1 Introduction

A network, with nodes representing entities and edges the connections between entities, is a representation of data or events. The data does not necessarily come in a pre-defined network structure, but rather the network is a representation generated from the data as sequences of recorded events (e.g., trajectories of vehicles, retweets of a message, streams of web clicks, etc.). Assuming that network analysis is the right framework for analyzing such data, it begets the question: how to construct network representations from data, such that the underlying phenomena in data are correctly captured? The correct representation is essential as all further analyses are performed on this input network, and if this input network is not a precise representation of the underlying data, then the result of such analyses become questionable.

For the same data set, there can be various network representations, depending on how nodes/edges are defined and/or weighted. The conventional way to construct a network uses nodes to represent entities, and creates edges from pairwise connections between entities, where edge weights are assigned as the sum of pairwise connections — e.g., the traffic between locations in an interval [1] [2] [3], the sum of the available space on all ships traveling between ports [4], the sum of weighted intensity of scientific collaborations [5].

The problem with such conventional network representation is that it implicitly assumes the Markov property (first order dependency) by considering only pairwise connections in data, while patterns that involve more than two entities are not represented in the network. That is, in a conventional network representation, the movement simulated on the network is only able to follow the probability distribution of first order (i.e., pairwise, dyadic) relationships and cannot reflect higher order dependencies in data, whereas movements may not only depend on the current location but also depend on previous locations. Furthermore, failure to represent such dependencies in the network will lead to inaccurate or even incorrect results when applying a wide range of network analysis tools that are based on the simulation of movements on the network, such as clustering with MapEquation [6], ranking with PageRank [7], various link prediction methods based on random walking [8], and so on.

Contributions:

Our Higher Order Network (HON) is:

- able to represent higher order dependencies in data, so that the simulation of movements on the network will be more accurate when data has higher order dependencies;
- compact in size by using variable orders of dependencies, i.e., instead of representing the whole network with a fixed high order, HON uses variable orders by adding auxiliary nodes and edges to a first order network only where necessary;
- compatible with existing network analysis tools, because HON is constructed by adding / rewiring a few nodes and edges to the conventional network representation, by using HON instead of a first order network, network analysis tools based on the simulation of movements can produce more accurate results.
2 Methods

The construction of the Higher Order Network (HON) consists of two steps: Rule extraction identifies higher order dependencies that have sufficient support and can significantly alter the probability distribution of choosing the next step — instead of assuming a fixed order for the whole network [9], we extract the true orders of dependencies for every path in the data; then network wiring adds these rules describing variable orders of dependencies into the conventional first order network by creating / relabeling nodes and rewiring edges, such that different orders can co-exist in the same network. The resulting network consists of only simple nodes and edges, so existing network analysis methods can be applied directly without being modified.

3 Experiments and results

Data. We use a synthetic data set to verify the correctness of our algorithm. We also use three real-life data sets of different types: the global shipping data that contain vessel movement information, the clickstream data that record users clicking through web pages, and the retweet data that records the flow of information on a social network.

Simulation of movements. The simulation of movements performed on HON is considerably more accurate than on existing network representations: for the global shipping data, for example, the accuracy for simulating one step on HON has more than doubled compared with first order network, and higher by one magnitude when simulating three steps.

Application examples: clustering and ranking. For network analyses methods based on the simulation of movements, we also show that by using HON instead of first order networks can yield more insights. For example, the clustering results using MapEquation [6] on the global shipping data represented as HON naturally gives overlapping clusters, in which international ports are effectively identified. For the clickstream data set, the PageRank [7] results on the HON representation, compared with the conventional first order network, ranks news homepages higher while ranks error pages and second pages lower.

Application on anomaly detection. Besides algorithms based on the simulation of movements, other applications such as anomaly detection on dynamic networks can also benefit from the HON. By using HON instead of first order network as the input of dynamic network anomaly detection algorithm, higher order anomalies that may have been ignored by using the first order network can be discovered by using HON.

Scalability and parameters. We show that HON is considerably smaller than the representation that uses fixed order of dependency, by adding auxiliary nodes and edges only where necessary. We also discuss the influence of minimum support, maximum order, and tolerance in terms of the accuracy of simulated movements.

4 Discussion

Our work has the potential to influence a wide range of applications as the network representation is consistent with the input expected by various network analyses methods. We have made the entire source code available as well. In future work we look forward to extending the application of HON beyond the simulation of movements to more dynamic processes, and improve the algorithm by reducing the parameters needed.

Acknowledgments

This work is based on research supported by the ND Office of Research via Environmental Change Initiative (ECI), the National Science Foundation (NSF) Award #1427157, the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053, the U.S. Air Force Office of Scientific Research (AFOSR), and the Defense Advanced Research Projects Agency (DARPA) grant #FA9550-12-1-0405. The discussions with Yuxiao Dong are gratefully appreciated.

References


