

Predicting Link Directions via a Recursive Subgraph-based Ranking

(Oh no, yet another ranking...)

Fangjian Guo

Web Sciences Center

June 21st, 2012

Link prediction

Task

- Estimating the probability of links that are missing or unobserved in our data

Link prediction

Task

- Estimating the probability of links that are missing or unobserved in our data

Evaluation

Link prediction

Task

- Estimating the probability of links that are missing or unobserved in our data

Evaluation

- *Hide & Check*

Link prediction

Task

- Estimating the probability of links that are missing or unobserved in our data

Evaluation

- *Hide & Check*

Studied mainly for undirected networks

Link prediction

Task

- Estimating the probability of links that are missing or unobserved in our data

Evaluation

- *Hide & Check*

Studied mainly for undirected networks

Methods

Link prediction

Task

- Estimating the probability of links that are missing or unobserved in our data

Evaluation

- *Hide & Check*

Studied mainly for undirected networks

Methods

- Similarity-based index

Link prediction

Task

— Estimating the probability of links that are missing or unobserved in our data

Evaluation

— *Hide & Check*

Studied mainly for undirected networks

Methods

- Similarity-based index
- Fitting a statistical model

Link prediction

Task

— Estimating the probability of links that are missing or unobserved in our data

Evaluation

— *Hide & Check*

Studied mainly for undirected networks

Methods

- Similarity-based index
- Fitting a statistical model
- ...

Link prediction

Task

— Estimating the probability of links that are missing or unobserved in our data

Evaluation

— *Hide & Check*

Studied mainly for undirected networks

Methods

- Similarity-based index
- Fitting a statistical model
- ...

Predicting link = predicting *existence*

Link prediction for directed networks

Link prediction for directed networks

But how to predict links for directed networks?

Link prediction for directed networks

But how to predict links for directed networks?

Generally, two schemes:

Link prediction for directed networks

But how to predict links for directed networks?

Generally, two schemes:

- *Existence & direction* simultaneously

Link prediction for directed networks

But how to predict links for directed networks?

Generally, two schemes:

- *Existence & direction* simultaneously
- 1. *Existence*

Link prediction for directed networks

But how to predict links for directed networks?

Generally, two schemes:

- *Existence & direction* simultaneously
- 1. *Existence* \rightarrow 2. *direction* if exist

Link prediction for directed networks

But how to predict links for directed networks?

Generally, two schemes:

- *Existence & direction* simultaneously
- 1. *Existence* \rightarrow 2. *direction* if exist

Methods for the first scheme

Link prediction for directed networks

But how to predict links for directed networks?

Generally, two schemes:

- *Existence & direction* simultaneously
- 1. *Existence* \rightarrow 2. *direction* if exist

Methods for the first scheme

- Fitting a statistical graph model

Link prediction for directed networks

But how to predict links for directed networks?

Generally, two schemes:

- *Existence & direction* simultaneously
- 1. *Existence* \rightarrow 2. *direction* if exist

Methods for the first scheme

- Fitting a statistical graph model

Clauset et al. *Hierarchical structure and the prediction of missing links in networks*, *Nature* (2008)

Link prediction for directed networks

But how to predict links for directed networks?

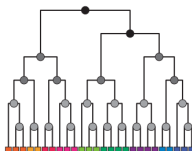
Generally, two schemes:

- *Existence & direction* simultaneously
- 1. *Existence* \rightarrow 2. *direction* if exist

Methods for the first scheme

- Fitting a statistical graph model

Clauset et al. *Hierarchical structure and the prediction of missing links in networks*, *Nature* (2008)



Link prediction for directed networks

But how to predict links for directed networks?

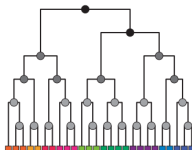
Generally, two schemes:

- *Existence & direction* simultaneously
- 1. *Existence* \rightarrow 2. *direction* if exist

Methods for the first scheme

- Fitting a statistical graph model

Clauset et al. *Hierarchical structure and the prediction of missing links in networks*, *Nature* (2008)



- Matching local motifs

Link prediction for directed networks

But how to predict links for directed networks?

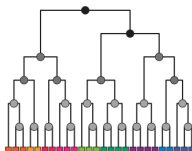
Generally, two schemes:

- *Existence & direction* simultaneously
- 1. *Existence* \rightarrow 2. *direction* if exist

Methods for the first scheme

- Fitting a statistical graph model

Clauset et al. *Hierarchical structure and the prediction of missing links in networks*, *Nature* (2008)



- Matching local motifs
Qianming Zhang, *unpublished*

Direction and ranking

However, we choose the second scheme

Direction and ranking

However, we choose the second scheme

Suppose we know $(i_1, j_1), (i_2, j_2), \dots, (i_n, j_n)$ are connected pairs, how to predict their directions?

Direction and ranking

However, we choose the second scheme

Suppose we know $(i_1, j_1), (i_2, j_2), \dots, (i_n, j_n)$ are connected pairs, how to predict their directions?

Key assumption

Direction and ranking

However, we choose the second scheme

Suppose we know $(i_1, j_1), (i_2, j_2), \dots, (i_n, j_n)$ are connected pairs, how to predict their directions?

Key assumption

The formation of a directed network is regulated by *an implicit ranking of nodes!*

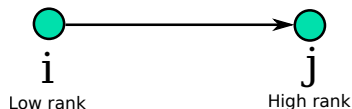
Direction and ranking

However, we choose the second scheme

Suppose we know $(i_1, j_1), (i_2, j_2), \dots, (i_n, j_n)$ are connected pairs, how to predict their directions?

Key assumption

The formation of a directed network is regulated by *an implicit ranking of nodes!*



Direction and ranking

Evidence

Direction and ranking

Evidence

- Comparatively only a few links are reciprocal.

Direction and ranking

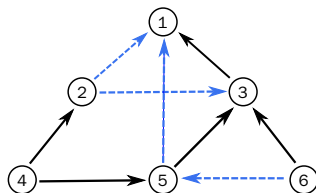
Evidence

- Comparatively only a few links are reciprocal.
- Constructing maximum *acyclic* graph from directed networks

Direction and ranking

Evidence

- Comparatively only a few links are reciprocal.
- Constructing maximum *acyclic* graph from directed networks



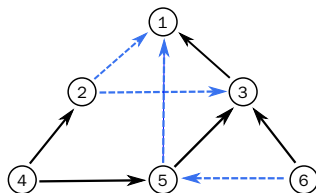
R_1 : 1 2 3 6 5 4

R_2 : 1 3 2 5 6 4

Direction and ranking

Evidence

- Comparatively only a few links are reciprocal.
- Constructing maximum *acyclic* graph from directed networks



R_1 : 1 2 3 6 5 4

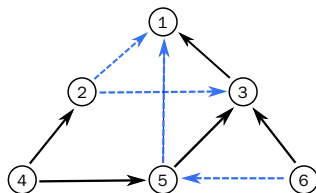
R_2 : 1 3 2 5 6 4

- Even *M. E. J. Newman* thinks it so.

Direction and ranking

Evidence

- Comparatively only a few links are reciprocal.
- Constructing maximum *acyclic* graph from directed networks



R_1 : 1 2 3 6 5 4

R_2 : 1 3 2 5 6 4

- Even *M. E. J. Newman* thinks it so.
Friendship networks and social status, Brian Ball and M. E. J. Newman, submitted.

Our method

Two steps

Our method

Two steps

- 1 Rank all nodes in the network G/E_c (E_c refers to the links to be predicted)

Our method

Two steps

- 1 Rank all nodes in the network G/E_c (E_c refers to the links to be predicted)
- 2 Predict links in E_c by the rule:

Our method

Two steps

- 1 Rank all nodes in the network G/E_c (E_c refers to the links to be predicted)
- 2 Predict links in E_c by the rule: $i \rightarrow j$, if $R(i) > R(j)$

Our method

Two steps

- 1 Rank all nodes in the network G/E_c (E_c refers to the links to be predicted)
- 2 Predict links in E_c by the rule: $i \rightarrow j$, if $R(i) > R(j)$

Evaluation

Our method

Two steps

- 1 Rank all nodes in the network G/E_c (E_c refers to the links to be predicted)
- 2 Predict links in E_c by the rule: $i \rightarrow j$, if $R(i) > R(j)$

Evaluation

Conformity

$$C = \frac{\|\{(i, j) \in E_c \mid R(i) > R(j), i \rightarrow j\}\|}{\|E_c\|}$$

How to rank?

How to rank?

Two ingredients

How to rank?

Two ingredients

- 1 Local indicator

How to rank?

Two ingredients

① Local indicator

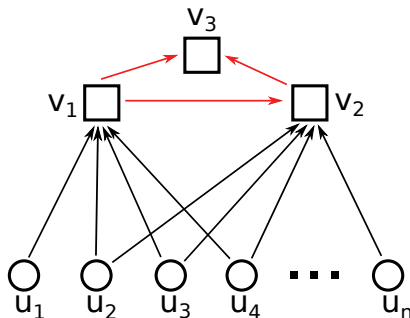
Degree difference $D^{\Delta} = D^{in} - D^{out}$

How to rank?

Two ingredients

① Local indicator

Degree difference $D^\Delta = D^{in} - D^{out}$



How to rank?

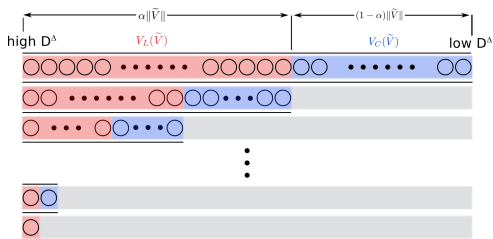
① Local indicator

How to rank?

- ① Local indicator
- ② Hierarchical organization via subgraph extraction

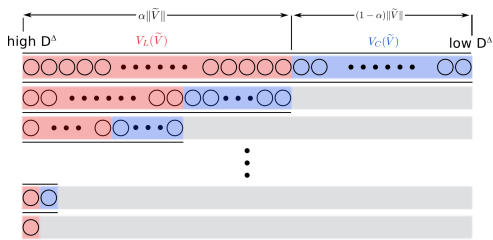
How to rank?

- ① Local indicator
- ② Hierarchical organization via subgraph extraction



How to rank?

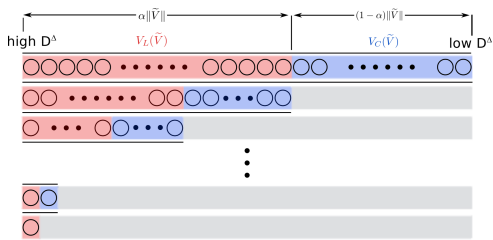
- 1 Local indicator
- 2 Hierarchical organization via subgraph extraction



$$\text{Leaders } V_L(\tilde{V}) = \{j \in \tilde{V} \mid I(j; \tilde{V}) < \alpha \|\tilde{V}\|\}$$

How to rank?

- ① Local indicator
- ② Hierarchical organization via subgraph extraction

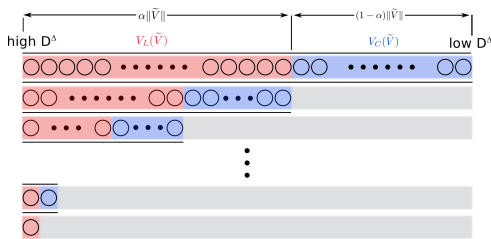


Leaders $V_L(\tilde{V}) = \{j \in \tilde{V} | I(j; \tilde{V}) < \alpha \|\tilde{V}\|\}$

Followers $V_C(\tilde{V}) = \{j \in \tilde{V} | I(j; \tilde{V}) \geq \alpha \|\tilde{V}\|\}$

How to rank?

- 1 Local indicator
- 2 Hierarchical organization via subgraph extraction



Leaders $V_L(\tilde{V}) = \{j \in \tilde{V} | I(j; \tilde{V}) < \alpha \|\tilde{V}\|\}$

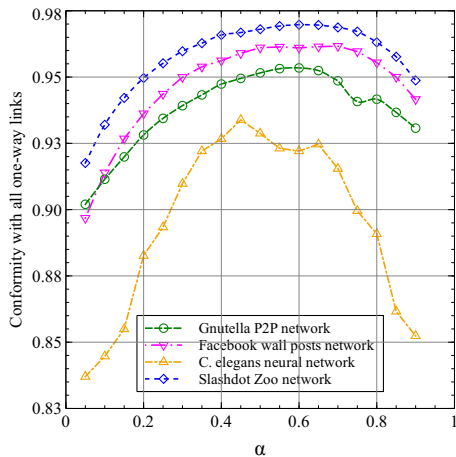
Followers $V_C(\tilde{V}) = \{j \in \tilde{V} | I(j; \tilde{V}) \geq \alpha \|\tilde{V}\|\}$

Recursive ranking

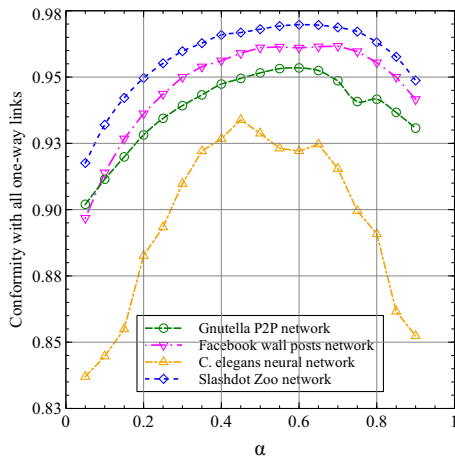
$$R(i; \tilde{V}) = \begin{cases} 1 & \|\tilde{V}\| = 1 \\ R(i; V_L(\tilde{V})) & \|\tilde{V}\| \geq 1, i \in V_L(\tilde{V}) \\ \|V_L(\tilde{V})\| + R(i; V_C(\tilde{V})) & \|\tilde{V}\| \geq 1, i \notin V_L(\tilde{V}), \end{cases}$$

Selecting α

Selecting α



Selecting α

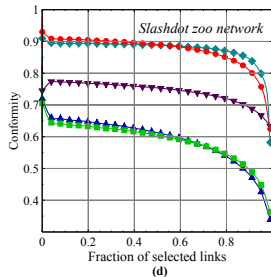
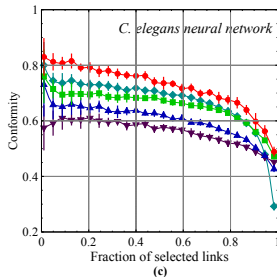
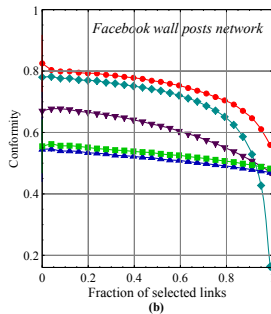
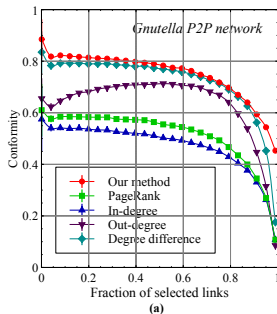


We thus choose

$$\alpha = 0.6$$

Performance

Performance



To be improved...

To be improved...

- ① Predicting reciprocal links?

To be improved...

- ① Predicting reciprocal links?
- ② A more solid rationale?

To be improved...

- ① Predicting reciprocal links?
- ② A more solid rationale?
- ③ Link breaking problem?

To be improved...

- ① Predicting reciprocal links?
- ② A more solid rationale?
- ③ Link breaking problem?

Thank you!

(Our paper can be downloaded at *arXiv:1206.2199*)