Modeling Temporal Behavior in Large Networks

From Predictive Modeling to Anomaly Detection

Ryan Rossi*, Brian Gallagher†, and Keith Henderson†

*Purdue University † Lawrence Livermore National Laboratory

rrossi@cs.purdue.edu *{bgallagher, keith}@llnl.gov*

Problem & Motivation

Problem: Understand and model the *behavior* and *evolution* of large complex networks and their individual nodes.

Motivation: An accurate model of behavior is important for

1. understanding network evolution and behavioral patterns,
2. predicting future behaviors,
3. detecting unusual behavior or compromised machines.

Challenges

- Overcome limitations of static models by leveraging and modeling the temporal dependencies
- Large temporal networks, missing data (active/inactive nodes and links), …
- Behaviors of nodes (and the semantics of roles) change
  - Roles must generalize across time
  - Anomalies and behaviors localized in time

Behavioral Analysis Framework

Given a sequence of *evolving graphs* where nodes and edges appear and disappear over time (limited assumptions):

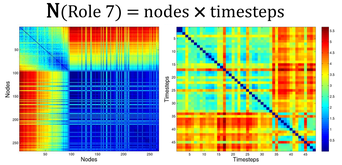
1. Learn features and extract them for each snapshot from the sequence of networks over time, $\boldsymbol{V} = \{V_t : t \in \mathrm{T}\}$
2. Discover behavioral roles using Non-negative matrix factorization (NMF) such that, $V_t \approx G_t F$, where
   $$V_t \in \mathbb{R}^{n \times f} \approx G_t \in \mathbb{R}^{n \times r}\ F \in \mathbb{R}^{r \times f}$$
3. Iteratively estimate $\boldsymbol{G} = \{G_t : t \in \mathrm{T}\}$ given F and $\boldsymbol{V} = \{V_t : t \in \mathrm{T}\}$ using NMF.

Understanding Temporal Behavior

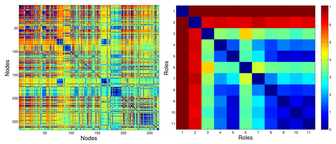
Mining Nodes, Time, and Roles.

Distance/similarity of nodes, roles, and timesteps

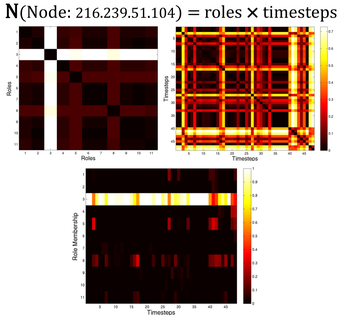
$\mathbf{N}(\text{Role 7}) = \text{nodes} \times \text{timesteps}$



$\mathbf{N}(\text{Time 34}) = \text{nodes} \times \text{roles}$



$\mathbf{N}(\text{Node: 216.239.51.104}) = \text{roles} \times \text{timesteps}$



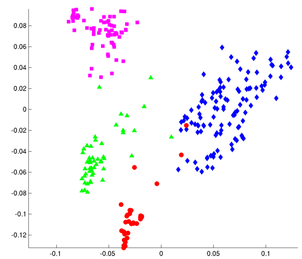
Temporal Behavioral Clustering.

Problem: Cluster nodes with similar behaviors (active vs. inactive or more complex motifs).

1.Estimate model for each node: $\mathrm{T}^{(i)}$

2.Compute similarity matrix: $S^{(i,j)} = \|\mathrm{T}^{(i)} \quad \mathrm{T}^{(j)}\|_F$

3.Clustering (NMF or traditional algorithm)



* ***Captures temporal patterns of nodes and roles***

Predicting Temporal Behavior

Given G_{t-1} and G_t estimate a transition model $T_{r \times r}$ (via NMF) that best approximates the network and node behavioral transition from t to t-1, such that $G_{t-1}T$ is the closest approximation of G_t. All models predict G_{t+1} using G_t.

Proposed Models.

Snapshot: Uses only the immediate past

Stacked: Uses training examples from k previous timesteps

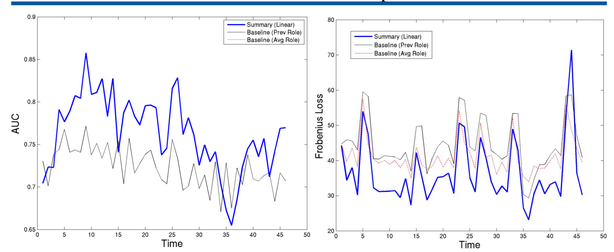
Summary: Weights the training examples using temporal dependencies from the k previous timesteps

Stacked-Summary: Hybrid of stacked & summary

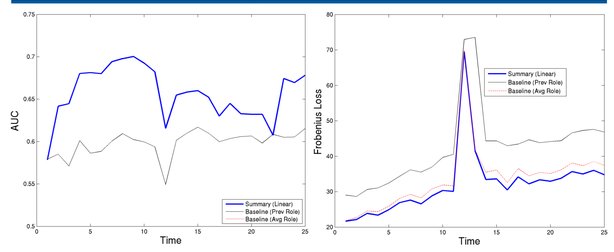
Multi-state Models. Active and inactive states are modeled independently using the above model types, denoted T_{active} and $T_{inactive}$

Baseline Models. Predict future role based on (1) previous role and (2) average role distribution.

Real-World IP Network Traces: Timesteps are 15 minutes



Internet Movie Database (IMDB): 1980 - 2007



* ***Summary models* most accurately predict future behaviors**

* **Behavioral roles generalize across time**

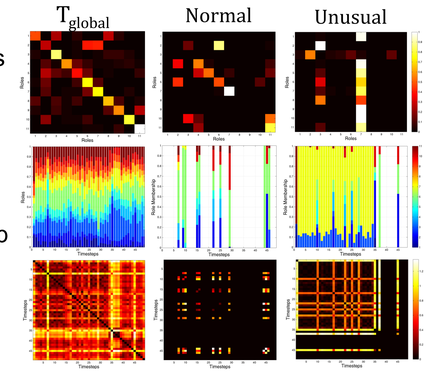
Detecting Anomalous Temporal Behavior

Transition Anomalies.

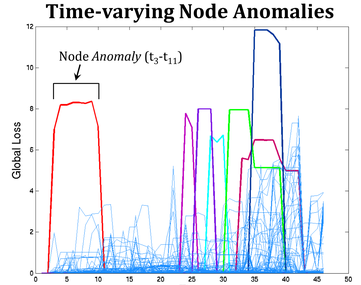
Network Anomalies. Track global loss

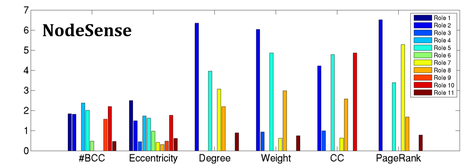
Node Anomalies.

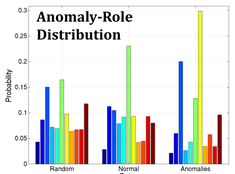
1. Estimate global model: T_{global}
2. Estimate model for each node: $T^{(i)}$
3. Anomaly score: $\|T_{global} \quad T^{(i)}\|_F$
4. Repeat steps 1-3 for each timestep

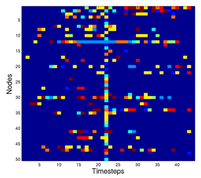


Time-varying Node Anomalies

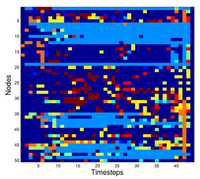




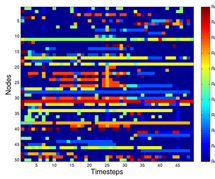




Randomly Sampled Nodes



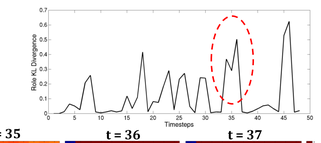
Eigen-Anomalies

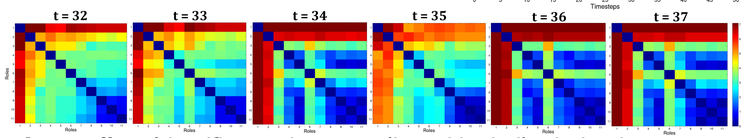


Transition-Anomalies

Eigen-Anomalies.

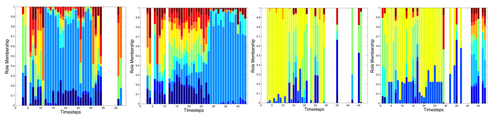
Network Anomalies. KL-Divergence of the principal-eigenvector of G_{t-1} *and* G_t





Node Anomalies. Identifies node anomalies with similar behaviors

1. Pairwise Euclidean distance of nodes
2. Compute k principle-eigenvectors
3. Sort each eigenvector



* **Discovers nodes and timesteps with *unusual behavioral transitions***

* **Detects nodes and timesteps with significantly different role memberships.**

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract No. DE-AC52-07NA27344.

LLNL-POST-492577