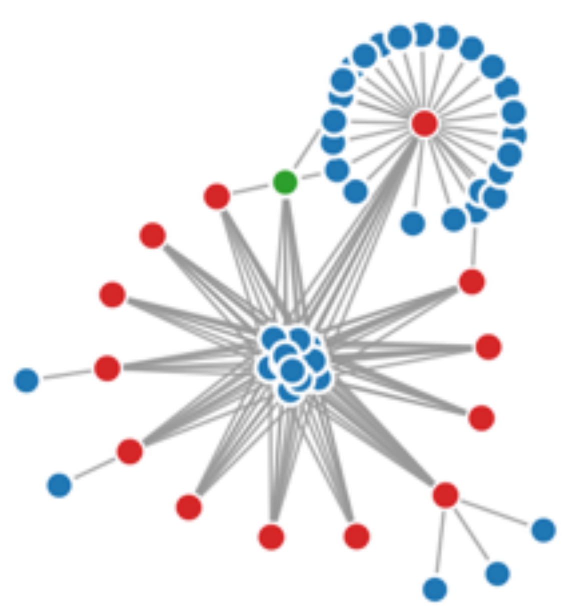
GeniePath adaptively selects "neighbors" to aggregate



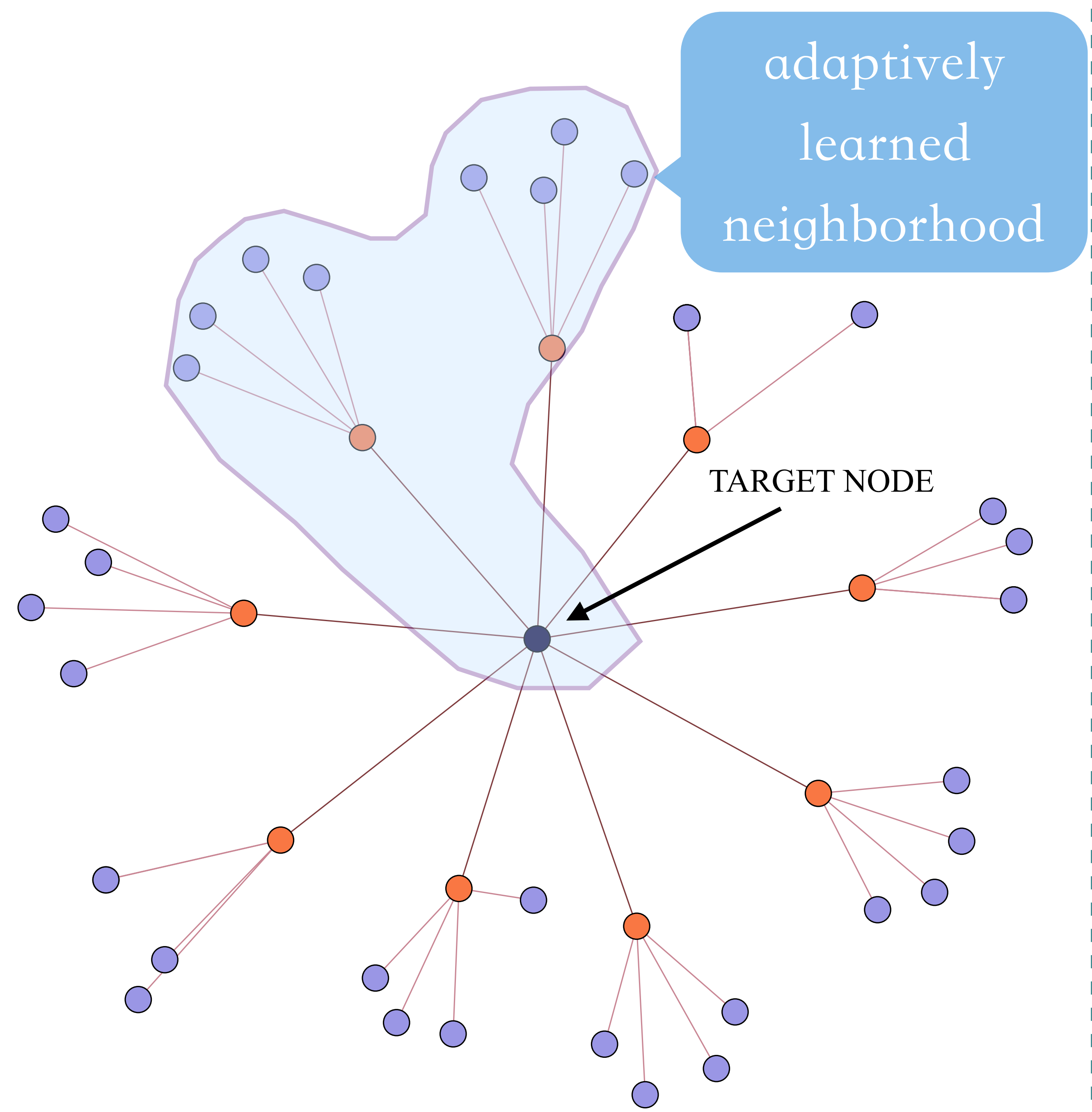
Existing Graph Neural Networks (GNN) study how to aggregate "neighbors" but ignore which "neighbor" to aggregate.

Why GeniePath?

1. Graph is noisy.
2. Different nodes, different roles.
3. Performance & interpretability



GeniePath, learns to explore the breadth and depth of neighborhood.



[1] Liu Z, Chen C, Li L, Zhou J, Li X, Song L. GeniePath: Graph Neural Networks with Adaptive Receptive Paths. arXiv preprint arXiv:1802.00910. 2018 Feb 3.