Computing Trusted Authority Scores in Peer-to-Peer Web Search Networks

Josiane Xavier Parreira ⋆ , Debora Donato ⋄ ,
Carlos Castillo ⋄ , Gerhard Weikum ⋆

⋆ Max-Planck Institute for Informatics
⋄ Yahoo! Research

May 8, 2007

Introduction

Motivation

- P2P systems for storing and sharing information.
- Decentralized nature opens doors to malicious behaviors from peers.

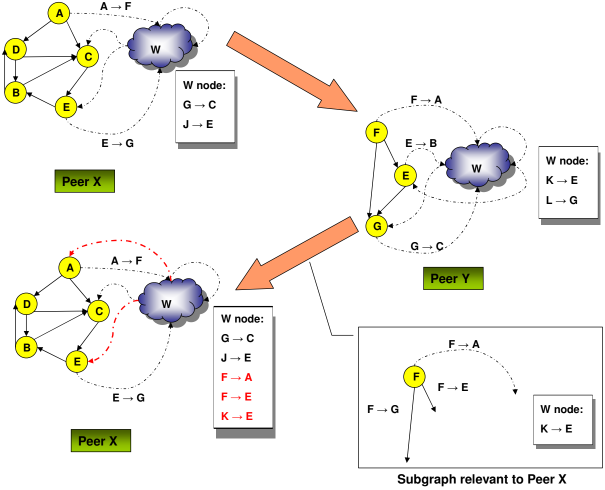
Previous Work

- JXP algorithm for computing decentralized PageRank-style authority scores in a P2P network [VLDB'06].
- Assumes peers are always honest.

Contribution

- Decentralized reputation system to be integrated into JXP.
- Allows computation of "trusted" authority scores.

JXP Algorithm [VLDB'06]



TrustJXP Algorithm

Idea

- Detect when peers report false scores at the meeting phase.
- Analyze peer's deviation from common features that constitute usual peer profile.

Forms of attack addressed

- Peers report higher scores for a subset of their local pages.
- Peers permute the scores of its local pages.

Malicious Increase of Scores

Why peers cheat

High authority scores for local pages can bring benefits to a peer.

Our approach

- Analyse the distribution of the scores reported by a peer.
- Use histograms to store and compare score distributions.
- Motivation: Web graph is self-similar $\rightarrow$ local scores distribution should resemble global distribution after a few iterations.

Histograms

Histograms

- Each peer stores a histogram H.
- Scores from other peers are inserted after each meeting.
- A novelty factor accounts for the dynamics of the scores.

$$H^{(t+1)} = (1-\rho)H^{t} + \rho D$$

D is the score distribution of the other peer, and ρ is the novelty factor.

Histograms

Comparing Histograms

Hellinger Distance

$$HD_{i,j} = \frac{1}{\sqrt{2}}[\sum_k(\sqrt{H_i(k)} - \sqrt{D_j(k)})^2]^{\frac{1}{2}}$$

k = total number of buckets
$H_i(k)$ and $D_j(k)$ = number of elements at bucket k at the two distributions

Malicious Permutation of Scores

Problem

- Peers can cheat and yet keep the original score distribution.
- Histogram comparison not effective in this case.

Our approach

- Compare the rankings from both peers for the overlapping graph.
- Observation: Relative order of scores very close to the actual ordering, after few meetings.

Comparing Rankings

Tolerant Kendall's Tau Distance

$$K'_{i,j} = |(a,b) : a < b \wedge score_i(a) - score_i(b) \geq \Delta$$
$$\wedge\, \tau_i(a) < \tau_i(b) \wedge \tau_j(a) > \tau_j(b)|$$

$score_i(a)$, $score_i(b)$ $=$ scores of pages a and b at peer i
τ_i, τ_j $=$ rankings of pages at peers i and j
Δ $=$ tolerance threshold

TrustJXP Algorithm

Computing Trust Scores

- Idea: Combine previous measures to assign trust scores to peers.
- Each peer assigns its own trust score to another peer, at each meeting step.
- How to combine the measures? We take a "safer" approach.

$$\theta_{i,j} = min(1 - HD_{i,j}, 1 - K'_{i,j})$$

- Trust score is integrated to the JXP computing, at the merging lists phase.

Integrating Trust Scores and JXP Scores

Integrating Trust Scores and JXP Scores

- When merging lists, scores from both lists can be combined by either averaging or taking the max score.
- If page is not present on a list $\rightarrow$ score $= 0$.

Averaging the scores

JXP: $L'(i) = (L_A(i) + L_B(i))/2$

TrustJXP: $L'(i) = (1 - \theta/2) * L_A(i) + \theta/2 * L_B(i)$

Taking max score

JXP: $L'(i) = max(L_A(i), L_B(i))$

TrustJXP: $L'(i) = max(L_A(i), \theta * L_B(i))$

Experimental Results

Web collection

- Obtained using a focused crawler.
- 134,405 pages, 1,915,401 links.
- 10 categories.

Setup

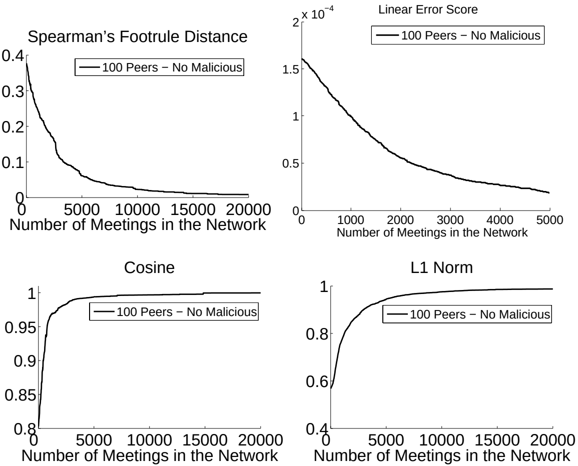
- 100 honest peers, 10 peers/category.
- Malicious peers
  - Perform JXP meetings and local PR computation like a normal peer.
  - Lie when asked by another peer about the local scores, according to attacks previously described.

Experimental Results

Evaluation Measures

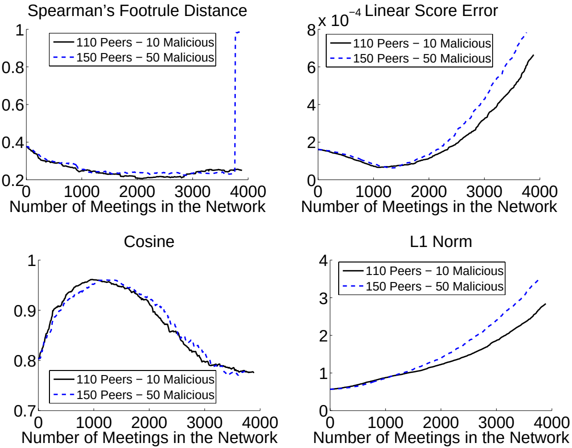
- "Global" JXP ranking vs. Global PageRank ranking.
- Spearman's Footrule Distance at top-k.
- Linear error score at top-k.
- Cosine at full ranking.
- L1 norm of full JXP ranking (L1 norm of Global PR always 1).

JXP Performance - No Malicious Peers

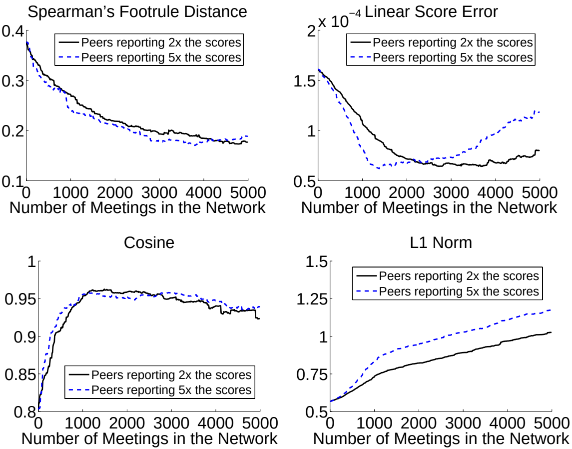


Impact of Malicious Peers

(Peers report 2x the true score value for all local pages)



Averaging the Scores



Trust Model

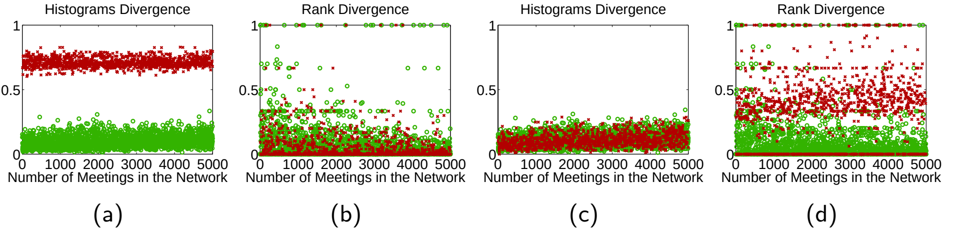
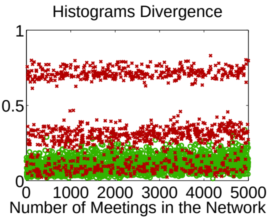
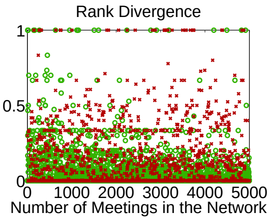
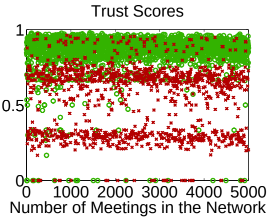


Figure: Increased-scores attack: (a) and (b). Permuted-scores attack: (c) and (d). A green circle ($\circ$) represents a meeting between two honest peers, and a red cross ($\times$) a meeting between an honest and a dishonest peers.

Trust Scores (Random Attacks)

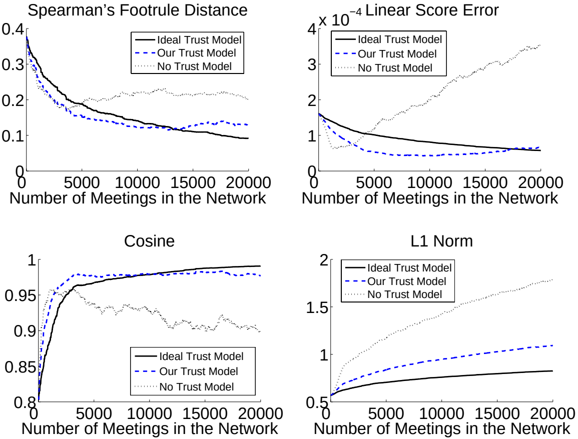






Max. θ	Detection rate	False positives
0.9	37.4%	4.7%
0.8	86.9%	12.1%
0.6	98.0%	54.5%

Trust JXP



* 150 Peers - 50 Malicious; Mixed malicious behavior

Conclusion

- TrustJXP algorithm for identifying and reducing the impact of cheating peers.
- Uses scores distribution and ranking analysis to detect malicious behavior.
- Experiments demonstrate viability of the method.

Future Work

- Detect other types of malicious behaviors.
- Network dynamics.