IEEE SMC 2017

IEEE INTERNATIONAL CONFERENCE ON SYSTEMS,MAN,AND CYBERNETICS (SMC'17), October 5-8, 2017, Banff, Canada

Solving Link-oriented Tasks in Signed Network via an Embedding Approach

Dongfang Du, Hao Wang, Tong Xu*, Yanan Lu,
Qi Liu, Enhong Chen

Anhui Province Key Laboratory of Big Data Analysis and Application
University of Science and Technology of China

Reporter: Dongfang Du

dfdu@mail.ustc.edu.cn

Outline

- **Backgroud**
- Related Work
- Methodology
- Experiments
- Conclusion

Background

- Social network
  - social networks like Facebook, Twitter, WeChat become more and more popular
    - users share feelings, photos and chat with friends
    - mainly based on **positive links**
  - there are still **signed networks** in real life
    - the relationship between nodes is **positive** or **negative**
    - Epinions
      - users **trust** or **distrust** others
    - Slashdot
      - participants are considered **friends** or **enemies**
    - Wikipedia
      - users **approve** or **oppose** the promotion of others

Background

- The applications of signed networks
  - link prediction, recommender system, node classification, etc
    - **negative links** reflect the special value and role
- Existing problems
  - the link relationships in the network are **sparse**
  - the number of positive links is more than negative links
  - the existing models are **limited** by the **sparseness and structures** of the networks
- Our objective
  - make full use of the **network structure** characteristics for **link prediction**

Problem Statement

- Input: given a signed network $G = (U, E^+, E^-)$
  - U : the set of nodes
  - E^+: the set of positive links
  - E^-: the set of negative links
- Output:
  - 1. **low-dimensional vector** $v \in \mathbb{R}^d$ of each node u
  - 2. the **sign** between two nodes u_i and u_j
  - 3. the **TopN** positive links and negative links for every node u

Outline

- Backgroud
- **Related Work**
- Methodology
- Experiments
- Conclusion

Related Work

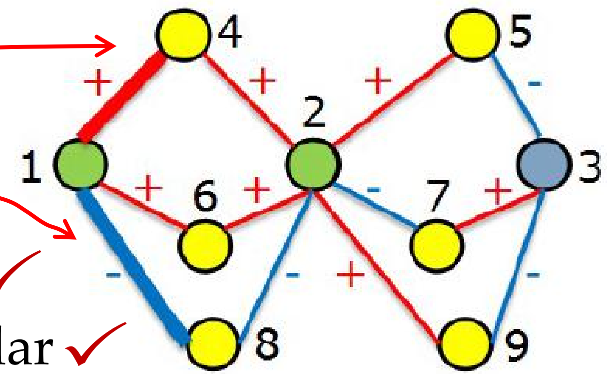
- Unsupervised method
  - based on similarity
    - limited to **network scale**, only for undirected networks
  - based on matrix decomposition
    - limited to **feature dimensions**
- Supervised method
  - extract the link feature, and train the classifier
    - did not well capture the network structure
- Network embedding method
  - mapping nodes to **low-dimensional vectors** which is superior to existing technologies
  - **rarely** used in signed network

Outline

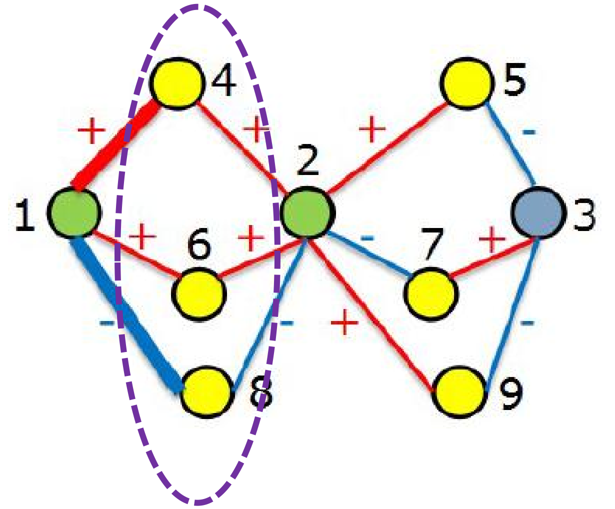
- Backgroud
- Related Work
- **Methodology**
- Experiments
- Conclusion

Data Analysis

- First-order Distance
  - the direct distance between two nodes
    - **verification**: positive --> nodes are similar ✓
    - **verification**: negative --> nodes are dissimilar ✓



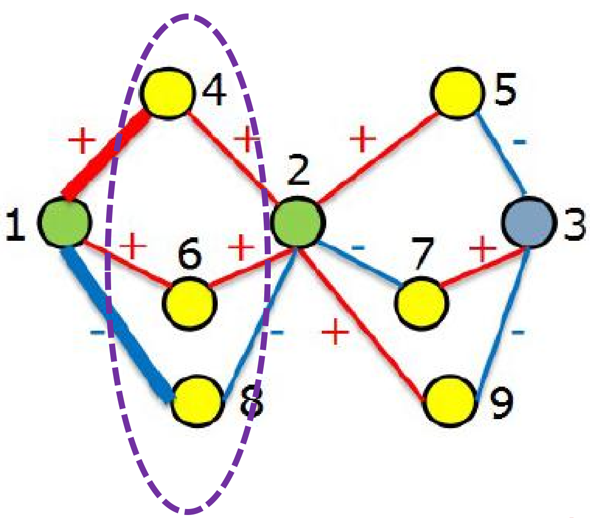
- Second-order Distance
  - the **similarity between the neighbors** of the two nodes
    - **verification**: neighbor structures are similar --> node are similar **?**
    - **verification**: neighbor structures are dissimilar --> node are dissimilar **?**



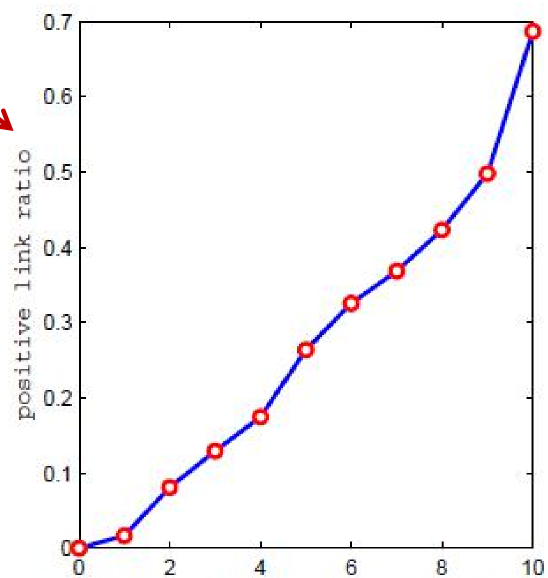
Data Analysis

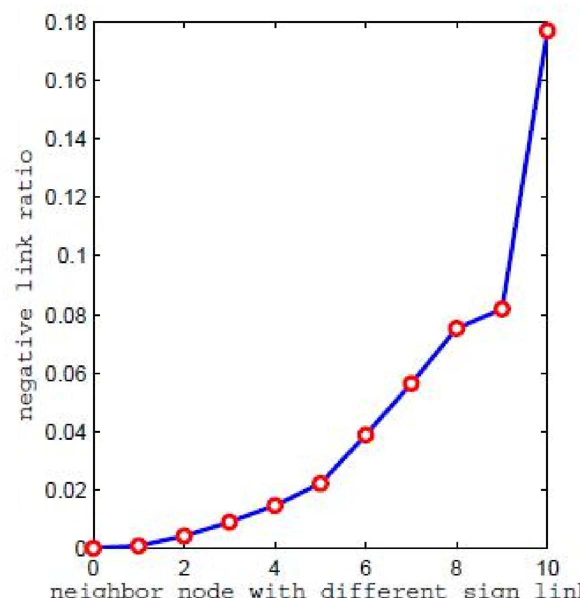
- Verification
  - neighbor structures are similar --> node are similar
  - neighbor structures are dissimilar --> node are dissimilar

dataset	#node	#link	#positive link	#negative link
Slashdot	77,357	516,575	369,378	120,197



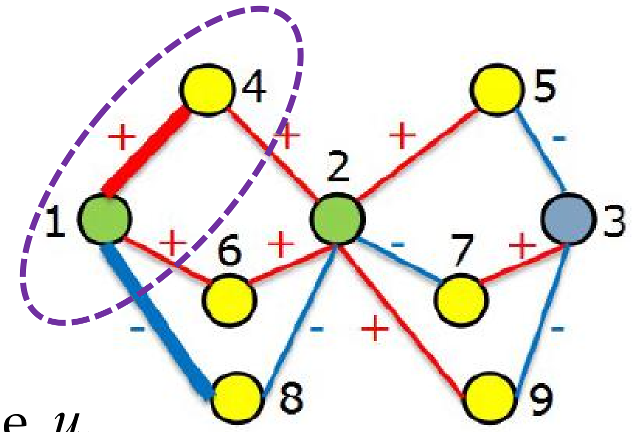
positive link ratio





neighbor nodes with the same sign link

First-order Distance Method



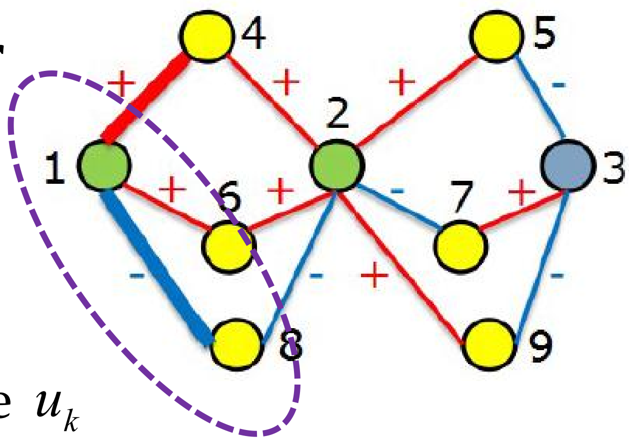
- Positive link --> nodes are similar
  - **joint probability** of positive link (u_i, u_j)

    $$p_1(u_i, u_j) = \frac{1}{1 + exp(-\vec{v}_i^T \cdot \vec{v}_j)}$$

    - where $v_i \in \mathbb{R}^d$ is the representation of node u_i
  - **empirical distribution** for (u_i, u_j) is $\hat{p}_1(u_i, u_j) = w_{ij} / W^+$
    - where w_{ij} is the weight of (u_i, u_j)
    - $W^+ = \sum_{(i,j)\in E^+} w_{ij}$
  - measure the difference between two distributions by **KL divergence** $O_1^+ = KL(\hat{p}_1(\cdot,\cdot), p_1(\cdot,\cdot))$
  - discard the irrelevant constants

    $$O_1^+ = -\sum_{(i,j)\in E^+} w_{ij} log p_1(u_i, u_j)$$

First-order Distance Method



- Negative link --> nodes are dissimilar
  - **joint probability** of negative link (u_k, u_s)

$$p_2(u_k, u_s) = 1 - p_1(u_k, u_s) = \frac{1}{1 + exp(\vec{v}_k^T \cdot \vec{v}_s)}$$

    - where $v_k \in \mathbb{R}^d$ is the representation of node u_k
  - **empirical distribution** for (u_k, u_s) is $\hat{p}_2(u_k, u_s) = w_{ks} / W^-$
    - where w_{ks} is the weight of (u_k, u_s)
    - $W^- = \sum_{(k,s)\in E^-} w_{ks}$
  - also measure the difference between two distributions by **KL divergence**
  - discard the irrelevant constants

$$O_1^- = - \sum_{(k,s)\in E^-} w_{ks} log p_2(u_k, u_s)$$

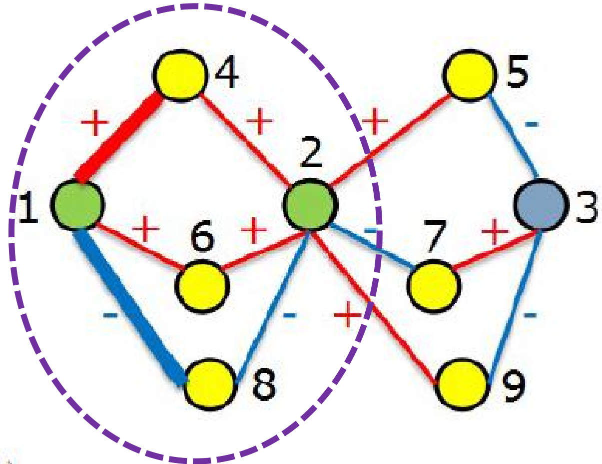
First-order Distance Method

- Objective function

$$O_1 = -\left(\sum_{(i,j)\in E^+} w_{ij} log p_1(u_i, u_j) + \sum_{(k,s)\in E^-} w_{ks} log p_2(u_k, u_s) \right)$$

Second-order Distance Method

- neighbor structures are similar
  --> node are similar



- movited by input and output matrices
  - positive link (u_i, u_j) is built from u_i to u_j
    then the conditional probability is

$$p_3(u_j|u_i) = \frac{exp((\vec{v}_j^{in})^T \cdot \vec{v}_i^{out})}{\sum_{t=1}^{|U|} exp((\vec{v}_t^{in})^T \cdot \vec{v}_t^{out})}$$

- **empirical distribution** for (u_i, u_j) is $\hat{p}_3(u_j | u_i) = w_{ij} / d_i^+$
- also use **KL divergence :** $O_2^+ = \sum_{i \in V} \lambda_i KL(\hat{p}_3(\cdot|v_i), p_3(\cdot|v_i))$

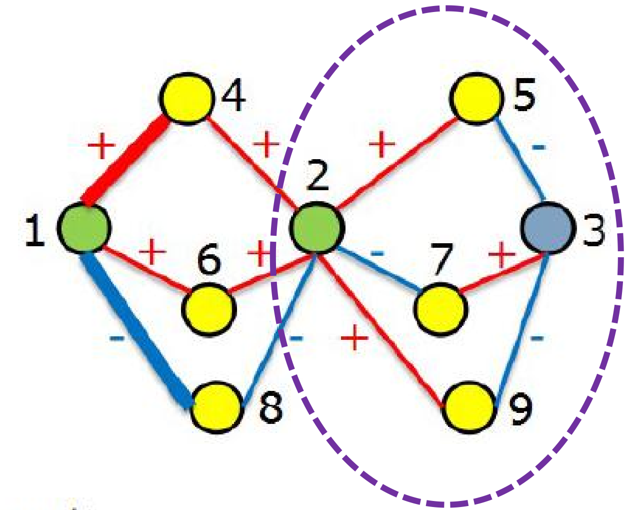
$\lambda_i = d_i^+$

- discard the irrelevant constants

$$O_2^+ = - \sum_{(i,j) \in E^+} w_{ij} log p_3(v_j|v_i)$$

Second-order Distance Method

- neighbor structures are dissimilar
  --> node are dissimilar
  - movited by input and output matrices
    - negative link (u_k, u_s) is built from u_k to u_s
      then the conditional probability is



$$p_4(u_s|u_k) = \frac{exp(-(\vec{v}_s^{in})^T \cdot \vec{v}_k^{out})}{\sum_{t=1}^{|U|} exp(-(\vec{v}_t^{in})^T \cdot \vec{v}_t^{out})}$$

- **empirical distribution** for (u_k, u_s) is $\hat{p}_4(u_k, u_s) = w_{ks} / d_k^-$
- also use **KL divergence** to measure the difference
- discard the irrelevant constants

$$O_2^- = -\sum_{(k,s)\in E^-} w_{ks} log p_4(u_s|u_k)$$

Second-order Distance Method

- Objective function

$$O_2 = -\left(\sum_{(i,j)\in E^+} w_{ij} log p_3(u_j|u_i) + \sum_{(k,s)\in E^-} w_{ks} log p_4(u_s|u_k)\right)$$

Model Optimization

- The calculations of the above model need a high time cost
  - motivated by **negative sampling**
    - for positive link (u_i, u_j) built from u_i to u_j

$$N \cdot log\sigma((\vec{v}_j^{in})^T \cdot \vec{v}_i^{out}) + \Sigma_{n=1}^{K} E_{u_n \sim P_n(u)}[log\sigma(-(\vec{v}_n^{in})^T \cdot \vec{v}_i^{out})]$$

negative samples

    - for negative link (u_k, u_s) built from u_k to u_s

$$N \cdot log\sigma(-(\vec{v}_s^{in})^T \cdot \vec{v}_k^{out}) + \Sigma_{n=1}^{K} E_{u_n \sim P_n(u)}[log\sigma((\vec{v}_n^{in})^T \cdot \vec{v}_k^{out})]$$

negative samples

Outline

- Backgroud
- Related Work
- Methodology
- **Experiments**
- Conclusion

Experiments Settings

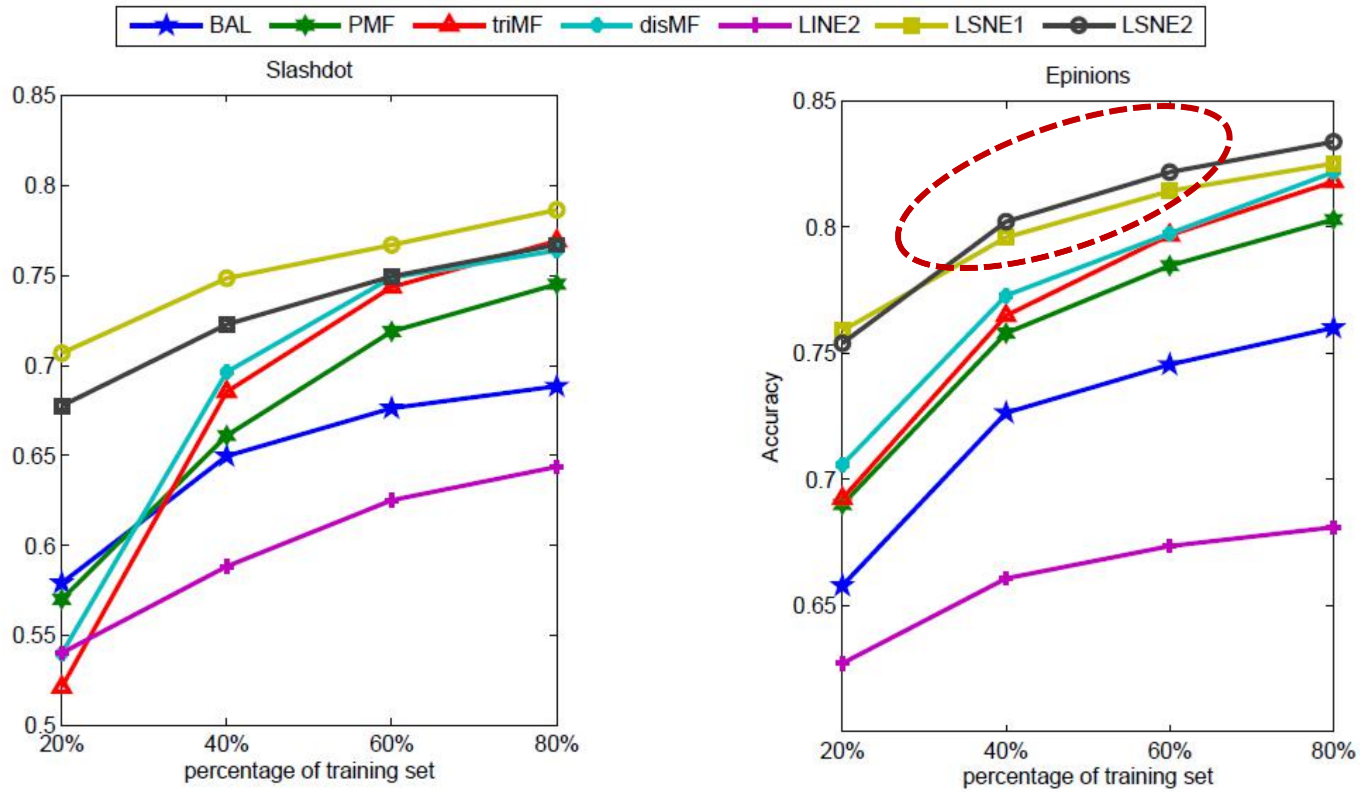
- Datasets

dataset	#node	#link	#positive link	#negative link
Slashdot	77,357	516,575	369,378	120,197
Epinions	131,828	841,372	717,667	123,705

- Baselines
  - BAL: model balance theory
  - PMF: probabilistic matrix factorization
  - triMF: extend PMF to handle directed links
  - disMF: combine BAL with triMF
  - LINE2: extend LINE
- Our methods
  - LSNE1 and LSNE2

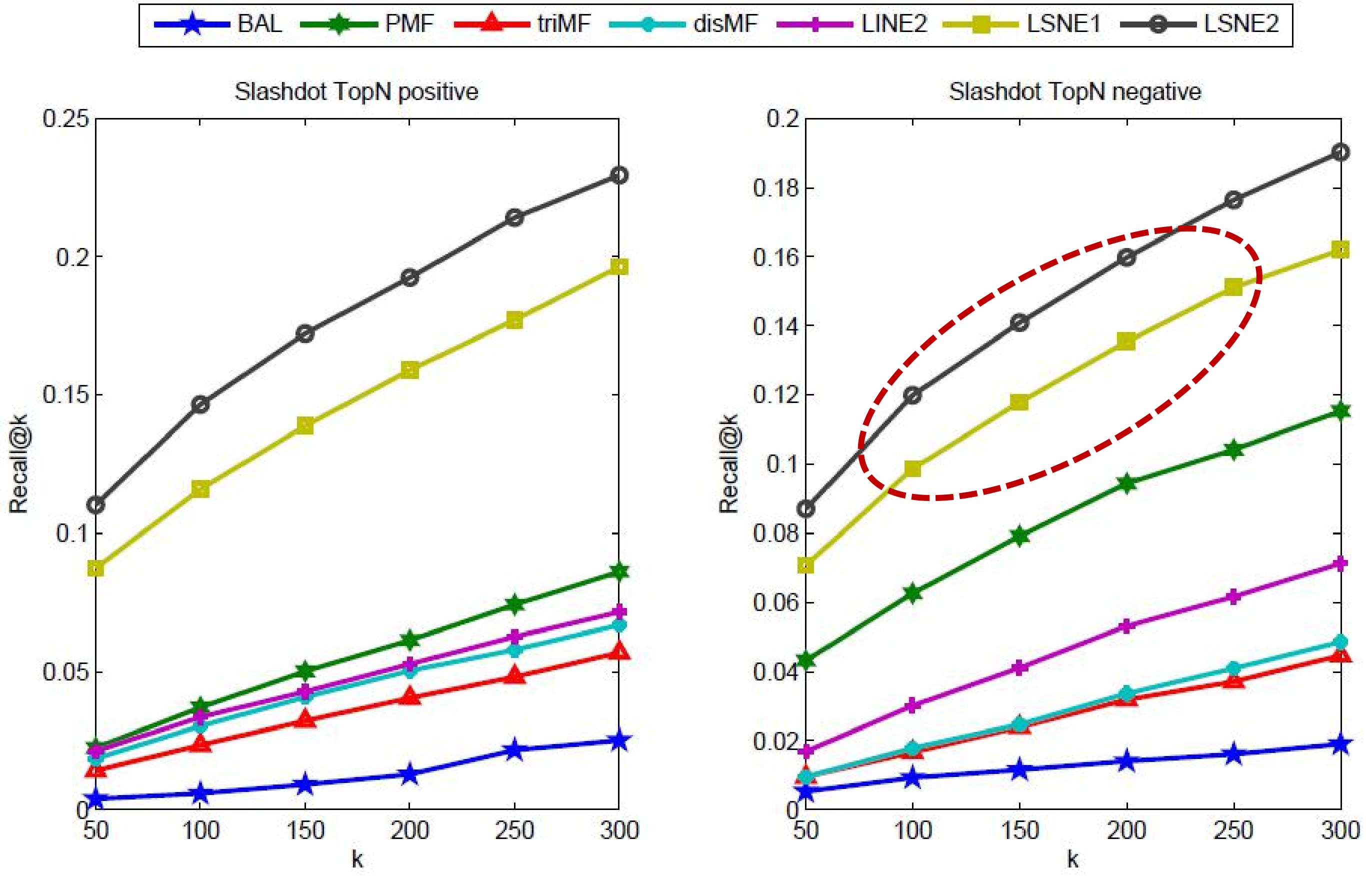
Sign Prediction Tasks

- Target 1:
  - select x% of the links for training, predict the signs of the remaining (100-x)% of links



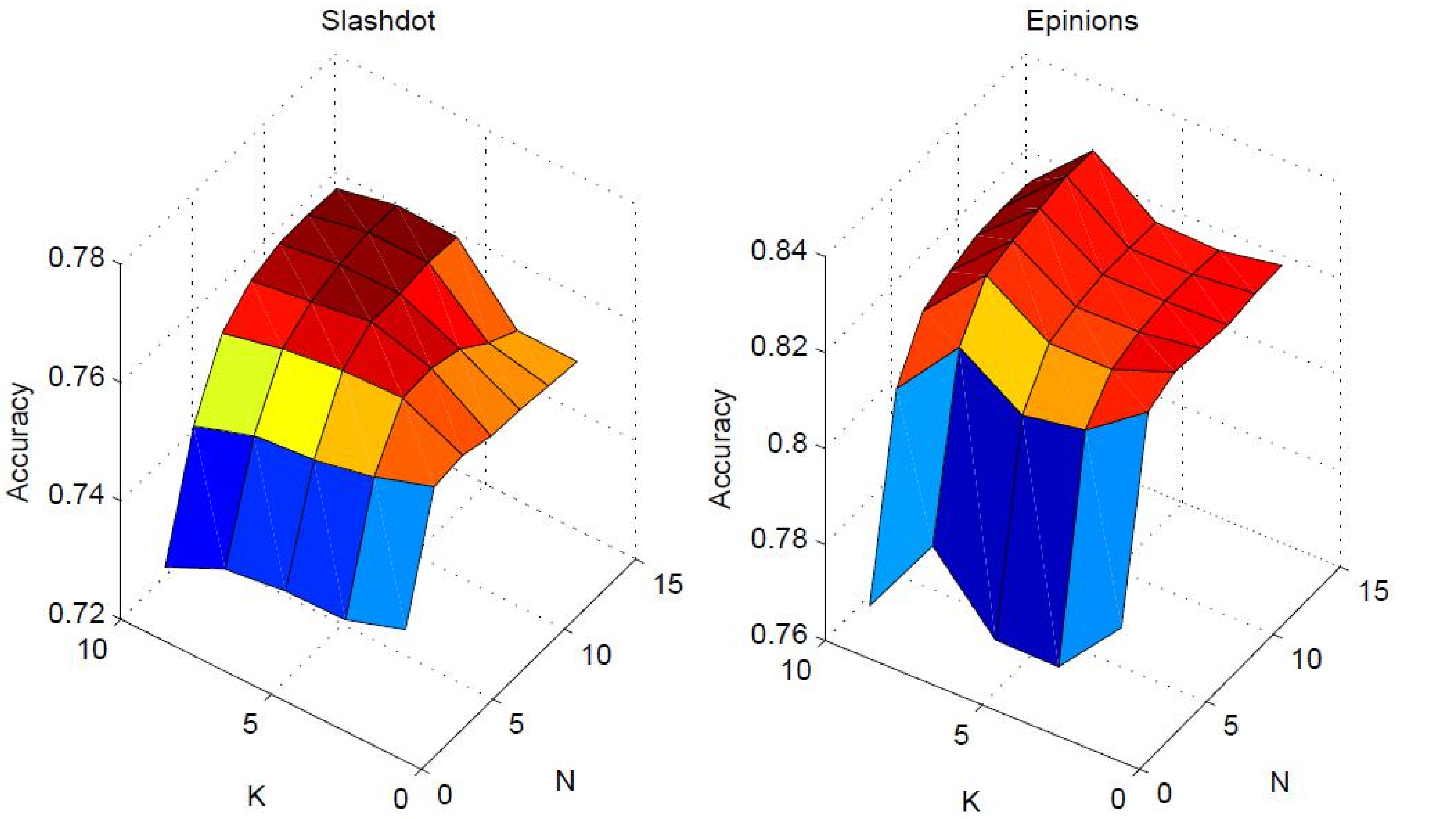
Link Prediction Tasks

- Target 2:
  - predict the TopN positive and negative links



Parameters sensitivity

- K: #negative samples; N: #positive samples



Outline

- Backgroud
- Related Work
- Methodology
- Experiments
- **Conclusion**

Conclusion

- **Problem**: learn **representations for** each node in signed networks and furture solve link-oriented tasks
- **Method**: LSNE1, LSNE2 based on alternating optimization
- **Contributions**:
  - demonstrate the effect of **second-order** information in signed network
  - propose a novel **framework** called LSNE
  - extensive experiments verifies that LSNE is capable of capturing the **intrinsic structural regularities** in signed network.

Q & A

Thanks !

Reporter: Dongfang Du
dfdu@mail.ustc.edu.cn