Negative Correlation in Traffic Recovery

Yichao Zhang[1],* Xing Li[1],† Jihong Guan[1],‡ and Shuigeng Zhou[2§]
[1]*Department of Computer Science and Technology,*
Tongji University, 4800 Cao'an Road, Shanghai 201804, China and
[2]*Department of Computer Science and Engineering, Fudan University, Shanghai 200433, China*
Shanghai Key Lab of Intelligent Information Processing, Fudan University, Shanghai 200433, China

With the rapid development of information and communication technology, we can access a wide range of network data. These data sets offer us the opportunity to analyze and predict network revolution. In this paper, we have proposed a way which combines link prediction methods to solve the traffic congestion problem. Based on the congested traffic line, we will predict the likelihood of new routes. One different from the link prediction problem is that in the traditional link prediction our goal was to find the edges to be created in the future, while in our traffic problem we have already known the edges to be predicted, our goal was to predict the sequence of these edges. Besides the algorithms we have proposed, we have conducted varies experiments to prove our methods to be effective to reduce the congestion index of the traffic. This work is an exploration of using link prediction method in the traffic problem, it provides a new view in the traffic control problem and may further affect the link prediction usage in the traffic field.

INTRODUCTION

Network structures have been studied for many years. First research in this area can be traced back to 1736 when Euler defined and solved the Seven Bridges problem of Knigsberg[1]. There was not much ground breaking development in the complex network research area until 1960s, when the Erdos-Renyi random graph model (ERmodel) was introduced. Due to the fact that there was a lack of large real world data, most of the work had been done on the analysis of phenomena existing in networked structures. During the time when ERmodel was introduced, progress has also been made by sociologists in researching real world human relationships . A new wave of research was set off by Watts and Strogatz who published a paper about the small world effect in 1998 and introduction of the scalefree net- work model by Barabasi and Albert one year later.

Link prediction in complex network is one of the popular research topics. Generally, the prediction problem is mainly studied from two angles: (i) network structure and (ii) attributes of nodes and connections. Structure reflects the information about network topology. Majority of the progress in the area of structure based prediction has been made by mathematicians and physicists. Some of the well known structure based prediction methods are Common Neighbor, Preferential Attachment Index, Adamic-Adar Index, Katz, etc. The link prediction problem also has been studied from the angle of the network attribute information.The majority of attribute-based prediction methods follow a machine learning approach, i.e. they use classificationbased methods to make predictions. Authors report that the performance of link pre- diction improves when machine learning approaches are used. However, this is done using additional network information that is not always available[2–6].

RELATED WORK

The simplest framework of link prediction methods is the similarity-based algorithm, where each pair of nodes x and y, is assigned a score s_{xy}, which is directly defined as the similarity between x and y. Similarity-based algorithm are classified into three categories: local indices; global indices; quasi-local indices, which do not require global topological information but make use of more information than local indices[7]. Considered of computation complexity here in our experiment we use two classic algorithm: PA in local indices and LP in quasi-local indices and other transformed algorithms.

*Electronic address: yiczhang@cs.ucl.ac.uk
†Electronic address: 15000218895@163.com
‡Electronic address: jhguan@tongji.edu.cn
§Electronic address: sgzhou@fudan.edu.cn

(1) Preferential Attachment Index (PA)[8]. The mechanism of preferential attachment can be used to generate evolving scale-free networks, where the probability that a new link is connected to the node x is proportional to k_x. A similar mechanism can also lead to scale-free networks without growth , where at each time step, an old link is removed and a new link is generated. The probability that this new link will connect x and y is proportional to k_x k_y. Motivated by this mechanism, the corresponding similarity index can be defined as

$$S_{xy}= k_x * k_y \tag{1}$$

(2) Local Path Index (LP)[9, 10]. Denote by A the adjacency matrix, where $A_{xy} = 1$ if x and y are directly connected, and $A_{xy} = 0$ otherwise. Obviously, $(A^2)_{xy}$ is the number of common neighbors of nodes x and y, which is also equal to the number of different paths with length 2 connecting x and y. And if x and y are not directly connected , $(A^3)_{xy}$ is equal to the number of different paths with length 3 connecting x and y. The information contained in A^3 can be used to break the degeneracy of the states, and thus we define a new measure as

$$S= A^2 + \varepsilon A^3 \tag{2}$$

The score of S_{xy} correspond to the element is S. And in our experiment we set $\varepsilon= 0.2$.

(3) Three order Similarity Index (SA). This algorithm are proposed in this study, the purpose was to magnify the recovery accuracy of the traffic routes. $\Gamma^3(x)$ means the set of first order neighbor second order neighbor and third order neighbor in all.

$$S_{xy}= |\Gamma^3(x) \cap \Gamma^3(y)| \tag{3}$$

The followed four algorithm are the transformer of the above three algorithm.

(4) Reverse Preferential Attachment Index (RPA). simply put the numerator into denominator, which means nodes with less neighbors play a more important role.

$$S_{xy} = \frac{1}{k_x \times k_y} \tag{4}$$

(5) Reverse Add Preferential Attachment Index (RAPA). One different from RPA is that at the denominator part we transfer multiply into add.

$$S_{xy} = \frac{1}{k_x + k_y} \tag{5}$$

(6) Reverse Local Path Index (RLP). simply put the numerator into denominator

$$S = \frac{1}{A^2 + \varepsilon A^3} \tag{6}$$

The score of S_{xy} correspond to the element is S. And in our experiment we set $\varepsilon= 0.2$.

(7) Reverse Three order Similarity Index (RSA). simply put the numerator into denominator

$$S_{xy} = \frac{1}{|\Gamma^3(x) \cap \Gamma^3(y)|} \tag{7}$$

DATASETS

To verify our methods, we have conducted varies experiments on three kinds of networks. The networks are as below:

BA network

we select the Barabasi-Albert (BA)[11] model to generate the topological network, the degree distribution of which is a power law $p(k) \propto k^{-3.0}$ where p(k) is the ratio of the nodes with degree k to the number of all nodes in the network. The model could represent the heterogeneous node degree of many real-world networks, including the Internet AS

	$\|V\|$	$\|E\|$	D	C	$< k >$	$< d >$	H
BA256	256	508	0.016	0.075	3.97	3.49	2.17
BA512	512	1020	0.008	0.052	3.98	3.68	2.87
BA1024	1024	2044	0.004	0.030	3.99	4.07	2.78
USAir	332	2126	0.039	0.625	12.81	2.74	3.46
Cal Road 10	121	156	0.021	0.142	2.58	9.54	1.13
Cal Road 20	409	525	0.006	0.101	2.57	17.71	1.17
Cal Road 30	1285	1750	0.002	0.077	2.72	23.28	1.16

TABLE I: statistics of the six networks.

level topology, the logical topology of unstructured P2P distributed systems, and so on. In the simulation, we set the mean degree of the network k=4.

USAir

We got USAir data from pajek datasets, The network is extracted from the USAir transportation system. The data contains 332 nodes and 2126 edges. Nodes in the network are individual cities and edges are airlines between these cities. Notice that this is a undirected network which means if you can buy a ticket from city A to city B then you can also buy a ticket from city B to city A. In some degree, USAir is a kind of BA network, some major City like New York and Los Angeles owns more airlines than small cities.

California road network

The dataset was obtained from Stanford dataset. The original data contains 1,965,206 nodes and 2,766,607 edges. Considered the network is too large for link prediction and traffic simulation, so we split out some small part from the California road network. First we randomly choose a node and take it as the center node, then we use Level Traverse method to spread the network out and we can get different size of road network for our needs. For example, if we take nodeId= 100 as the center node, then we select the node 100s neighbor and append them into the new network, then append the 100s neighbors Neighbor, and the next steps are as before. In California road network, if we choose nodeId= 800120 as the center node and the spread level is 30 we can get a network with 1285 nodes and 1748 edges.

The basic topology statistics of the six networks are summarized in Table I.

$|V|$ is the number of vertices and $|E|$ is the number of edges in a given network. D denotes the density of the network which is to calculate $\frac{2|E|}{|V-1|*|V|}$. A nodes clustering coefcient states that, if there are k_i neighbors owned by a vertex v_i and the maximal feasible edges among them could be $k_i * (k_i 1)/2$, the local clustering coefcient for the vertex v_i is to calculate $C_i =$ [14], where N_i stands for the number of neighbors of v_i. C denotes the average clustering coefcient. $|k|$ is the average degree of the network. Here, a shortest path is dened as a path connecting two unconnected nodes with least edges and thus the shortest path distance is the number of edges existed within the shortest path. $< d >$ denotes the average shortest path distance of the network. Degree heterogeneity is a statistical property, which quantitatively characterizes the fluctuation of the degree sequence of a network. This property was frequently mentioned, while its explicit definition was not manifested until a recent work[12]. To be consistent with previous studies, we denote the degree heterogeneity by H. H is defined as $\frac{2|E|}{|V-1|*|V|}$, where $< k >$ denotes the average degree[13].

EXPERIMENT

In this section we will give our method to improve the transmission efciency of the trafc network, show the simulation results, and then discuss them extensively.

1.Method

1) In order to test our method we should generate traffic- like topological networks. Here we got three kinds of networks, BA network, USAir network and California road network. To simplify the simulation, they are set to be unweighted and undirected.

2) The next step is to simulate the congestion of the traffic net. We think that in the status of congestion the traffic in the route almost Stagnated, so to some degree it equals the route has been broken. So in this process we randomly select a part of the route, and break them up. Notice that we allow the situation that after the break of the network, there are exists individual nodes.

3) Since we have broken links, so we can use link prediction methods to rank the edges of nonsexist, and determine the sequence of recovery links. One different from link prediction is that in link prediction we will rank all the edges of nonsexist while in route recovery we just need to judge the important of broken edges and recover them in order. Here we have eight methods to recover the routes which was mentioned before.

4)To verify the efficiency of different methods based on link prediction, we have conducted package simulation experiments. Several models have been proposed to simulate packet trafc dynamics on complex networks by introducing random generation of packets in each time step and various routing strategies. We here adopt one widely used before.

In each time step, each node generates $\frac{R}{N}$ packets, where N is the size of the network and R is a parameter tuning the generation rate, i.e., there are R packets generated in the network at each step. When $\frac{R}{N}$ is not an integer, we create Int($\frac{R}{N}$) packets determinately and create a packet simultaneously with probability p=$\frac{R}{N}$ - Int($\frac{R}{N}$), where Int($\frac{R}{N}$) is the integral part of $\frac{R}{N}$ and thus p is the fractional part. The packets are initialized with random destinations. Moreover, we set a transmission capacity C_i to the node i, i=1,2, ... ,N, which means that, at each time step, the maximal number of packets transferred by the node i to the next node according to the routing table is C_i . When the node cannot transfer all the packets accumulated in its queue, it deals with them following the rst-in-rst-out rule. Hence, the routing strategy also plays an important role in the trafc dynamics. We see that, in the Internet, the routing strategy of within domains is the shortest path algorithm and, between domains, i.e., for the AS level, the border gateway protocol causes the packets to be transmitted along almost the shortest path[15]. Therefore, in the paper, we adopt the shortest path routing strategy. When a packet reaches its destination through the shortest path routing, the packet will be deleted from the system.In order to analyze the transition from free ow to a congested state, we use the order parameter presented in previous studies[16],

$$\mu < R >= \lim_{t\to\infty} \frac{<\triangle w>}{R\triangle t} \tag{8}$$

where W(t) is dened as the number of packets on the network at time step t, and W=W(t+t)W(t), with $< >$ indicating averaging over time windows of width t. In other words, the order parameterrepresents the ratio between the existing ow and the inow of packets calculated through a long enough time period. Obviously, in the free ow state, i.e., where there is no congestion in the network, the system can deal with the generated packets and thus the existing ow is close to zero. Otherwise, in the congested state, the number of generated packets is too large to be transmitted and the ow existing in the network will also be large, which causes the order parameter to approach 1.

2.Metrics for evaluation

We have got eight methods to recovery the links in the network. Different methods get different recovery efficiency. First we initialized an array like bestPerform = [0, 0, 0, 0, 0, 0, 0, 0], the array contains 8 elements and each element correspond to a method, the methods are [Random, PA, RPA, RAPA, LP, RLP, SA, RSA] in order. In a certain congestion situation, we use these different methods to recover the links respectively, when we have finished the recovery of the roads, we will calculate the order parameter $\mu < R >$. If one method get a lowest order parameter in this period (recovery process) of time, we will choose the corresponding element in the array bestPerform and add by 1. We will repeat the process in a certain times, and finally count the elements in the bestPerform. If an element got the highest value, then it means that the corresponding methods performs best in the link recovery.

3.Result

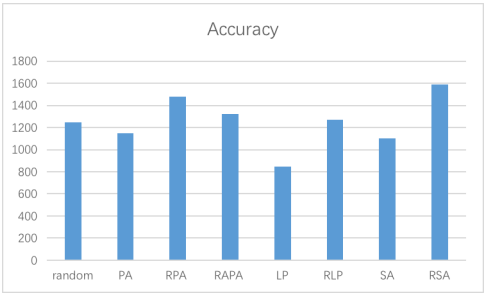

FIG. 1: In this experiment, we first create a California road network, the data was obtained from stanford Datasets.Intersections and endpoints are represented by nodes and the roads connecting these intersections or road endpoints are represented by undirected edges.The network contains 1,965,206 nodes and 2,766,607 edges.Since the network is too large to process, so we need to select a part of the graph. We randomly choose a node from the network, like nodeId= 800120, then we regard the node as the center node, next we use the center node to spread out the network, just like BFS algorithm.Here in this experiment, we set the spread level equals 15, and we get the network with 223 nodes and 291 edges.

Since road network has the lowest heterogeneity among BA and USAir, the degree of the nodes in the network mainly lies in 1,2,3,4,5, so it is hard to identify the important of the edges. But from 10,000 round of comparison, our experiment still reveals the Negative Correlation effect in road network. RPA, RAPA, RLP, RSA performs better than random way, and PA, LP, SA performs badly.From the fig1, we can observe that RSA performs best, I think

R	5	10	15	20	25	30	35	40	45	50
Order parameter	0.005	0.24	0.34	0.46	0.54	0.60	0.64	0.68	0.70	0.73

TABLE II: Correspond to Fig.2, the average order parameter in different value of R.

this may be the reason that we have fully utilize the information of one node in the situation of low heterogeneity. This phenomenon also remind us to use more similarity information to improve the performance of road recovery.

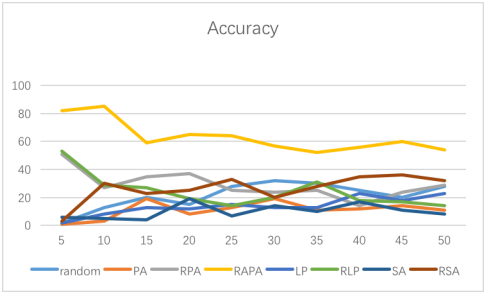

FIG. 2: In order to verify the robust of different methods. We have controlled the value of R. In the above experiment, we create a USAir network, it has 332 nodes and 2126 edges. We then set the ratio of destroyed edges equals to 0.05. Under different R, we got the algorithm's performance as above.

USAir is a kind of transportation system. We can see RAPA always performs best, then follows as RLP, RSA.With the increasement of R, although negative way still performs better, but the gap between positive and negative ways are becoming smaller, the reason for this phenomenon may be that at a very small value of R, the congestion level of the roads are not serious, so the recovery of the roads can release the congestion level a lot. But with the increase of R and congestion, the recovery of roads can't change the congestion situation a lot, so the difference between the recovery methods isn't as big as above. Since the performance of negative ways overwhelms positive link prediction methods under different traffic flow rate, so our methods proves to be robustness. Since normal traffic system won't be too congested, so negative methods can be far more better than random recover the routes.

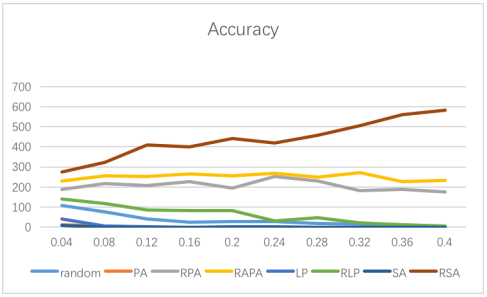

FIG. 3: In order to know how congestion ratio affect the recovery of traffic. We have generate a BA network with 256 nodes and 508 edges, x-axis represent the ratio of congested route. In the above picture, 0.04 means there are int(0.04*508) edges have been congested. Y-axis represent the accurate prediction sequences in 1000 round experiment. The packages generated in one time step is 10.

A lot of traffic system can be modeled as BA network. From fig.3 we can see with the increasement of fraction ratio, negative recovery methods perform far more better than random method and positive one's. At the fraction ratio of 0.4, in 1000 round experiment, RSA method performs best in about 600 round comparison, RPA and RAPA performs about 200 round best respectively while other methods almost have no chance in the guidance of traffic recovery. I think there are two reason for the result, with the increasement of fraction ratio, the recovery sequences have more combination so we can differentiate them more easily. Another reason is that big fraction ratio requires more time in the recovery process, in a larger period of time, we can verify our methods more easily.

In fig.4 we can see that when the network contains 100 nodes, eight methods haven't much difference in the traffic recovery, this is mainly because that at a very small network there are no big difference for each node so we can't distinguish them clearly. But when the size of the net reaches a certain scale nodes in the network play roles quite differently, hub nodes and normal nodes can't be the same. When the size of BA network reaches 200, the difference of performance can be observed easily. With the increasement of BA size, negative recovery methods perform better than

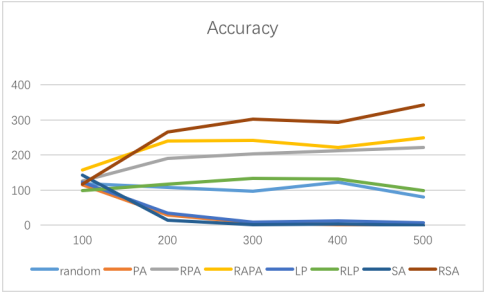

FIG. 4: To test our methods in different size of networks, we have generate five BA networks, they contain nodes with 100, 200, 300, 400, 500 respectively. The probability for each node to generate a package in one time step is 0.02 and the fraction ratio is 0.05.

random method and positive methods. At the size of 500, positive ways almost have no chance in accurately recover the routes in a certain sequence. The reason for that negative methods in larger BA network plays better mainly because the larger BA network has larger heterogeneity, so the nodes play roles differently and we can distinguish them more easily. Normally traffic system won't be too small, they usually contain nodes over 200, so our negative ways can be of use in routes recovery especially the reverse similarity(RSA) method.

CONCLUSION

Traffic problem affect the operation efficiency of the society, current study mainly focus on add new route or rebuild the network. But even the best system can't avoid congestion entirely. Our methods mainly focus on how to recover the routes sequences in a certain order to minimize the negative impact. Inspired by link prediction methods, we have recovered the the route in three strategies. These strategies are positive ways which include PA, LP and SA; negative ways which include RPA, RAPA, RLP and RSA; random way. From our experiment we find that negative ways perform better than the random recover of routes, random way are better than positive ways which is obviously since negative ways and positive ways are reverse methods. Beside the verification of negative link prediction methods in routes recovery. We also find that larger network can improve negative link prediction methods' accuracy; Higher degree heterogeneity improves experiment accuracy also, so negative methods are effective especially for BA like topological; At a high level of congested routes like half of the routes have been broken, negative methods almost have accuracy of 100% in the routes recovery.

Our study is tentative in using link prediction methods solving traffic problem, our result shows a tremendous potential combine link prediction methodology and real traffic problem. Since few people have doing researches in this field, so there are many shortcomings to be further optimized. In lack of real traffic flow data, so we simplified our simulation with a certain rate of packages generated in a certain time step. If possible, further studies can obtain real traffic flow and doing recovery experiment. Also weighted networks deserve to be discussed. In all, our research have found some simple ways to recover the congested ways efficiently, this may a new branch in the traffic problem.

Author contributions

Yichao Zhang and Xing Li contribute equally.

References

[1] N. Biggs, E. K. Lloyd, and R. J. Wilson. Graph Theory, 1736-1936. Claren- don Press, New York, NY, USA, 1986.
[2] MohammadAlHasan,VineetChaoji,SaeedSalem,andMohammedZaki. Link prediction using supervised learning. In In Proc. of SDM 06 workshop on Link Analysis, Counterterrorism and Security, 2006.

[3] Ryan N. Lichtenwalter, Jake T. Lussier, and Nitesh V. Chawla. New per- spectives and methods in link prediction. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Min- ing, KDD 10, pages 243252, New York, NY, USA, 2010. ACM.
[4] Zhen Liu, Qian-Ming Zhang, Linyuan Lu, and Tao Zhou. Link prediction in complex networks: A local naive bayes model. EPL (Europhysics Letters), 96(4):48007, 2011.
[5] Andrew Chen-Brian Tran Ole J. Mengshoel, Raj Desai. Will we connect again? machine learning for link prediction in mobile social networks. 2013.
[6] Kai Yu, Wei Chu, Shipeng Yu, Volker Tresp, and Zhao Xu. Stochastic relational models for discriminative link prediction. In Advances in Neural Information Processing Systems, pages 333340. MIT Press, 2007.
[7] L, Linyuan, and T. Zhou. "Link prediction in complex networks: A survey." Physica A Statistical Mechanics & Its Applications 390.6(2011):1150-1170.
[8] Xie Y B, Zhou T, Wang B H. Scale-free networks without growth[J]. Physica A Statistical Mechanics & Its Applications, 2008, 387(7):1683-1688.
[9] Zhou T, L L, Zhang Y C. Predicting missing links via local information[J].
[10] The European Physical Journal B, 2009, 71(4):623-630. Ackland R, Ackland R. Mapping the U.S. Political Blogosphere: Are Conservative Bloggers More Prominent?[C]// 2005.
[11] A.-L. Barabsi and R. Albert, Science 286, 509 (1999).
[12] Zhou T, L L, Zhang Y C. Predicting missing links via local information[J]. The European Physical Journal B, 2009, 71(4):623-630. 71(4):623630.
[13] Zhang Y, Aziz-Alaoui M A, Bertelle C, et al. Knowledge Diffusion in Complex Networks[C]// IEEE, Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE, Intl Conf on Autonomic and Trusted Computing and 2015 IEEE, Intl Conf on Scalable Computing and Communications and ITS Associated Workshops. 2015.
[14] Watts DJ, Strogatz SH. Collectivedynamics of small-world networks[C]// Nature. 1998:440-442.
[15] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U.Hwang, Phys. Rep. 424, 175 (2006).
[16] S. N. Dorogovtsev, J. F. F. Mendes. Evolution of networks[M]// Evolution of networks :. Oxford University Press, 2003:1842-1845.