2020

Detection of Weak Ties

Jerry Scripps
Grand Valley State University, scrippsj@gvsu.edu

Detection of Weak Ties

Jerry Scripps
Grand Valley State University
Allendale, Michigan

ABSTRACT

In a small world social network, strong and weak ties exist that define tightly clustered areas and isolated links between them. Granovetter provided an heuristic that strong links have a higher common neighbor count than weak ones. The problem of identifying weak links is the central focus of this paper. The proposed metric *vett* will be shown that within certain constraints of a (modified) small world network the accuracy of the proposed method is 100%.

ACM Reference Format:
Jerry Scripps. 2020. Detection of Weak Ties. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In his 1973 paper, Granovetter describes the advantages of weak ties [14]. In a social network, weak ties are the isolated relationships that people have with others outside of their tightly connected groups. Although it is not defined objectively, the author gives us the heuristic that a strong tie will have more common neighbors than weak ties. The problem of identifying strong/weak links is the central focus of this paper. The metric *vett* is proposed to help with identification.

Strong and weak links are an intregal part of many networks. In social networks, sociologists refer to two basic human needs that drive new link exploration: the safety drive, for a tight group of close friends (strong links) and the effectiveness drive, to find new fiendships beyond one's close friends (weak links) [16]. Small world networks [31] have two important properties (high clustering and short average path length) exhibited by many networks found in nature. In their work, strong links from regular lattices and weak, random links, create these two properties.

To show the effectiveness of *vett*, it is useful to make a modification of the small world model. A network is defined with tightly connected, dense which are loosely connected to each other. The dense areas will have links that have a higher common neighbor value than the links connecting the dense area. This model fits within the spirit of the small world model but conforms to Granovetter's definition of strong and weak links.

In the modified small world network, it will be shown that if the dense area has links with $c > 2/3d$ then *vett* will predict the strong links with 100% precision. Also within specific values of p, (probability of placing a random link), weak links can also be identified with 100% precision. A naïve solution would be to find the right common neighbor threshold to separate the two groups. However, because of preferential attachment [2] some nodes will have a higher degree than others and some clusters will be larger than others. To effectively adjust to these conditions, the metric is necessarily defined to be sensitive to the global and local surroundings.

It is important to identify strong and weak links for many reasons. First, of course are the reasons set out in Granovetter's paper. Social media applications have helpful suggestions for its users such as finding new friends. They could also help users leverage their weak links for finding a job.

Second, like so many other metrics, strong and weak links can help to describe a network. They can be used to identify roles (users with many weak links, bridges, etc.) They could also be used to identify pathways of information flow.

Third, strong and weak links can be used to help with other link mining tasks, such as link prediction, ranking, link-based classification, influence maximization and community detection. Due to space limitations, this paper will show results only for using *vett* to help find communities. When a given network is suspected of having communities, it is natural to think of the links within the communities as strong and those between the communities as weak. It will be shown in the experiments section, that removing weak links can expose communities and be a preprocessing step to finding them.

After this introduction, there is a literature review. The definition of terms and the growth model follow that. The Section 4 describes the metric and algorithm to find and show mathematical support for its effectiveness within the defined model. Supporting experiments follow and concluding remarks follow that.

2 RELATED WORK

While Granovetter may have been the first to define strong/weak ties he was not the last. This section separates the previous work into 3 groups: definitions that use only the graph, definitions that make use of data from outside the graph and predicting tie strength.

2.1 Definitions that use the graph

Some studies define strong/weak links in terms of the strong triadic closure rule [28] [27]. This rule states that open triangles of two strong links cannot exist. This approach optimizes a different graph property than the present work; it will be shown in the experiments that the two approaches are not similar. Jaccard is used to approximate tie strength in [30]. They then use a Baysian Personalized Ranking system to learn the threshold used to separate strong links from weak. There is an example using the Football network in the Section 3 that illustrates the pitfall of using a threshold. Signed networks are used in [18] to predict missing link labels (+/-), however, the method in this paper, does not assume that label information is available.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

2.2 Use of data outside the graph

Since this paper relies only on data from the graph the following work is presented here only for completeness. Some proposed using a mobile call network to determine the length of time two people talked [22] and using relationship length and strength [32]. Similarly, strong/weak ties can be defined by online behavior and demographics [15]. Another study makes the assumption that strong ties are those within communities while weak ties are between them [8]. In [26] the information from one graph is used to predict the strong/weak links in another. In [3] a convolutional neural net is used with node attributes to predict strong/weak ties. Random forests and linear regression are used to predict tie strength using Jaccard in [21]. Both [19] and [5] rely on weighted graphs to predict tie strength.

2.3 Predicting tie strength

Some papers measure tie strength based on external factors such as time, depth of relationship and other quantitative and qualitative data: [20] [32] [11] and [15]. In other works, additional information can be gathered from surveys [9] or transactions [17]. A few papers make comparisons of weak links to other phenomenon. The authors in [22] show that there is a relationship between tie strength and community bridges. In [10], they show the relationship between links in facebook and social capital. While it is valuable to be able to predict tie or link strength, the purpose of this paper is to identify strong and weak links.

3 DEFINITIONS

A network $G = (V, E)$ is a system of nodes $V = \{v_1, \ldots, v_n\}$ which are connected to each other by links $E \subset V \times V$. An adjacency matrix $A = [a_{ij}]_{n \times n}$, is used to represent the links, where link $a_{ij} = 1$ for every link $e_{ij} \in E$ between v_i and v_j. This paper also uses a weight matrix $W = [w_{ij}]_{n \times n}$ that represents the link weights (or tie strength). In this paper, d_i represents the degree for v_i and c_{ij} represents the common neighbors for v_i and v_j.

3.1 Defining weak ties

Granovetter [14] discussed social networks in which each pair of nodes (actors) are connected by a weak tie (link), a strong tie or are not connected at all. His subjective definition is that a strong tie will have more common neighbors than a weak tie. This is the motivation for the metric proposed in the next section. While not an objective definition it does give some direction. While the circumstances may change what we consider to be a weak tie, a higher common neighbors measurement indicates a stronger tie. Links between are either strong or weak and observers can see the difference with smaller networks. With larger networks, it is not possible to manually assess each link as strong or weak.

3.2 Small worlds

The small world model [31] (which will be referred to as SW1) defines a network as a combination of a regular and random network. The regular links provides tight clustering, as measured by the clustering coefficient and the random links give it short traversal paths, as measured by the average geodesic path between all node pairs.

SW1 networks reflect Granovetter's vision of strong and weak links. The (strong) links that are part of the original regular network

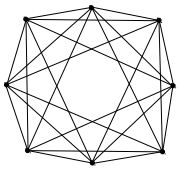

Figure 1: Example of community of 8 nodes, each with degree of 6 and 24 links, each with c=4

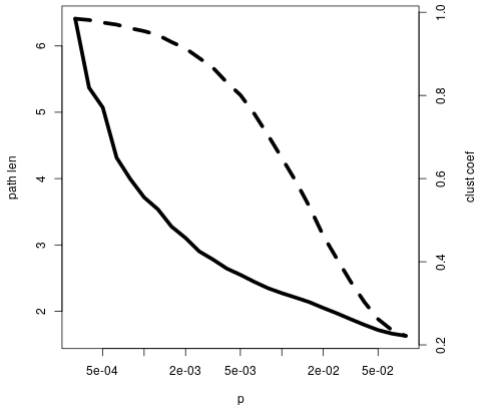

Figure 2: Example graph of clustering coefficent plotted along with average geodesic path length in SW2 networks.

are there to provide a high level of clustering. The (weak) random links provide a shortcut path from one part of the network to a previously distant part of the network. The SW1 growth model does not enforce a high c value for strong links though. For example, consider the circular lattice where each node is connected to its 6 closest neighbors. Some of the links will have c=4, while others will have c=2. As the number of connections get higher, the variability in the c will also get larger.

A change to SW1 will make c more uniform for strong and weak links while still conforming to the general concept behind small world. Instead of having a lattice as the starting point for the strong ties, SW2 will create a number of communities with equal numbers of nodes and connect the nodes so that each one has a similiar degree and each link has an approximate, minimum c (see Figure 1).

In SW2 like SW1, random links (weak ties) are created between nodes in different communities based on a probability p. The random links are created by applying the probability p to each possible weak tie (those between communities). Figure 2 shows the average path length and clustering coefficient plotted for networks of 20 communities of 20 nodes each, with each node having a degree of 15, and various values of p. Notice that the area that defines a small world network (high clustering coefficient, low avg path length) is similar to the original paper from [31], in the range $0.001 \leq p \leq 0.02$.

Table 1: Common neighbors for strong and weak links in football

cn	within	between
0	0	83
1	0	46
2	5	20
3	8	3
4	64	1
5	108	0
6	115	0
7	94	0
8	10	0
9	0	0

3.3 Thresholding limitations

To identify strong/weak links, one could simply choose a threshold $\hat{c}$ and label any link with $c_{ij} > \hat{c}$ as strong and those with $c_{ij} < \hat{c}$ as weak. Of course things are not that easy. Different parts of the network are more dense than others. The chosen threshold might work for some parts of the network but not others. For a concrete example, consider the football data set [12], where each of the 115 nodes represents a US college football team and the links represent the games. Most of the teams belong to a conference (of about 6 to 13 teams). Teams mostly play other teams in their conference (strong links) but they also play a few teams from other conferences (weak links).

Table 1 contains a cross-tabulation of the links for the football network by common neighbors and link type (within or between). This tabulation appears to be reasonable with many of the (strong) within links having high values of c and low values of c for the (weak) between ones. However, even with this nearly ideal SW2 data set, there is no threshold that would neatly separate the strong from the weak links. Notice that for $c = 2$, most of the links are between communities (conferences) but there are some where 2 teams in a large community have only 2 common neighbors.

3.4 Problem definition

The problem is to identify weak ties in SW2 networks given Granovettor's definition of weak ties and in the presence of networks of varying densities. The metric c is the basis for the definition of weak ties, but was shown to be insensitive to its local and global surroundings. Jaccard is a normalization of c, $J_{ij} = N(v_i) \cap N(v_j)/N(v_i) \cup N(v_j)$, where $N(v_i)$ is the neighborhood of v_i, is sensitive to its local surroundings but not global.

PageRank [23] used global information to solve the problem of the assigning an authority rank to web pages. A page that has hyperlinks from highly ranked nodes will be ranked higher than one with lower ranked ones. The rank of a node depends on the number of neighbors but it is also the rank of those neighbors. The PageRank vector is defined in terms of itself (using an eigenvector formulation), where the rank of a node is circularly defined by the ranks of its neighbors.

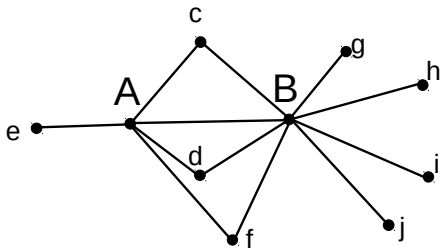

Figure 3: nodes A and B with neighbors

In Figure 3, the same concept is illustrated for links. The strength of the link e_{AB} can be measured by the number of common neighbors (which is 3) or the Jaccard metric (which is $3/8$). The metric proposed in the next section, considers not only the links to common and other neighbors but also, the strength of those links. Like PageRank, the metric will define the strength of a link in terms of the strength of its adjacent links, where the weights diffuse globally.

4 DETECTION PROCESS

4.1 Detection using undirected weights

This section introduces a new metric, *vett*, that is designed to identify weak and strong ties. There are two formulations, one using undirected weights ($w_{ij} = w_{ji} \forall i, j$) and the other using directed weights. Undirected weights are considered first. The definition for *vett* is shown in Equation 1.

$$w_{ij} = \frac{w_{ij} + \sum w_{ik} w_{jk} a_{ij}}{w_{ij} + \sum (w_{ik} w_{jk} a_{ij} + w_{ik}(1 - a_{jk}) + w_{jk}(1 - a_{ik}))} \tag{1}$$

The symbol $\sum$ means $\sum_{k=1}^{n}$.

The new value is assigned to the variable w_{ij} on the left with the existing values of W used in the expression on the right. The numerator is the existing value of the link weight, w_{ij}, plus the product of the weights of the two links for each common neighbor of v_i and v_j. The denominator is the value of the numerator plus the sum of the link weights from v_i and v_j that are not connected to a common neighbor. $\forall i, j : 0 \leq w_{ij} \leq 1$. Links with no common neighbors will approach 0 while those with no neighboring links not connected to common neighbors will approach 1.

The process for calculating W is the power method, setting all of the weights $w_{ij} = a_{ij}$. Then each weight is iteratively recalculated from the prior values - updates are not done until all of the weights have been recalculated. The process stops when the weight values have converged. Weights for any node pair that does not have a link are always zero.

4.1.1 Detecting strong links in SW2. Here it will be shown that $w_{ij} > 0.5$ when w_{ij} is a within community link in an SW2 network and when c is sufficiently large. It will also be shown that it is very likely that $w_{ij} < 0.5$ when w_{ij} is a random link in an SW2 network for reasonably small values of p.

Recall that in SW2, nodes and links are placed into communities before the random links are added. These are all strong links and the

random ones are weak links. Before the random links are added, the weights for links should all be equal, since links all have the same value for c and all of the nodes have the same degree. Substituting into the original formula leads to:

$$w = \frac{w + cw^2}{w + cw^2 + 2(d - c - 1)w}$$

Multiply both sides by the denominator,

$$w^2 + cw^3 + 2(d - c - 1)w^2 = w + cw^2$$

and divide by w, and then multiplying and combining terms yields

$$cw^2 + (2d - 3c - 1)w - 1 = 0$$

which is a quadratic equation. Using the quadratic formula, two possible solutions for w can be found. For values of $w > 0.5$ (the indication of a strong tie):

$$\frac{-(2d - 3c - 1) \pm \sqrt{(2d - 3c - 1)^2 + 4c}}{2c} > 0.5 \qquad (2)$$

First examine the $+$ root of Equation 2. After multiplying by the denominator, rearranging terms and squaring both sides:

$$-2dc + 3c^2 + 5c > 0$$

and finally

$$c > \frac{2d - 5}{3} \qquad (3)$$

Next examine the $-$ (negative) root of Inequalty 2. After multiplying by the denominator, rearranging terms, squaring both sides and again rearranging terms it becomes

$$-8d^2 + 10dc + 4d - 13c^2 - 9c - 2 > 0$$

This root is quadratic in c. It can be shown that finding a solution for c using the quadratic formula leads to answers that are imaginary numbers. Therefore, Inequality 3 is the only solution for which c can be calculated given d. Note that this implies that for our metric to effectively identify strong ties, the common neighbors of the initial communities must be greater than $2/3$ of the degree, for large values of d.

All of the work done in this section so far assumes that there are no random links. Adding random links will change things, which will be addressed in Section 4.1.3.

4.1.2 Detecting weak links in SW2. The random links are placed between the communities after the communities are created as described above. The question to be answered here is, will *vett* be able to identify these weak ties? Remembering that common neighbors is the driving force behind the metric, it will be shown that there is a low probability that c for a random link will become large enough for *vett* to mis-identify the link as a strong tie.

There are two ways for a random link e_{ij} to have a common neighbor, say node v_k. The first is for e_{ik} to be a within community (strong) link and for e_{jk} to be another random link. This forms a triangle of a weak, another weak and a strong link (WWS). The other way is for all three links to be weak (WWW). WWS is considered first.

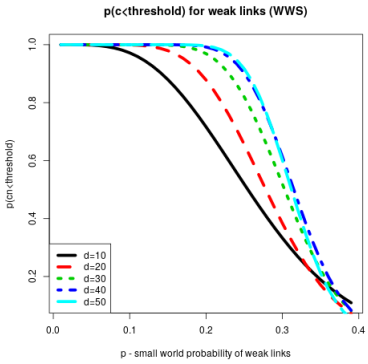

Figure 4: Probability of common neighbors metric being below SW2 threshold ($\frac{2d-5}{3}$) for triangles involving a strong link

Begin with a simple example - placing the first random link. Since this is the first one, there are no other links between communities, so c should be zero. With zero common neighbors, the value of *vett* will approach zero. To expand this to the general case (placing subsequent links), a way is needed to calculate the probability of the number of common neighbors for a random link.

Considering a random link e_{ij}, to have a common neighbor using WWS it would have to have another random link connecting v_i to one of v_j's strong neighbors or a random link connecting v_j to one of v_i's strong neighbors. Using the binomial distribution to calculate the probability of having exactly k common neighbors follows:

$$p(c = k) = \binom{n^*}{k} p^k (1 - p)^{n^* - k}$$

where $n^* = 2(d-1)$ and d is the degree of each node before adding the random links. Figure 4 shows the probability of the common neighbors metric being less than the threshold (Formula 3). The chart is based on a networks with $n = 1000$ nodes, varying the degree from 10 to 50 and using different values for the small world probability of placing a weak link. Recall that with our definition of small world networks, the area for the small world effect is between .001 and .01. Changes to n does not affect the shape of the curve.

WWW is similar but leads to a different outcome. Given a weak link e_{ij}, to calculate the probability for the single common neighbor v_k, it is necessary to multiply the probability of links e_{ik} and e_{jk} where v_i, v_j and v_k are all in different communities. This also is a binomial distribution with different parameters:

$$p(cn = k) = \binom{n^*}{k} q^k (1 - q)^{n^* - k}$$

where n^* is $n - 2d$ and $q = p^2$. The plot in Figure 5 shows that with a low value of p, the probability is very high that the common neighbors will be less than the threshold. As p increases it becomes less likely. It is hardly noticible with networks $n < 10,000$ nodes. However, it becomes increasing more profound as n gets larger. With $n > 90,000$

In this analysis, the number of communities is kept constant at 100 and the degree constant at 10. Changing the community number

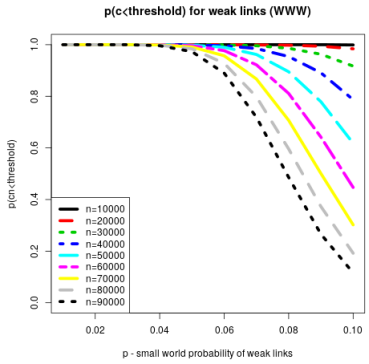

Figure 5: Probability of common neighbors metric being below SW2 threshold ($\frac{2d-5}{3}$) for triangles involving 3 weak links

could make a slight change while, increasing the degree would also increase the threshold c which should improve the likelihood as well. The point of this section is that optimum results are guaranteed for some large networks and most smaller ones. Experimental results in the next sections support this claim but go further to show that good results are possible for many small world networks.

4.1.3 Effect of random links on strong links in SW2. In the section above on detecting strong links, the analysis was based on a network of only strong links, before adding in random links. It is natural to wonder if the random links will influence the analysis. The answer is that if there are just a few weak links that are low weighted, it should not have a profound effect on the analysis. If these random links are indeed weak – that is, they have few if any common neighbors with the node in question – then they should have a low weight. However, as p increases, then too, will c for the random links increase.

One can imagine a network where p is large enough to make strong and weak links indistinguishable from each other because they all have large values of c. Looking at Figures 4 and 5, at the point where this happens p would become large enough for the network to no longer be considered to be a small world network.

4.2 Detection using directed weights

The *vett* metric sums the non-common neighbors in the denominator. A node with a large degree can skew the results, making a strong connection appear weak. To accomodate the imbalance, *vett* can be changed from an undirectional to a directional metric. For each link there are two formulas based on direction:

$$w_{ij} = \frac{w_{ij} + \sum w_{ik}w_{jk}a_{ij}}{w_{ij} + \sum w_{ik}w_{jk}a_{ij} + w_{ik}(1 - a_{jk})} \tag{4}$$

$$w_{ji} = \frac{w_{ij} + \sum w_{ik}w_{jk}a_{ij}}{w_{ij} + \sum w_{ik}w_{jk}a_{ij} + w_{jk}(1 - a_{ik})} \tag{5}$$

In the directional formulas, a link between v_i and v_j would have two weights. A simplistic way to think of it is that w_{ij} is a measure of how important v_j is to v_i and w_{ji} is a measure of how important v_i is to v_j. There are two ways a nodes non-common neighbor links can influence the metric - the number of links and their weight. So it is very possible that if v_i has a very high degree and v_j is very low, w_{ij} indicates a weak link and w_{ji} indicates a strong link.

While applications can use the directional weights as the programmer sees fit, many will wish to create an undirectional network by removing either the strong or weak links. In the experiments, when removing weak links, if either was strong, the link was considered strong. Using these new formulas, it is possible to do the same analysis performed above on a SW2, small world network. Most of the analysis is the same except for calculating the value of common neighbors with respect to degree. The new formulas result in the following threshold for c:

$$c > \frac{2d - 4}{3} \tag{6}$$

which is nearly the same as the undirectional *vett*. So one could expect the same level of precision with respect to identifying strong/weak links.

4.2.1 Practical considerations. The analysis in this section assumed networks conforming to the SW2 definition. That allowed for specific claims of accuracy under given circumstances. Some real networks may behave according to the SW2 model but of course, some may not. That does not mean that *vett* will not be helpful but only that it is not guaranteed to be highly accurate. In Section 5, tests will be presented to show the effectiveness of *vett* under different circumstances.

Another practical consideration is the choice between directed and undirected *vett*. The main difference is that directed *vett* assumes that every node belongs to one and only one community. Consider a node with many links to other nodes that are all linked to each other. These links will be identified as strong by *vett*. If that node has a few other links to other non-connected nodes, they will be identified as weak. Consider though, a node that has 6 links connected to a connected group of 6 nodes and then 5 other nodes connected to a different group of 5 connected nodes. The 6 nodes will be identified as strong while the 5 will be identified as weak. Using directed *vett*, all 11 links would be identified as strong. The choice depends on whether one wants the metric to recognize nodes with overlapping community membership or not.

The last practical consideration is the complexity of *vett*. Assuming that the network is represented using an edge list, *vett* needs to make I iterations before it converges. In each iteration, the weight for each node is recalculated by tracing the neighbors of each of its neighbors. Using a adjacency list, the complexity is $O(I \times n \times d^2)$

5 EXPERIMENTS

Testing the effectiveness of *vett*, could be done with a network where the links were labeled with strong and weak links. The authors are not aware of any such test bed of network data. Even if there were, it might not be helpful if the designations were based on something outside of the network. If a network was built based on users descriptions of the friends as strong or weak relationships, it might not coincide with the use of common neighbors. So as an alternative, experiments are presented to demonstrate the usefulness of *vett* using the concept that strong links are within communities and weak ones are between them.

Table 2: List of datasets

dataset	n	links	clustcoef	avgPathLen	p
football	115	613	0.40	2.49	0.034
jazz	198	2742	0.63	2.22	0.097
revere	254	9706	0.94	1.69	0.073
usAir	332	2126	0.75	2.73	0.039

5.1 Data sets, algorithms and methods

Example data sets based on real networks were used to give a visual depiction of separating strong links from weak links. The data files are listed in Table 2. All are undirectional. They are intentionally small so that the reader could see the effects in a two-dimensional plot. Notice that all appear to be categorized as small world networks having a high clustering coefficient and a low path length. The last column is a rough, approximate p value, calculated retroactively. It was derived by dividing the number of links that were identified as weak by the total number of possible links minus the number of strong links. It is really a highest possible p value.

To show the effectiveness for community finding and for aiding other community finding algorithms, synthetic networks were created. The synthetic networks were generated both by a generator written by the authors (SW2g) and by the LFR benchmark [1]. In all experiments, for each setting, 10 networks were generated with the results averaged.

SW2g randomly generates k communities of nn nodes with a degree of d. With a large enough degree the communities have mostly strong links with a high common neighbor value. Then the algorithm randomly places links between the communities with a probability of p. For the LFR benchmark, networks were generated using the parameters: the number of nodes (N), the average degree (k), the maximum degree ($maxk$) and the mix (mu). The mix is the percent of inter-community links. Networks were created by varying the number of nodes N.

To compare the results with communitying finding algorithms, igraph [7] was used. Of the algorithms available, we chose fast-greedy [6], label propagation [24] and modularity-multilevel optimization [4]. These were chosen to provide a variety of different methods of community finding.

In the tests comparing *vett* with the community finding algorithms, ten networks were generated according to the parameters. Then, separately communities were found using the algorithms identified above, and links were identified as strong or weak using *vett*. Finally, the statistics were gathered and accuracy was calculated using each link. Both LFR and our own network generator output the "ground truth" community that is used in creating the network. True positives are when a link is in the same community from both the generator and the algorithm (or labeled "strong" using *vett*). True negatives are links generated from different communities that are in different communities for the algorithms (or labeled "weak" using *vett*).

The process for using the metric for preprocessing is essentially the same as the previous paragraph except that after the network is generated, *vett* is used to remove the weak links from the network before it is processed by the algorithms. The accuracy is then compared to the accuracy to using the same algorithm without the preprocessing.

5.2 Visualizing small networks

This section simply displays the network plot of each of the data sets next to the network plot with the weak links removed. With all of these networks, some care was taken to group the nodes to reveal the communities within.

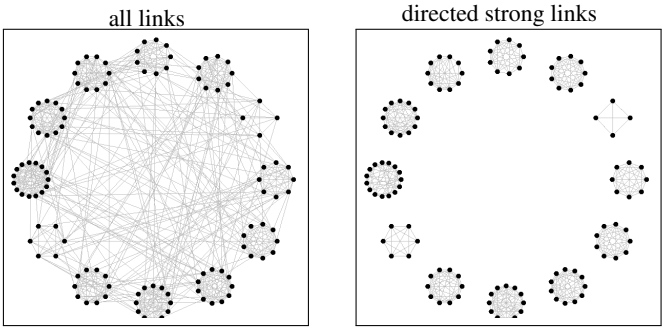

Figure 6: football data set

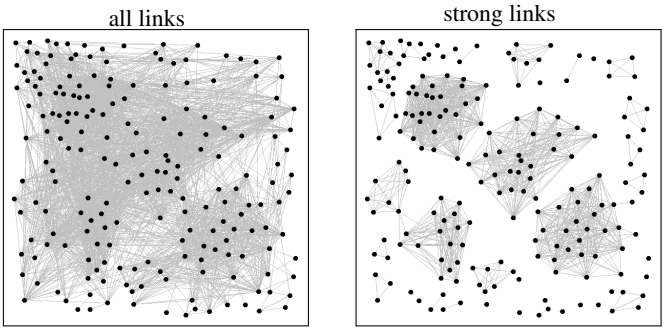

Figure 7: jazz data set

The football set [12] is a the same set described in Section 3 This is a good example of an SW2 small world network. Using directed *vett*, all of the intra-conference games are labeled as weak links and all of the inter-conference games are labeled as weak links (see Figure 6). Of the independents, 4 formed their own community and the other 2 were absorbed into the conferences with whom they had the most games. The undirected plot is not shown as it is similar to the directed plot except a few of the intra-conference links were identified as weak.

In the Jazz set [13] the nodes represent 198 early twentieth century jazz bands. It was build from a currated web site of bands and musicians. The links between bands indicate that a musician played in both bands. This is a very dense data set so that the communities are not visually clear without removing the weak links.

The image in Figure 7 shows the network with all links and just those undirected strong links. Since we did not have access to the original data files, we attempted to reconstruct the descriptive attributes for the nodes from the archived website. It appears that the groups are somewhat geographical. This seems reasonable, that in a time when transportation was more difficult, musicians would belong to different groups that were geographically close.

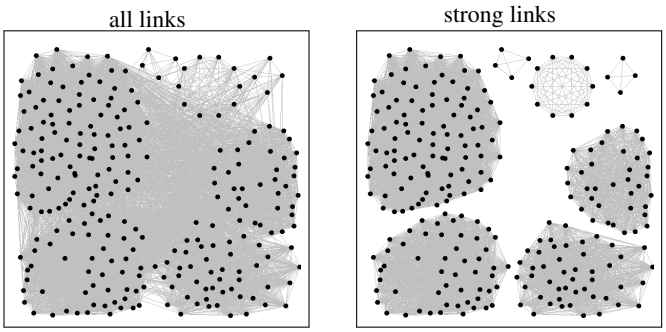

Figure 8: revere data set

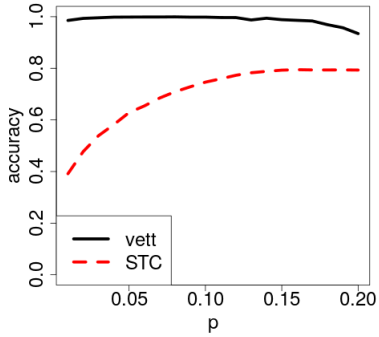

Figure 9: Comparison of *vett* with the greedy STC metric

The revere set [25] is a social network of the early members of the movement in the US to separate from Great Britain. The network was constructed from historical documents about the activities of seven organizations that discussed revolution. The 254 nodes represent the patriots that were at the meetings and links are drawn between any two patriots that belonged to the same organization.

Because so many patriots belonged to more than one organization, the network is very dense. The plot in Figure 8 on the right, shows the undirected strong links, which is just seven cliques representing the seven organizations. Patriots that belonged to more than one organization are placed in the largest organization to which they belonged (recall that undirected *vett* tends to identify as weak, all those links from a node that do not belong to the largest group).

5.3 Community detection

The following experiments use *vett* to detect communities so that it can be compared to community finding algorithms. It should be recalled that the purpose of *vett* is not to find communities but to identify strong/weak links. However, for networks that generally follow the principal of strong links within communities and weak links between them, *vett* can be an effective method of finding communities.

Two preliminary experiments are presented first to show the difference between *vett* and another metric and to show the reason accuracy was chosen. In the first experiment, *vett* is compared to the greedy algorithm from [28]. The greedy algorithm identifies

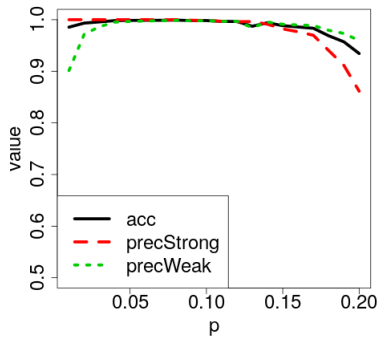

Figure 10: Comparison of accuracy with precision

strong/weak links by minimizing violations of strong triadic closure rule. As mentioned before, since this represents a completely different way to identify links it should not be expected that they work in the same way. Experiments were carried out using both network generators. Figure 9 shows the results using SW2g where it can be seen that greedy has a much lower accuracy than *vett*. Notice that as p increases the accuracy improves for greedy. This is because it minimizes the weak labels, so when the random links are sparse, it is possible to label them all as strong. But as p increases, the random links become denser, so greedy labels more as weak, leading to better accuracy. Importantly, within the region of small world networks, *vett* significantly outperforms greedy. The results from the LFR benchmark were similar and so are not included.

Accuracy was chosen for the experiments below because with community detection, it is equally important to identify strong links as it is weak ones. It is instructive, though, to observe the precision of strong, precision of weak and accuracy. In Figure 10, again, the SW2g generator was used to create networks with p varied from 0.01 to 0.20. Notice that accuracy starts out at about 0.99, rises to 1.0 then drops again as p gets to about 0.18. Looking at the precision lines helps to explain. The precision for strong links is 1.0 until about 0.15 where it drops. This happens because with the many random links the c value gets larger for some of them causing the weak links to be labeled as strong.

The precision for weak links starts below 1.0, goes up to 1.0 and then drops again as p gets larger. When the network has very few or no random links, *vett* identfies some strong links as weak. Recall that *vett* uses global information and if there are no weak links, it will assume that some of the strong links within the communities are weak because they have slightly lower c values. The precision drops again because the many additional random links occassionally cause a node to have more random links than ones in the community causing *vett* to mislabel the strong links as weak.

The rest of the experiments in this section will compare *vett* to the three community finding algorithms (fast, multi, label) described above. The first plot of Figure 11 shows networks created with SW2g. The lines show the accuracy for the different algorithms for networks with differing values of p. With small values of p, it is not surprising that all of the algorithms perform well as the network is nearly just a set of tightly connected groups with only a few links between the

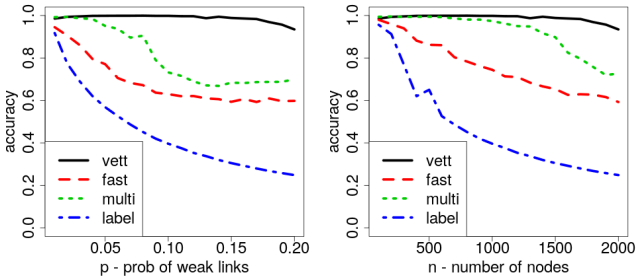

Figure 11: *vett* compared to community finding algorithms

groups. As the value of p grows larger though the accuracy goes down for the three algorithms whereas *vett* stays relatively high until $p > .18$ where it begins to drop.

In the other plot, the algorithms are compared using the LFR benchmark generator. It shows networks where the parameter N varies from 100 to 2000 nodes ($k = 10$ and $mu = 0.3$). Once again, using *vett* results in very high accuracy. The accuracy starts to drop as $N > 1700$. This is more a result of increased density rather than sheer number of nodes. It is thought that increasing $maxk$ along with N will keep the density constant but the parameters in the experiments were not tuned precisely to do that, so the density increased with N.

5.4 Preprocessing

The next experiments show that *vett* may be helpful as a preprocessing step for other analysis techniques. In these experiments the results of finding communities on networks before and after removing the weak links, were compared.

Figure 12 (left side) shows the results for each of the algorithms in three different plots using SW2g generator and varying p. It can be seen that using *vett* is helpful for each algorithm to a point. At about $0.10 < p < 0.12$ the effectiveness of using *vett* drops. This is the point where the additional random links becomes dense such that more of the between links become strong. As they become stronger, the distinction between strong and weak becomes smaller and more links within the community become weak. The plots on the right side of Figure 12 show similar experiments where *vett* is helpful to a point but where the distinction between strong and weak become narrower, its helpfulness becomes less. These experiments used the same LFR benchmark parameter values used in the in the previous experiments.

This section ends with one last note on using real data sets for experiments. Networks with ground-truth communities available from known sites [29] were also used for experiments. However, in the experiments, the accuracy of *vett* was not very high. It appears that the reason for this is that common neighbors c appears to be randomly distributed over links within and between the communities. This goes against the basic premise of *vett*.

6 CONCLUSION

This paper introduces a new metric, *vett*, for identifying strong and weak links in networks. It is shown to have very high accuracy

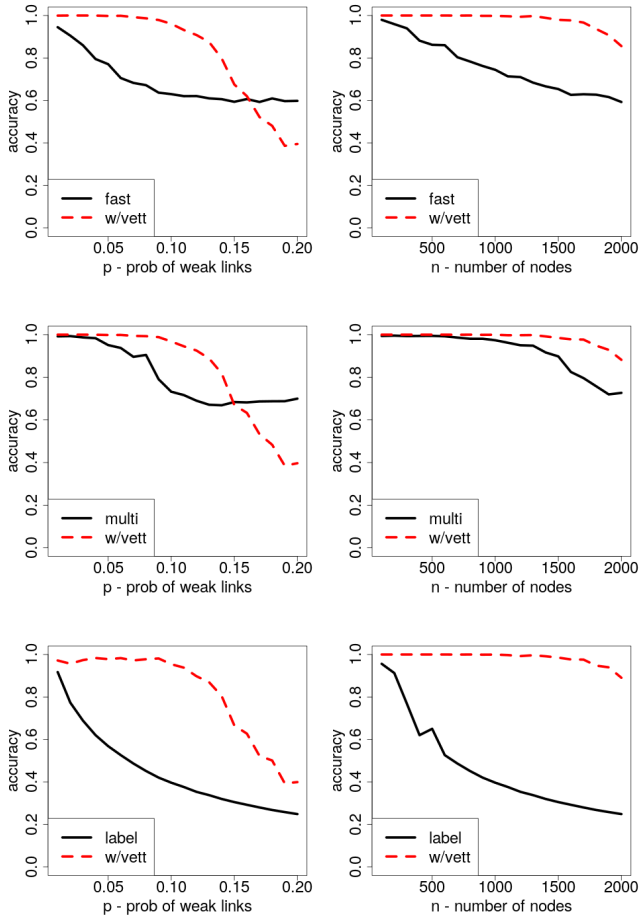

Figure 12: Comparison of algorithms using SW2g(left) and LFR (right) generated networks.

for specific networks in the spirit of the Watts and Strogatz small world model. In a modification of that model, the accuracy is at or near 100% when the common neighbors value for links within networks is $c = (2d + 5)/3$ and $0.001 < p < 0.2$. Even for networks that are not within those ranges, experiments have shown that it can be very effective. The experiments show that *vett* can be used to find communities or as a preprocessing step to finding communities. Other uses seem possible and will be the subject of ongoing research.

REFERENCES

[1] A.Lancichinetti, S. Fortunato, and F. Radicchi. 2008. Benchmark graphs for testing community detection algorithms. *Phys. Rev. E* 78 (2008).
[2] R. Albert and A. Barabási. 2000. Topology of evolving networks: Local events and universality. *Physical Review Letters* 85 (2000), 5234–5237.
[3] Neda Bidoki, Alexander Mantzaris, and Gita Sukthankar. 2020. Exploiting Weak Ties in Incomplete Network Datasets Using Simplified Graph Convolutional Neural Networks. *Machine Learning and Knowledge Extraction* (Jun 2020).
[4] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* 2008, 10 (Oct 2008).

[5] Jie Chen and Ilya Safro. 2009. A Measure of the Connection Strengths between Graph Vertices with Applications. *arXiv* (2009).

[6] A. Clauset, C. Moore, and M. E. J.Newman. 2006. Structural Inference of Hierarchies in Networks. In *Proceedings of the 23rd International Conference on Machine Learning (ICML), Workshop on Social Network Analysis*.

[7] Gabor Csardi and Tamas Nepusz. 2006. The igraph software package for complex network research. *InterJournal* Complex Systems (2006), 1695.

[8] Pasquale De Meo, Emilio Ferrara, Giacomo Fiumara, and Alessandro Provetti. 2014. On Facebook, Most Ties are Weak. *Commun. ACM* 57, 11 (2014), 78–84.

[9] Nathan Eagle, Alex (Sandy) Pentland, and David Lazer. 2009. Inferring friendship network structure by using mobile phone data. *Proceedings of the National Academy of Sciences* 106, 36 (2009), 15274–15278.

[10] Nicole Ellison, Charles Steinfield, and Cliff Lampe. 2007. The Benefits of Facebook Friends: Social Capital and College Students Use of Online Social Network Sites. *J. Computer-Mediated Communication* 12 (07 2007), 1143–1168.

[11] Eric Gilbert and Karrie Karahalios. 2009. Predicting Tie Strength with Social Media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI 09)*. Association for Computing Machinery, New York, NY, USA, 211220. https://doi.org/10.1145/1518701.1518736

[12] M. Girvan and M. E. J. Newman. 2002. American College Football. *Proc. Natl. Acad. Sci.* 99 (2002), 7821–7826.

[13] P. Gleiser and L. Danon. 2003. Community Structure in Jazz. *Adv. Complex Syst* 6 (2003), 565. http://deim.urv.cat/ aarenas/data/welcome.htm.

[14] M. Granovetter. 1978. Threshold Models of Collective Behavior. *The American Journal of Sociology* 83 (1978).

[15] J Jones, J Settle, R Bond, C Fariss, C Marlow, and J Fowler. 2013. Inferring Tie Strength from Online Directed Behavior. *PLoS ONE* 8, 1 (2013).

[16] C. Kadushin. 2012. *Understanding Social Networks: Theories, Concepts, and Findings*. Oxford University Press, USA. https://books.google.com/books?id=ALOhpMgkW_cC

[17] Indika Kahanda and J. Neville. 2009. Using transactional information to predict link strength in online social networks. *ICWSM* (01 2009), 1–10.

[18] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Predicting Positive and Negative Links in Online Social Networks. *arXiv* (2010).

[19] Zhen Liu, Hu li, and Chao Wang. 2020. NEW: A Generic Learning Model for Tie Strength Prediction in Networks. arXiv:2001.05283 [cs.SI]

[20] Peter V. Marsden and Karen E. Campbell. 1984. Measuring Tie Strength*. *Social Forces* 63, 2 (1984), 482–501.

[21] Heather Mattie, Kenth Eng-Monsen, Rich Ling, and Jukka-Pekka Onnela. 2018. Understanding tie strength in social networks using a local bow tie framework. *Scientific Reports* 8, 1 (Jun 2018).

[22] J.-P. Onnela, J. Saramäki, J. Hyvönen, G. Szabó, D. Lazer, K. Kaski, J. Kertész, and A.-L. Barabási. 2007. Structure and tie strengths in mobile communication networks. *Proceedings of the National Academy of Sciences* 104, 18 (2007), 7332–7336.

[23] L. Page, S. Brin, R. Motwani, and T. Winograd. 1998. *PageRank citation ranking: Bringing order to the web*. Technical Report. Stanford University.

[24] Usha Nandini Raghavan, Rka Albert, and Soundar Kumara. 2007. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E* 76, 3 (Sep 2007).

[25] revere 1994. Paul Revere's ride. http://kieranhealy.org/blog/archives/2013/06/09/using-metadata-to-find-paul-revere/.

[26] Rahmtin Rotabi, Krishna Kamath, Jon Kleinberg, and Aneesh Sharma. 2017. Detecting Strong Ties Using Network Motifs. *Proceedings of the 26th International Conference on World Wide Web Companion - WWW 17 Companion* (2017).

[27] Polina Rozenshtein, Nikolaj Tatti, and Aristides Gionis. 2017. Inferring the Strength of Social Ties. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Aug 2017).

[28] Stavros Sintos and Panayiotis Tsaparas. 2014. Using Strong Triadic Closure to Characterize Ties in Social Networks. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[29] snap. [n.d.]. Stanford Large Network Dataset Collection. http://snap.stanford.edu/data/.

[30] Xin Wang, Wei Lu, Martin Ester, Can Wang, and Chun Chen. 2016. Social Recommendation with Strong and Weak Ties. In *Conference on Information and Knowledge Management*. 5–14.

[31] D. J. Watts and S. H. Strogatz. 1998. Collective dynamics of small-world networks. *Nature* (Jun 1998), 440–442.

[32] Rongjing Xiang, Jennifer Neville, and Monica Rogati. 2010. Modeling Relationship Strength in Online Social Networks. In *Proceedings of the 19th International Conference on World Wide Web*.