AURORA: Auditing PageRank on Large Graphs

Presented By Jian Kang

Jian Kang  Meijia Wang  Nan Cao  Yinglong Xia  Wei Fan  Hanghang Tong
Ranking on Graphs: PageRank

- Webpages are no longer independent
- Rank the webpages by their importance/relevance
More Applications

- Recomender System [Gori’07]
- Social Network Analysis [Weng’10]
- Sports Team Management [Radicchi’11]
- Biology [Singh’07]
PageRank: Formulation

**Assumption:**
- A webpage is important if it is linked by many other webpages

**Formulation:**
- Iteratively solve the following linear system
  \[ r = cA r + (1 - c)e \]
  - Mathematically elegant, only topological information is needed

**Many Variants Exist:**
- Personalized PageRank
- Random Walk with Restart
- And so on
Why Auditing PageRank?

- **Problem:** end-users do not understand how the results were derived

- **Potential Outcomes:**
  - Render crucial explainability of ranking algorithms
  - Optimize network topology
  - Identify vulnerabilities in the network (e.g. preventing adversarial attacks)
Roadmap

- Motivations
- AURORA Formulation
- AURORA Algorithms
- AURORA Generalizations
- Experimental Results
- Conclusions
Prob. Def.: PageRank Auditing Problem

Given:

- (1) adjacency matrix $A$;
- (2) PageRank $r$;
- (3) loss function over PageRank vector $f(r)$;
- (4) user-specific element type (edges vs. nodes vs. subgraph);
- (5) integer budget $k$.

Find: a set of $k$ influential graph elements

Intuitive Example:
AURORA Formulation

- **Intuition:** find a set of influential elements that have largest impact on the loss function over PageRank vector.

- **Optimization Problem:**
  \[
  \max_S \Delta f = \left( f(r) - f(r_S) \right)^2
  \]
  \( s.t. \quad |S| = k \)

- **Choices of Loss Function:**

  - **Square**

    **TABLE II:** Choices of \( f(\cdot) \) functions and their derivatives

    | Descriptions   | Functions                      | Derivatives                      |
    |----------------|--------------------------------|---------------------------------|
    | \( L_p \) norm | \( f(r) = ||r||_p \)          | \( \frac{\partial f}{\partial r} = \frac{r_0||r||_p^{p-2}}{||r||_p^{p-1}} \) |
    | Soft maximum   | \( f(r) = \log(\sum_{i=1}^{n} \exp(r(i))) \) | \( \frac{\partial f}{\partial r} = \left[ \frac{\exp(r(i))}{\sum_{i=1}^{n} \exp(r(i))} \right] \) |
    | Energy norm    | \( f(r) = r'Mr \)            | \( \frac{\partial f}{\partial r} = (M + M')r \) |

    (\( M \) in Energy Norm is a Hermitian positive definite matrix.)
Challenges

- C1: Measure of Influence
- C2: Optimality
- C3: Scalability
Challenges

**C1: Measure of Influence**

- Understanding Black-box Machine Learning Models
  - Quantify influence by perturbing features or training data.
  - **Obs:** Inconsistent with unsupervised graph ranking settings.

- Influence Maximization
  - Measure the size of ‘infected’ nodes in information propagation process.
  - **Obs:** fundamentally different from finding influential elements in graph ranking settings.

- **Question:** how to define the influence in the context of graph ranking?

---

[1] Adler, P., Falk, C., Friedler, S. A., Nix, T., Rybeck, G., Scheidegger, C., Smith, B., & Venkatasubramanian, S. (2018). Auditing black-box models for indirect influence. *Knowledge and Information Systems, 54*(1), 95-122.

[2] Koh, P. W., & Liang, P. (2017, July). Understanding Black-box Predictions via Influence Functions. In *International Conference on Machine Learning* (pp. 1885-1894).

[3] Kempe, D., Kleinberg, J., & Tardos, É. (2003, August). Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 137-146). ACM.
Challenges

- **C2: Optimality**
  - Finding a set of influential graph elements is NP due to its combinatorial nature.
  - **Question**: how to find a set of influential graph elements accurately?

- **C3: Scalability**
  - **Question**: how to scale up the influential elements finding process?
Definition: Graph Element Influence

- **Graph Element Influence**
  - The influence of an edge \((i, j)\) is defined as the derivative of \(f(\mathbf{r})\) w.r.t. the edge.
    \[
    II(i, j) = \frac{df(\mathbf{r})}{dA(i, j)}
    \]
  - The influence of a node \(i\) is defined as the aggregation of all in and out edges.
    \[
    II(i) = \sum_{j=1, j \neq i}^{n} II(i, j) + II(j, i)
    \]
  - The influence of a subgraph \(S\) is defined as the aggregation of all edges in the subgraph.
    \[
    II(i) = \sum_{i,j \in S}^{n} II(i, j)
    \]
Calculating Influence

- **Method:**
  - Define $Q = (I - cA)^{-1}$, PageRank: $r = (1 - c)Qe$
  - Apply chain rule
    $$\frac{\partial f(r)}{\partial A(i,j)} = \text{Tr}[(\frac{\partial f(r)}{\partial r})' \frac{\partial r}{\partial A(i,j)}] = 2cr(j)\text{Tr}[r'Q(:,i)]$$

- **Matrix Form Solution:**
  $$\frac{df(r)}{dA} = \begin{cases} \frac{\partial f(r)}{\partial A} + (\frac{\partial f(r)}{\partial A})' - \text{diag} \left( \frac{\partial f(r)}{\partial A} \right), & \text{if } A \text{ is undirected graph} \\ \frac{\partial f(r)}{\partial A}, & \text{if } A \text{ is directed graph} \end{cases}$$

where $\frac{\partial f(r)}{\partial A} = 2cQ'rr'$, each element in $\frac{\partial f(r)}{\partial A}$ is $\frac{\partial f(r)}{\partial A(i,j)}$

- **Limitation:** $Q'rr'$ is an $n \times n$ full matrix, need $O(n^2)$ space
- **Question:** how to scale up to large graphs?
Scale Up

Solution: exploring low-rank structure

- Note that PageRank $r = (1 - c)Qe$

$$\frac{\partial f(r)}{\partial A} = 2cQ'r'r'$$

- Reduce $O(n^2)$ space to $O(n)$ space
Roadmap

- Motivations
- AURORA Formulation
- AURORA Algorithms
- AURORA Generalizations
- Experimental Results
- Conclusions
AURORA Algorithms

- **Goal:** select a set of $k$ influential graph elements

- **Observation:**
  - $\frac{\partial f(r)}{\partial A}$ is a non-negative matrix, so does $\frac{df(r)}{dA}$.
  - Enjoys diminishing returns property \( \Rightarrow \text{submodular function} \)

- **Greedy Strategy:**
  - iteratively select the most influential element in each round;
  - remove the selected element and re-rank;
  - repeat above procedure $k$ rounds.

- **Challenges:** computationally expensive to calculate $\frac{\partial f(r)}{\partial A}$

- **How to speed up?** \( \Rightarrow \text{power iterations} \)
Roadmap

- Motivations
- AURORA Formulation
- AURORA Algorithms
- AURORA Generalizations
- Experimental Results
- Conclusions
AURORA Generalizations: Normalized PageRank

- **Intuition:** normalize PageRank vector to magnitude of 1
- **Key Idea:** divide each PageRank score with the sum of all PageRank scores
- **Formulation:**
  - Let $S(r) = \sum_{i=1}^{n} r(i)$, then
    \[
    \frac{\partial f(r)}{\partial A} = cQ'(-\frac{2f(r)}{S(r)}1 + \frac{2}{S(r)}r)r'
    \]
- **Solution:** apply similar strategy as AURORA
- More details in the paper
**AURORA Generalizations: NoN**

- **NoN** (Network of Networks) is defined as a triplet $< G, A, \theta >$.
  - $G$: main network
  - $A$: domain-specific networks
  - $\theta$: mapping function

- **Ranking on NoN:**

  $$\min J(r) = cr'(I_n - A)r + (1 - c)\|r - e\|_F^2 + 2ar'Yr$$

  - within-network smoothness
  - query preference
  - cross-network consistency

  - equivalent to PageRank with transition matrix $W = \frac{c}{c+2a}A + \frac{2a}{c+2a}Y$

- **Solution:** Apply similar strategy as AURORA

---

[1] Ni, J., Tong, H., Fan, W., & Zhang, X. (2014, August). Inside the atoms: ranking on a network of networks. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 1356-1365). ACM.
AURORA Generalizations: Attributed Networks

- **Intuition**: find influential attributes in attributed networks.
- **Key Idea**: treat attributes as *attribute nodes* and form an *augmented graph*.
- **Supporting Node Attributes**:
  - (1) $A$: node-to-node adjacency matrix;
  - (2) $W$: attribute-to-node adjacency matrix.
  - Form an augmented graph $G = \begin{pmatrix} A & W' \\ W & A' \end{pmatrix}$
- **Supporting Edge Attributes**:
  - Let $A$ be an $n \times n$ adjacency matrix and $x$ be the number of different edge attributes.
  - Embed edge attributes into edge-nodes.
  - Form an $(n + x) \times (n + x)$ augmented graph.
- **Solution**: Apply similar strategy as AURORA

[1] Tong, H., Faloutsos, C., Gallagher, B., & Eliassi-Rad, T. (2007, August). Fast best-effort pattern matching in large attributed graphs. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 737-746). ACM.
[2] Pienta, R., Tamersoy, A., Tong, H., & Chau, D. H. (2014, October). Mage: Matching approximate patterns in richly-attributed graphs. In *Big Data (Big Data), 2014 IEEE International Conference on* (pp. 585-590). IEEE.
Roadmap

- Motivations ✓
- AURORA Formulation ✓
- AURORA Algorithms ✓
- AURORA Generalizations ✓
- Experimental Results
- Conclusions
## Datasets

- Over 10+ real-world datasets

| Category     | Network | Type | Nodes     | Edges    |
|--------------|---------|------|-----------|----------|
| **SOCIAL**   |         |      |           |          |
| Karate       | U       | U    | 34        | 78       |
| Dolphins     | U       | U    | 62        | 159      |
| WikiVote     | D       | D    | 7,115     | 103,689  |
| Pokec        | D       | D    | 1,632,803 | 30,622,564 |
| **COLLABORATION** |     |      |           |          |
| GrQc         | U       | U    | 5,242     | 14,496   |
| DBLP         | U       | U    | 42,252    | 420,640  |
| NBA          | U       | U    | 3,923     | 127,034  |
| cit-DBLP     | D       | D    | 12,591    | 49,743   |
| cit-HepTh    | D       | D    | 27,770    | 352,807  |
| cit-HepPh    | D       | D    | 34,546    | 421,578  |
| **PHYSICAL** |         |      |           |          |
| Airport      | D       | D    | 1,128     | 18,736   |
| **OTHERS**   |         |      |           |          |
| Lesmis       | U       | U    | 77        | 254      |
| Amazon       | D       | D    | 262,111   | 1,234,877 |

(In Type, U means undirected graph; D means directed graph.)
Experimental Settings

- Evaluation Metric
  - Effectiveness: difference in $f(r)$
  - Efficiency: running time

- Baseline Methods

| AURORA (Our Methods) | Baseline Methods          |
|----------------------|---------------------------|
| AURORA-E             | Brute force               |
| AURORA-N             | Random selection          |
| AURORA-S             | Top-k degree              |
|                      | PageRank                  |
|                      | HITS                      |
Effectiveness: Fixed Budget
(Higher is Better)

- Observation: AURORA outperforms baseline methods
Observation: AURORA outperforms baseline methods.
Efficiency

- **Observation:** linear complexity w.r.t. $k$ and $m$
Case Study on Airport Dataset

- **Goal:** find important airline routes and airports

- **Results:**

| Task                | PageRank   | AURORA          |
|---------------------|------------|-----------------|
| Edge Auditing       | ATL-LAS    | **DEN-ATL**     |
|                     | ATL-DFW    | **LAX-ORD**     |
| Node Auditing       | SFO        | **CLT**         |

DEN serves as a major hub airport to connect west and east coasts.

It directly connects Los Angeles (LAX) and Chicago (ORD), two largest cities in United States.

Busiest Airports: CLT(6th) > SFO (7th).
Proximity: existence of LAX and SJC.
Case Study on NBA Dataset

- **Goal:** find a team in collaboration network
- **Query:** Allen Iverson
- **Results:**

| Task                              | PageRank          | AURORA               |
|-----------------------------------|-------------------|----------------------|
| Subgraph Auditing (Graph size: 5) | Allen Iverson     | Allen Iverson        |
|                                   | Larry Hughes      | Larry Hughes         |
|                                   | Theo Ratliff      | Theo Ratliff         |
|                                   | Joe Smith         | Joe Smith            |
|                                   | **Drew Gooden**   | **Drew Gooden**      |
|                                   |                   | **Tim Thomas**       |

NEVER played with Allen Iverson.
Roadmap

- Motivations
- AURORA Formulation
- AURORA Algorithms
- AURORA Generalizations
- Experimental Results
- Conclusions
Conclusions

- **Problem:**
  - PageRank Auditing Problem
- **Solution:**
  - Family of AURORA algorithms
  - Near-optimal results
  - Scalability
- **Results:**
  - Outperform other baseline methods
  - Achieves linear time complexity
  - Finds intuitive and meaningful explanations
- **More details in the paper**