ATLANTIS PRESS

Advances in Intelligent Systems Research, volume 147

International Conference on Network, Communication, Computer Engineering (NCCE 2018)

Research of Website Community Identification Based on Node2vec

Yang Xiao [a)] and Jun Liu [b)]

School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China

[a)] zackxy@bupt.edu.cn
[b)] liujun@bupt.edu.cn

Abstract. In recent years, the world has witnessed massive growth in the field of information technology and intelligent device, giving rise to exponentially expansion of internet websites in terms of quantity as well as complexity. Meanwhile, the public has also enlarged demand for characteristic analysis of internet websites, since they become an indispensable element of everyone's daily life. One significant focus is website community identification, which is able to optimize user experience and resource allocation to a large extent. In this paper, we present a comprehensive solution featuring node2vec algorithm to identify website community, expressly discovered website relationship from vast user behaviors which has been ignored for long. The experiment credibly demonstrates availability and reliability of our solution, which is significative in practical application.

Key words: Website community; Graph theory; Affinity measurement; node2vec.

INTRODUCTION

It is absolutely a misconception that the internet is getting more untraceable, even though the reality is the rapid development of information technology and craftsmanship prompts expansion of the internet, resulting in a tremendously complex topological structure of internet websites. In the meantime, more pathbreaking smart devices and internet applications spring up, facilitating web service and encouraging people's reliability. According to '2017 global social journalism study [1]', 90 percent interviewee use social networks (e.g., Facebook, Google+) in their work, people are more dependent on web services while generating huge amounts of data traffic. All these factors above attributed to a high-demanding requirement for service providers. Under the circumstance that the internet exists in the form of multi-cluster network, resource should be allocated unevenly. However, here we focus on the websites since it is regarded as terminal of browsing behavior. Practically, hub nodes can be biased designated, but community classification still contains many undiscovered relationships. Once explored and utilized appropriately, it will effectively optimize user experience and resource allocation. In this paper, we put forward an integrated algorithm characterized by affinity measurement and node2vec, which has a satisfactory performance in website community identification.

The internet is unique for its large scale and multi-connection, which makes graph theory the best representation for it. Generally speaking, nodes usually refer to websites while edges mostly represent shared users or common web content. A greater degree of similarity in shared users or common content corresponds to a better quality of connection. Significantly, a graph is said to have communities if some of nodes are closely connected internally, but have sparser edges connecting to the rest of the graph. A complete understanding of community allows us better taking advantage of website network. Existing community identification methods are generally either based on link or content, they respectively extract community information from the link structure of the hyperlinked web environment and define relationships between websites in terms of similarities between their contents. Although

many accomplishments have been carried out, there are still some drawbacks could not be avoided. Both two methods above work out source file though data mining or web crawler which consumes a lot of computing resource let alone time-efficiency uncertainty, they also neglect user behaviors but place emphasis on website content which can be easily misinterpreted under some circumstances.

User behaviors, constantly dynamic, well accord with the definition of website community since it reflects browse tendency of users and intimate relationship of websites, massive amount of user behaviors also eliminate subjectivity of personal behaviors. In this paper, we present an integrated website community identifying solution, featuring a new measurement for website relationship through network traffic extraction, as well as an innovative algorithm that maximizes the likelihood of preserving network neighborhoods of nodes while learning a mapping of nodes to a low-dimensional space [2]. At the very beginning we model a network of websites with their pageview as an affinity graph, nodes represent websites while edges refer to potential connections, but as excruciating as it is, the amount of edges increase intensively in the wake of network enlargement. In order to reduce workload and focus on priorities, we define a customized threshold to filter edges with inferior affinity measurement value. Afterwards we apply node2vec algorithm to construct feature representation vectors for nodes, which can derive a bunch of Euclidean Distance that signifying closeness of different websites. Assigning hub nodes by network flow or client's preference and choosing nodes with closest Euclidean Distance ultimately make up the hub-assigned website communities. A corresponding experiment followed gives a distinct expression to its practicability. In the last part of this paper, we give our conclusion.

BACKGROUND AND RELATED WORKS

Web community, originally defined as a set of websites shared by a group of people having similar interest [3], aims at effectively gathering these websites together. Many previous researches have been conducted in which abundant theoretical achievements and practice experience have been accumulated. However, most of the researches are based on intrinsic hyperlink of websites, the difference in principle is to model a network as a graph in which nodes referred to websites and edges referred to links. Gibson defined a hyperlink-based website community consists of authoritative central nodes and linked hub webpages [4], then the HITS algorithm [5] put forward that hyperlinks between websites make up a web graph, by calculating authority score and hub score we find out hub websites and authority websites to form a web community. Some researchers otherwise proposed using bipartite graph [3][6] by finding out all the subgraphs of bipartite graph and renamed core, then derive community from every core[7]. The maximum flow and minimum cut theorem is also applied to community identification, in which a community is specially defined as a group of nodes that have more connections with each other internally than nodes outside the group [8]. The identification work is to apply maximum flow and minimum cut algorithm and choose nodes linked to form communities. On the other hand, instead of graph theory, vectors are often used to represent content or hyperlink characteristics [9]. This approach is unique for reflecting a network's feature explicitly, also fitly be appropriate for various cluster analysis algorithm for instance, the K-mean clustering. Relative works [10] get satisfying results and have much reference value.

These identification methods have inspired countless people to fulfill their practical tasks. However, due to a large consumption of computing resource and overlook of user behaviors, there is still much to optimize.

METHODOLOGY

Firstly, we summarize our method into three parts: graph generation, feature learning and community identification. Specifically, we model these internet websites as a weighed and undirected graph named affinity graph [11], nodes referred to different websites while edges referred to connections we assume to exist between two nodes. By setting a threshold we can ignore weaker connections and reveal primary relationships in the graph explicitly for the sake of further research. The innovative node2vec algorithm gives us a brand-new way to learn characteristics of a graph by compiling it into low-dimensional space, naturally but creatively, we can identify communities through modifying parameters as well as measuring low-dimensional vectors' similarity.

TABLE 1. Adopted Notations

Notation	Description
G = (O, E, W)	The undirected and weighted affinity graph with node set O, edge set E and weight W
U	The set of total users
μ(oi)	The set of users accessing node oi
aff (oi, oj)	Affinity measurement value of oi and oj
spa (oi, oj)	The sparsification function of oi and oj
τ	Sparsify threshold
G' = (O, E')	The undirected and unweighted affinity graph with node set O, edge set E'
u	initial node
l	walk length
P	Transition probability
ni	the ith node in a walk
C	normalizing constant
α	search bias
dvu	the shortest distance between node v and node u
p	Return Parameter
q	In-out Parameter

Graph Generation

Given that we are dealing with problems concerning large number of websites and page view, we use a weighted and undirected graph G= (O, E, W) also an affinity graph to simulate websites with their relationships and save data systematically. O referred to the set of nodes representing websites, and E referred to the set of undirected edges representing relationships between websites, the weight W of which is calculated by what we defined as affinity measurement function.

Suppose U denotes the set of total users, μ(oi) denotes the set of users accessing node oi, obviously we know that:

$$\cup_{i=1}^{n} \mu(o_i) = U \tag{1}$$

And the affinity measurement function expresses as:

$$aff(o_i, o_j) = \begin{cases} \dfrac{\mu(o_i) \cap \mu(o_j)}{\mu(o_i) \cup \mu(o_j)} & i \neq j \\ 0 & i = j \end{cases} \tag{2}$$

The definition shows that for any given node set O, the affinity measurement function produces a group of affinity measurement value between each pair of nodes. Under the circumstance of affinity measurement between a node and itself, we assume affinity measurement value aff(oi,oj) = 0. Afterwards, we can construct an undirected and weighted affinity graph G= (O, E, W) with every node in O representing a website and W being the weight value derived by the affinity measurement function on undirected edge set E.

From the affinity measurement function, we can also find that except those affinity measurements between a node and itself, once two nodes have any common user, an edge will be constructed, resulting in the fact that most of the connections are so weak that only exist as redundancy. Based on the situation we sparsify the graph by setting a threshold τ, only these edges with higher weight value than threshold value can be regarded as being exist, otherwise edge value will be set to 0. Subsequently, for the sake of process simplification and efficiency enhancement, we transform the affinity graph G= (O, E, W) into a sparsified affinity graph G'= (O, E'). Amongst the sparsified affinity graph G'= (O, E'), E' is the set of unweighted edges. The sparsification function expresses as:

$$spa(o_i, o_j) = \begin{cases} 1 & aff(o_i, o_j) \geq \tau \\ 0 & aff(o_i, o_j) < \tau \end{cases} \tag{3}$$

From above, an undirected and unweighted edge $e'_{ij} \in E'$ will be constructed if $spa(o_i, o_j) > 0$, otherwise no edges will be constructed. So far, we have generated an ideal undirected unweighted graph containing nodes and edges implicating strong connections.

Feature Learning

From the previous work we have defined and derived a sparse graph containing all the nodes and edges we need, now it is necessary to process the graph for feature learning. Here we mainly focus on the issue of prediction over nodes and edges, more specifically, identifying and classifying communities. According to the notion of a recent research presenting node2vec algorithm, our graph can convert into low-dimensional form(vector) represent for neighboring environment.

The first problem emerged is determining the way to traverse the graph. As we know, nodes in a graph are multi-connected, whereas the pattern of traverse can only be one after another. Traditional graph traversal is categorized into Breadth-first Search and Depth-first Search, but under circumstances like path graph or star graph, BFS and DFS will accordingly be the worst efficient traverse pattern, causing enormous waste of resources. If we model the process of identifying communities as an optimization task, two independent factors conditional independence and symmetry in feature space are focused. On the one hand, conditional independence requires several nodes linking to each other and being in the same one neighboring environment, on the other hand, whether two nodes are linked to the same group of nodes is not necessary when it comes to symmetry in feature space. As long as several nodes are in the same topological environment, they resemble regardless of distance in a network. For instance, if two people share the same interests and disposition, in spite of unacquaintance, they are "symmetric" to some extent. We select node2vec as algorithm for the next step of research, whose characteristic is a compromise between BFS and DFS achieved by biased random walks traverse strategy.

Random Walk

Suppose we know the initial node u and walk length l, the last parameter needed to determine a complete traversal as well as destination node is the transition probability P. In fact, when it comes to learning the entire graph, every node is equally important and supposed to be learned as initial node. Because traversal under random walk are second order Markovian process, the scenario that which direction to go between current node and the next one, will not affected by previous walks. That enables us to compute transition probability in advance and improve time efficiency to a great extent.

We define the ith node in a walk as ni and a normalizing constant C, then nodes in a walk can be expressed as:

$$P(n_i = x \mid n_{i-1} = v) = \begin{cases} \frac{P_{vx}}{C} & (v, x) \in E \\ 0 & otherwise \end{cases} \tag{4}$$

Search Bias α

Intuitively, we are easily associate edge weight w with transition probability P, since edge weight usually comes down to a measurement of connectivity or similarity. Nevertheless, different structure of network has not been considered, also as mentioned above, conditional independence and symmetry in feature space are in coexistence. We need to introduce a bias mechanism which combines thoughts of structural equivalence and homophily, bringing the most efficient possible traverse pattern and extract shared feature of different nodes precisely.

The bias mechanism includes two parameters: Return Parameter p and In-out Parameter q, specific to second order random walks. Suppose a random walk has travelled from node v to node x, the second order random walk to reach node u has a biased transition probability Pxu = αpq(v,u)・wxu. The expression of search bias α is:

$$\alpha_{pq}(v,u)=\begin{cases}\frac{1}{p} & d_{vu}=0\\ 1 & d_{vu}=1\\ \frac{1}{q} & d_{vu}=2\end{cases} \tag{5}$$

Here dvu denote the shortest distance between node v and node u, and we assume all edges has the distance of 1. Note that transition probabilities here must meet:

$$\sum^{u}\alpha_{pq}(v,u)\cdot w_{xu}=1 \tag{6}$$

Thus, the definition of search bias is complete since it covers all three possible situations, and traverse tendency depends on parameters p and q.

Return Parameter p and In-out Parameter q

According to the definition of search bias α, return parameter p and In-out parameter q controls the tendency of random walks to revisit neighboring nodes or traverse outward respectively. For instance, if return parameter p is greater than 1 and In-out parameter q, the random walks are less inclined to revisit a internal nodes. On the other hand, if p is less than 1 or q, the random walks are tend to backtrack. As for In-out parameter q, a high value of q represents unwillingness of traverse further outward, otherwise the opposite. These features are well demonstrated by the expression of α, being appropriate for diverse practical network structures. Because consecutive random walks are always concerning search bias, node2vec is a algorithm considering"cause and effect", such supervision mechanism keeps an equilibrium between BFS and DFS, reaching a status of integrate optimization

Algorithm: The node2vec

Learn Features (Graph G = (V, E, W), Dimensions d, walks per node γ, walk length l, Context size k, Return p, In-out q)

π = Preprocess Modified Weights (G, p, q)
G' = (V, E, π)
Initialize walks to Empty
for iter = 1 to γ do
for all nodes u ∈ V do
walk = node2vec Walk (G', u, l)
Append walk to walks
f = Stochastic Gradient Descent (k, d, walks)
return f
node2vecWalk (Graph G'= (V, E, π), Start node u, Length l)
Initialize walk to [u]
for walk_ iter = 1 to l do
curr = walk [-1]
Vcurr = Get Neighbors (curr, G')
s = Alias Sample (Vcurr, π)
Append s to walk
return walk

Community Identification

Now that we have a set of coordinates representing large amounts of nodes in low-dimensional space, the task of identifying communities remains unsolved. Usually there are two significant factors taken into consideration when it comes to measuring degree of representation between complex structure and low-dimensional graph: connection and distance. Here, the errors arise from dimensional conversion can be ignored, since the parameters of node2vec assigned are tendentious while traversing the graph. With vectors representing nodes existing in the same one coordinate system, we calculate and compare Euclidean Metric of each pair of nodes as speculation of similarity. After manually choosing several hub nodes, we can pick out top-n nodes with minimum Euclidean Distance to form a central-appointed community.

It should be pointed out that even two graphs are the same, they still might be separate into entirely different communities, due to different assignment of parameters. As mentioned above, parameter p and q determine the pattern of traversal. Suppose we set a relatively high value of p and low value of q, random walks are inclined to travel further and collect nodes according to the idea of symmetry in feature space. In view of our practical requirement, we have to modify the value of p lower and the value of q higher. Specific value can be compromised on account of detailed demand.

EXPERIMENT

Community identification has been greatly emphasized in studies of website network, with its wider application as well as heavier tasks, we call for a method that has both high efficiency and excellent accuracy. In order to evaluate our solution, we choose data from top-100 websites with highest page view, after analyzing proportion of shared users and constructing edges without a threshold we derived a 3D graph.

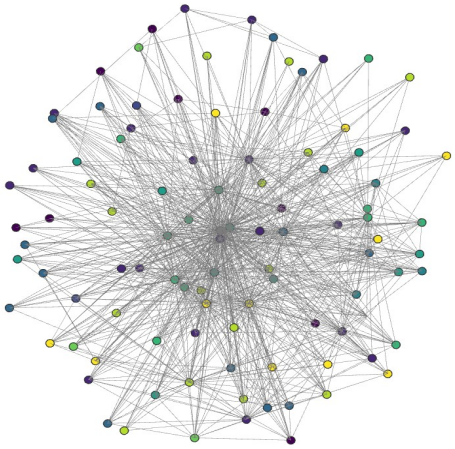


FIGURE 1. 3D graph of 100 websites' network.

As demonstrated in Fig. 1, excessive edges are constructed so the website network remains complicated, which disturbs subsequent process. To simplify the network and extract main information we set a threshold of 0.4, the number of edges drop to 933, which means when the ratio of shared users to users of total users is greater than 0.4, we assume the two websites are related. Under the circumstance average connection of each node is 9.33, or 9.33% websites are regard as closely connected, which is a reasonable result as a sparsified affinity graph.

According to discussion in Feature Learning, when we focus on community identification, we are more interested in nodes in the neighborhood instead of nodes that are far away from hub nodes. Since we intend to backtrack more and traverse outward less, we set p = 0.5 and q = 2, and get result from node2vec in form of coordinates in 10 dimensions for 100 nodes.

We choose 8 websites with most pageviews as hub nodes, which are 'ucweb', 'umeng', 'google', 'flurry', 'apple', 'baidu', 'qq' and 'sina', then calculate their Euclidean Distance to every other 99 nodes respectively. The shorter their Euclidean Distance is, the closer relationship in network space they are, by setting another threshold we can extract a community of any scale. After that we compare community identification result by node2vec with

sparsified affinity graph, if two nodes get together in the same community in node2vec result, in the meantime, statistical proportion of shared users is large enough to exceed threshold and construct a corresponding edge in the sparsified affinity graph, the result is credible. Here we present result for 'apple' as example.

TABLE 2. Result Comparison of Node2vec and sparsified affinity graph(apple).

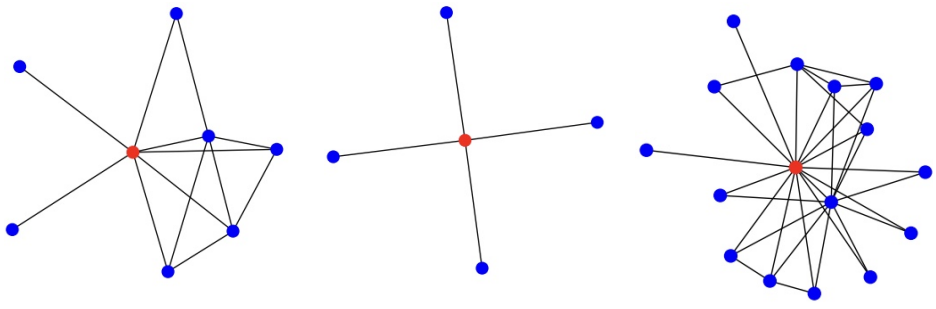
Website	Euclidean Distance to hub node	Edge existence
apple	0	N/A
adsage	0.577906098	Exist
edgesuite	0.949014150	Exist
outfit7	1.043954626	Exist
webscache	1.089156196	Exist
yahooapis	1.468610313	Exist
ppstv	1.469572095	Not exist
cloudfront	1.506743730	Exist
flurry	1.643299023	Not exist
...	...	...

The Table 2 is arranged in an Euclidean Distance descending order. From the result we find that all 6 nodes connected to hub node in sparsified affinity graph are included in top 7 nodes with closest Euclidean Distance, the outcome is satisfactory. Moreover, if we calculate ratio of number of nodes connected to hub node in affinity graph to number of nodes closer than farthest connected node, the result is:

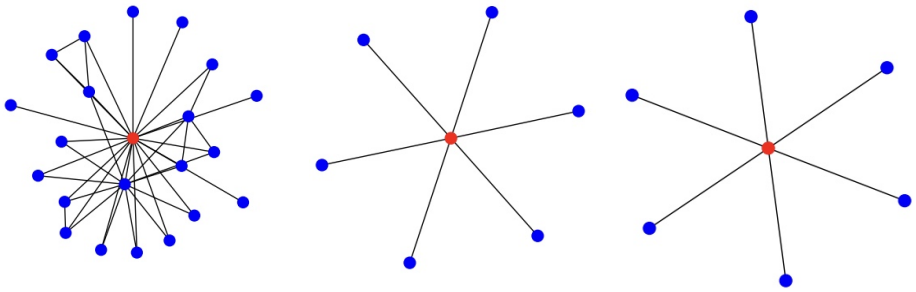
TABLE 3. Accuracy of prediction for 7 hub nodes.

Website	Nodes Connected in Graph	Nodes in Community	Proportion
ucweb	16	20	80%
umeng	6	7	85.7%
google	4	4	100%
flurry	7	7	100%
apple	6	7	85.7%
baidu	23	31	74.2%
sina	2	2	100%

According to Table 3, our method performs well in community identification especially when the community is small and centered and has an average prediction accuracy of 89.4%. We also draft graph examples demonstrating community structure for some websites.



(a) (b) (c)



(d) (e) (f)

(a) for ‘flurry’, (b) for ‘google’, (c) for ‘ucweb’, (d) for ‘baidu’, (e) for ‘apple’, (f) for ‘umeng’.

FIGURE 2. Graph examples of community structure.

CONCLUSION

The exploration of website community will never come to an end, there are still some lot characteristics for us to discover, even more solutions to cope with increasingly complex problems, more methods to better understand the Internet world. In this paper, we develop an integrated solution with regard to identifying website communities. First of all, we constructed a sparsified affinity graph after computing raw data and artificially setting a suitable threshold. Then we apply the node2vec algorithm to transform our graph into low-dimensional space, during which we modify parameters to conform to requirement. At last we pick out several hub nodes to form communities. The experiment proves that our solution has a remarkable performance in identifying communities, meanwhile, a comparison of experimental and statistical results demonstrates availability and reliability of our solution.

REFERENCES

1. Cision Inc., 2017 global social journalism study, 12 Sep 2017.
2. A. Grover and J. Leskovec, node2vec: Scalable Feature Learning for Networks, ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2016.
3. R. Kumar and P. Raghavan, Trawling the web for emerging cyber-communities, Computer Networks, 1999, 31 (11–16) :1481-1493.
4. D. Gibson and J. Kleinberg and P. Raghavan, Inferring web communities from link topology, Proceedings of the 9th ACM conference on Hypertext and hypermedia, 1998, pp. 225-234.
5. J. Kleinberg, Authoritative Sources in a Hyperlinked Environment, IBM, 1998.
6. P. K. Reddy and M. Kitsuregawa, An Approach to Relate the Web Communities through Bipartite Graphs, WISE, Kyoto, Japan, 2001, pp. 0301.
7. N. Imafuji and M. Kitsuregawa, Finding Web Communities by Maximum Flow Algorithm Using Well-Assigned Edge Capacities, IEICE Trans. Inf. and Syst, 2004.
8. G. W. Flake and S. Lawrence and C. L. Giles, Efficient identification of web communities, Proceeding of the 6th ACM SIGKDD international conference on Knowledge discovery and data mining, 2000, pp. 150-160.
9. F. Ricca and P. Tonella, an empirical study on keyword-based website clustering, Proceeding of the 12th IEEE International Workshop on Program Comprehension, 2004, pp. 204-213.
10. H. P. Kriegel, M. Schubert, Classification of Websites as Sets of Feature Vectors, Databases and applications (2004) 127-132.
11. J. Liu and N. Ansari, Identifying Communities in Mobile Internet based on Affinity Measurement, Computer Communications, 2014, 41 (4) :22-30.