Identifying Influential Users in Social Network with Review Data

Yilin He

Machine Learning Department
School of Computer Science
Carnegie Mellon University, Pittsburgh, PA, 15213

Abstract

Background: Social networks have been widely utilized by a variety of popular review websites, and different users have a different impact on how information propagates through the network. Identifying influential users in such system could be beneficial for advertising purposes.

Aim: To estimate the influence propagation probability for social ties in the network using review data which contains rating and temporal information. Then to identify influential users in the social network. The new method should be able to tolerate some basic attacks.

Data: The Yelp dataset was used in this analysis, and we focused on a subset of 5691 users who have at least one of 40077 reviews about 2724 businesses in Pittsburgh. Synthetic dataset for attacks is generated on top of the Yelp dataset where up to 10000 fake social ties are added in the network.

Method: The convex network inference model was adopted and applied the algorithm with prior knowledge of the social network. We then used a new method combining the two modifications on top of this method: the self-exploring model which supports user's information discovery without knowledge propagation; the rating model which incorporated rating data. The credit distribution model for influence maximization was used for comparison.

Result: Our new model achieved the best outcome in finding a set of influential users that have a high influence on the network. We also simulated attacks on the social network which changed 0.55% to 55% of the social network. Our results show that the total weight of the entire network can be changed up to 80%, but the influence per review for a set of the 100 most influential users only changes by 18% with this attack. If attackers focus on using users with only high number of reviews, this change could go up to 26.6%.

Conclusion: The network inference model is adopted and modified to estimate the influence propagation probability with review data, which is shown to perform well in identifying influential users in the network while being able to tolerate attacks on the social network without a severe impact on the result.

1 Introduction

The social network has been largely utilized by a variety of popular review websites. A lot of such websites supports login via social network sites such as Facebook or Google plus, and by doing so, it can quickly create a friend list for users to share the content with friends without suffering from a cold start problem. For example, popular review websites such as Yelp and Goodreads both support login through other social media account and will automatically link users' account with their friends' using outside social network information. Once the network is created, users can see their Facebook friends' reviews on Yelp or Goodreads without going through the process of manually adding friends after account creation.

The core features of these sites and traditional social network sites is the content creation and sharing among users. Based on positive or negative feedback given from friends, users might take different actions toward an individual product. Since users have different behavior pattern where some create content and some solely consume content, different users have a different impact on how information is propagated through the network.

Identifying these influential users in such social networks could be very beneficial for commercial websites as advertisement is the core profit strategy for to a lot of such companies. For cost-profit maximization purpose, targeting influential users in the social network is an efficient way of advertising. As a consequence, there are users and even bots dedicated to promoting themselves in such social network to profit through advertisement. Besides, recommend impactful users to other users who just joined the network might have a positive impact on their activity level.

In some case the social network influence is measurable, for example, the influence of Twitter users [1] can be evaluated through peer to peer interaction. However, most review websites do not have information regarding the direct interaction between users. And for user feedback sites such as Yelp, it might be hard to estimate the influence of each edge in the social network. This suggests a challenge for determining the probability of knowledge transfer between edges and thus finding the most influential users in the social network.

In this paper, we focus on solving the problem of estimating the influence propagation probability between users and identifying influential users in the system. To achieve this goal, we adopted an algorithm for inferring latent network structure and provided a variation of the algorithm that infers influence propagation probability between users in the social network, based on the rating and temporal information about their reviews.

2 Related Work

Richardson et al. [2] first started the study of influence maximization problem using probabilistic approaches and later Kempe et al. formulated the problem as finding a small subset of nodes k that maximizes the expected number of influenced nodes under a stochastic cascade model. This problem is proved to be NP-hard and a greedy algorithm is provided to solve the problem. However, this algorithm has a huge drawback of efficiency, so a lot of more recent work have been focusing on improving the scalability of the algorithm. In [3],

Leskovec et al. proposed "lazy forward" algorithm which largely improved the efficiency of the algorithm by exploiting the submodularity property of the objective matrix. Chen et al. [4] later proposed yet another greedy algorithm for with new degree discount heuristics that further improves the efficiency even further while achieving a matching performance to the original greedy algorithm [5].

Most of the influence maximization work assume that the ground truth of influence propagation probability is known, which is not the case in some real world situations. In [6], Goyal et al. proposed several statistic model, continuous time model and discrete time model for learning influence probability in a social network graph and predicting user action time. Their result shows that simple statistical model does not provide a good estimate of the influence probability whereas the continuous time model provides the best result. The discrete time model also provides a similar performance to the continuous time model but its much faster. Later Goyal et al. [7] built credit distribution model which directly solves the problem of influence maximization without inferring the influence probability for the network by considering the trace of information propagation and its temporal property during the process of influence maximization. The credit distribution model is used in this paper as a comparison method.

Myers et al. proposed a network inference method in [8] which solves the problem of inferring latent social networks based on network diffusion for modeling, which only requires an action log containing user and event time. They also proved the convexity of the model with l1-like sparsity penalty term. The inferred network is a weighted social network graph that can also be treated as influence propagation probability graph. In this paper, we will use this algorithm along with several proposed variation of the algorithm to estimate the influence propagation probability between edges with prior knowledge about the social network and additional information about reviews.

3 Problem Definition

To formally state the problem, we were given an undirected social network graph $G = (V, E)$ where V represents the set of users in the social network and edge $(u, v) \in E$ represents a social connection between user u and user v. We also have a set of subject S where $s \in S$ is the subject for user to review, such as a restaurant or a book. A review tuple is (u, s, r, t) where $u \in V$, $s \in S$, $r \in \{1, 2, 3, 4, 5\}$ is the rating that user u gives subject s at time t. And the complete review history $Review\,(User, Subject, Rating, Time)$ for all users is available for the analysis.

We want to infer a weighted directed version of the graph $G = (V, E')$. Set V represents the set of users in the social network and edge $e' = (u, v)$ is directed with weight $p_u v$. The weight represents the probability of influence propagation from u to v through edge e'.

Considering rated reviews from users, we define

$$\begin{aligned} p_{u,v} &= P(\text{influence propagate from node } u \text{ to node } v) \\ &= P(\text{node } v \text{ visit subject } s | \text{node } u\text{'s rating } r_{u,s} \text{is positive}) \\ or \quad & P(\text{node } v \text{ does not visit subject } s | \text{node } u\text{'s rating } r_{u,s} \text{ is negative}) \end{aligned}$$

4 Data

In this paper, we used a data set from Yelp Dataset Challenge. The entire data set contains a network of 552 thousand users with 3.5 million edges and their 2.2 million reviews for 77 thousand businesses from ten cities. A subset of the dataset is used in our analysis which contains all 2724 businesses which are located in Pittsburgh. Among the 17124 users have reviewed at least one of the selected businesses, we then selected the largest connected components in this subset of the social network containing 5691 users. Rest of the connected component only contains eight users or less and is thus ignored for rest of the study. The final data contains 5691 users with 18115 social ties and their 40077 reviews for 2724 Pittsburgh businesses. For our analysis, we used the following information from the dataset:
User file:

uid an unique identifier for each user.

friends a list of uid for current user's friends in the social network. This friendship is undirected, which means if user u is a friend of v, user v is also a friend of u.

Review file:

uid The unique identifier of reviewer.

sid The unique identifier of the business being reviewed.

review timestamp the date of which the review was made, counting from 1/1/2000 in the unit of days.

rating user's rating for this business, ranging from 1 star to 5 stars in integers.

Figures 1 and 2 show that most of the users have a very small number of friends and reviews. Both y-axes are logarithmically scaled to show that the number of the user increases exponentially as the number of friends or reviews decreases.

4.1 Following Reviews

To estimate the relationship between social ties and reviews, we introduce the idea of following reviews. Assume reviewer u and reviewer v are friends, and both of them reviewed business s at time t_u^s and t_v^s respectively, if $t_u^s < t_v^s$, which means user u reviewed the business first, then we define the first review r_u^s as the original review and the second review r_v^s as a

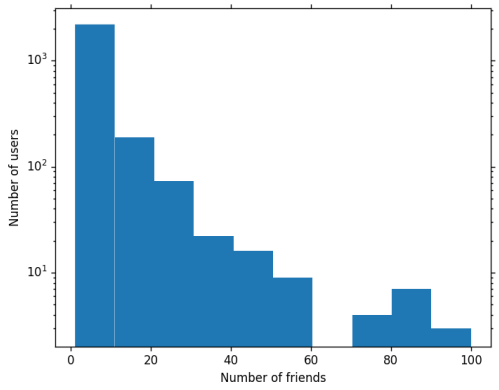

Figure 1: histogram of number of friend for each user in the social network

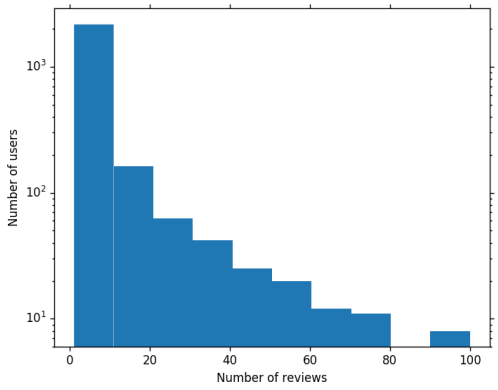

Figure 2: histogram of number of reviews made by each user

following review. Note that following reviews does not necessarily implies that user v made review r_v^s because of user u's previous review r_u^s but rather an estimation of such possibility because we do not know the truth regarding why user make their reviews. Among all 40077 reviews, 14198 of them are following reviews. Figure 3 shows the histogram of number of following reviews received for each user.

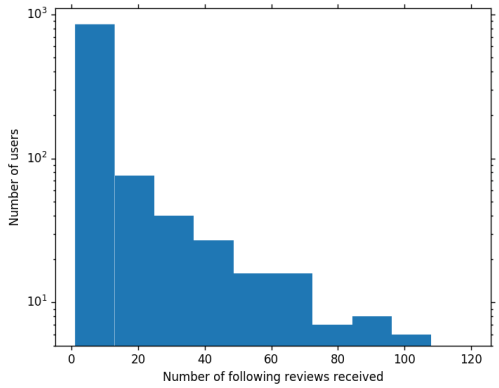

Figure 3: histograms of number of following reviews received

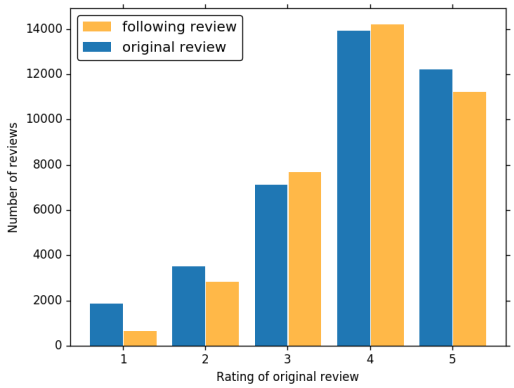

Figure 4: rating for original review vs. number of original and following reviews

Figure 4 shows that users are more likely to follow a review that is 4 stars and are less likely to follow a 1-star review. In this paper, we treat all the ratings in the reviews towards a certain subject as reliable information. The blue columns show the number of ratings for each review and the yellow column indicating the number of following reviews given the rating of the original review.

4.1.1 Temporal Property

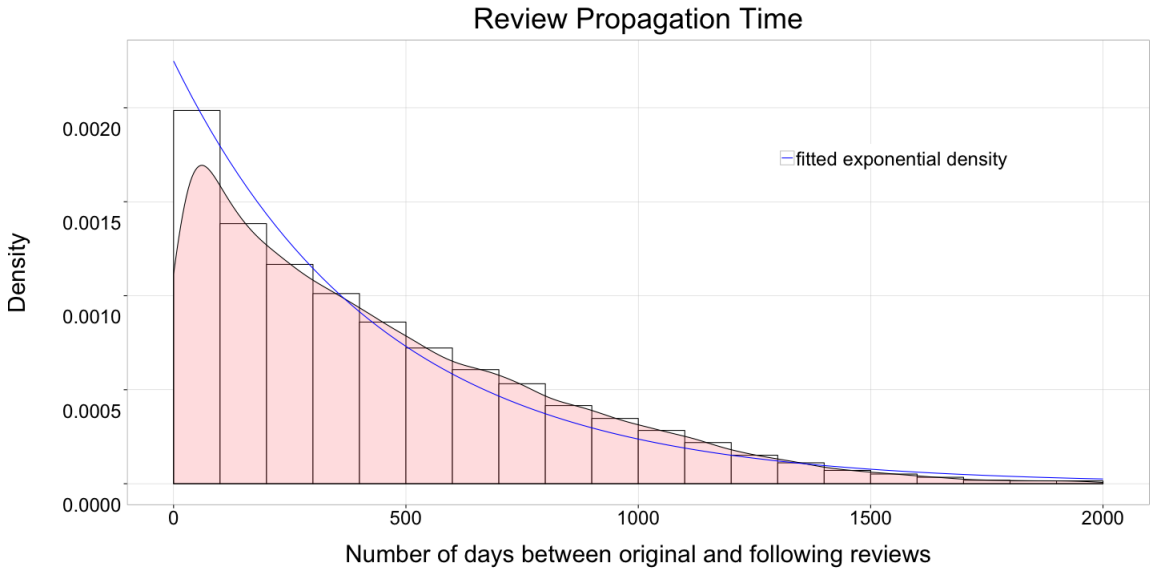

Figure 5: Density of time between original review and following review and fitted exponential distribution. This shows that review propagation time follows an exponential distribution.

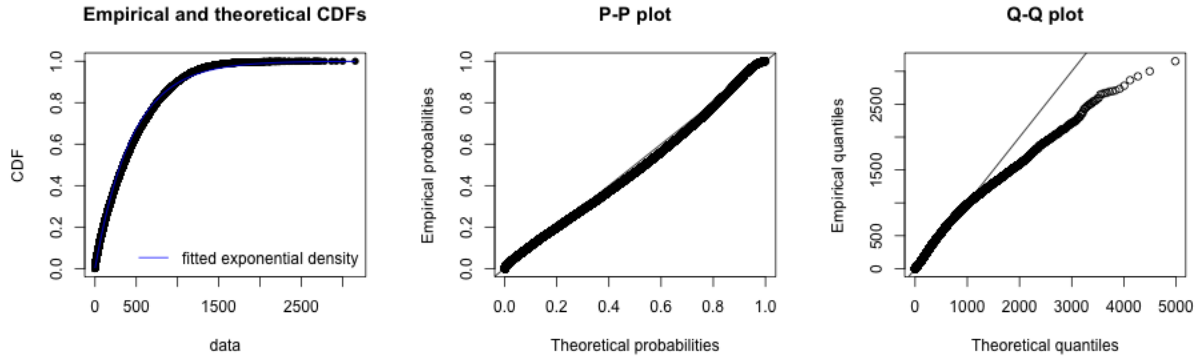

Figure 6: The empirical and theoretical CDFs, P-P plot and Q-Q plot for fitting distribution for time between original review and following review. This further supports the fact that review propagation time can be modeled by an exponential distribution.

Usually, a temporal distribution governs influence propagation. For example, it is intuitive that if user u told user v that a restaurant is good, it is more likely for user v to visit the restaurant sooner than later. In other words, if user v visited a restaurant, it is more likely that they are influenced by some information they obtained recently than last year. Figure 5 and 6 show the density of time difference between each pair of original review and the following review, suggesting that the review propagation time follows an exponential

distribution. Our analysis indicates that the distribution could be fitted with an exponential density function $\alpha \exp^{-\alpha}$ with $\alpha = 0.0022$, which will be used in the rest of this paper.

4.1.2 Following Reviews Prediction

To further understand the factors that influences information propagation, we run a logistic regression for predicting whether user would follow up on their friends' reviews. To construct the dataset, we use following review as positive samples and the set of potential following review minus following reviews as negative samples. Here a potential following review is defined as follows: let review r_u^s be the review user u made at time t_u^s, if user v is a friend of u and haven't reviewed subject s before t_u^s, we call (u, v, r_u^s) as a potential following review. The rating of original review r_u^s, number of friends for both user u and user v and the number of reviews from original reviewer u are included as features. We are not able to include any temporal features because we are missing the information about the negative samples. Since only around 2.5% of the potential following reviews turns out to be positive, we randomly selected 2.5% from the negative examples as the sample.

Using logistic regression, we can achieve an accuracy of 64.3% with a five-fold cross-validation under the following coefficients:

Coefficient	rating	number of friends for u	number of friends for v	number of reviews from u
Estimate	1.451e-01	-6.635e-04	3.148e-03	7.999e-05
$Pr(>\|z\|)$	<2e-16	<2e-16	<2e-16	0.358

It shows that the rating of the original post is the biggest indicator for following reviews in this set of features, and a higher rating does have a positive effect on promoting following reviews. The number of friends for both reviewer and follower both play a smaller role in this prediction compared to the rating. We initially assumed that number of reviews from the original reviewer might have some negative effect on creating following reviews, but we discovered that it does not have any effect on whether their friend would follow up on users' review or not.

4.2 Attack Simulation

Influential users are good subjects to pick for adverting new product. For instance, a newly opened restaurant might offer a special coupon or free dinner for these users and ask for their reviews in return. Since being selected as influential users are could lead to physical rewards, some user might attack the system in order to receive these rewards. An easy way of attacking is spamming friend request to all users to achieve a higher influence score. Therefore, we tested the performance of proposed method under this kind of attacks.

To simulate the attacks, we randomly selected n nodes from the network and added m random edges for each of them. So the total number of newly added edges is $n * m$. We then run the modified network inference algorithm on the new graph with additional fake edges

we generated. We then measured the change of network weight against the result original graph and plotted the change in the format of the heat map. In the following graphs, the x-axis shows the number of added edge m and the y-axis indicates the number of added node n. We picked four values: 10,25,50 and 100 for both n and m, giving a total of 16 experiments. Since the total number of edges in the network equals to 18115, simulated attack will change the network by 0.55% to 55%.

5 Method

5.1 Convex Network Inference

Myers et al. [8] proposed the method of convex network inference which solves the problem of inferring the diffusion matrix based on cascade information. In general network inference problems, diffusion matrix is a dense matrix of size $N * N$, which represents the influence probability between each pair of users. In the setting of network influence estimation, since we already have prior knowledge about the structure of the network, the diffusion matrix can be reduced to a sparse matrix with only K non-zero entries each represents a directed social tie.

To solve network influence estimation problem, we will use the cascade model. A cascade is the trace left in the network during the process of information spread. For this method, a cascade can be initiated by a random node at time 0; then the information propagates to another node through social ties following a temporal model $w(t)$. There are many choices for this time distribution such as power-law, exponential and Weibull distribution.

In the setting of review network, each cascade can represent a subject such as restaurant or book for the user to review. In other words, if node i reviewed a subject at time ti then the transmission time from u to v follows a temporal distribution $w(t_u - t_v)$. With cascade model, the probability a non-initiator user u to review subject s can be represented as the probability that it is influenced by any neighbors. Thus,

$$p\left(u \text{ review } s \text{ at } t_u^s | X_s\left(t_u^s\right)\right) = \left(1 - \prod_{v \in f(u), t_v^s < t_u^s} \left(1 - w\left(t_u^s - t_v^s\right) p_{vu}\right)\right)$$

Here t_u^s denotes the time at which user u visited subject s, $f\left(u\right)$ denotes the set of neighbors for user u and $X_s\left(t_u^s\right)$ represents the set of nodes that were infected at time t_u^s. The probability that user u visit s at time t_u^s is the probability that it was influenced by any of the neighbor who have visited subject s at a prior time. Since this method does not consider ratings, only positive reviews is used for the calculation.

The likelihood function for all users given the set of all cascades D is the product of two parts: the probability that user was influenced by its neighbors for all the subject at particular time t and the probability that user was not influenced by its neighbors for all the subject that they did not review.

$$L\left(P,D\right)=\prod_{s\in D}\left[\prod_{u;t_u^s<\infty}\left(1-\prod_{v\in f(u);t_v^s<t_u^s}\left(1-w\left(t_u^s-t_v^s\right)p_{vu}\right)\right)\prod_{u;t_u^s=\infty}\left(1-p_{vu}\right)\right]$$

$$L\left(P,D\right)=\prod_{s\in D}\left[\prod_{u;t_u^s<\infty}\left(1-\prod_{t_u^s<\infty;v\in f(u);t_v^s<t_u^s}\left(1-w\left(t_u^s-t_v^s\right)p_{vu}\right)\right)\prod_{u;t_u^s=\infty}\left(1-p_{vu}\right)\right]$$

Since each node in the network is independent of other nodes given its neighbors, the likelihood function for single nodes and all cascades can be written as follows:

$$\begin{aligned}L_u\left(P_{:,u}|D\right)&=\prod_{s\in D}\left[\prod_{t_u^s<\infty}p\left(u\text{ review }s\text{ at }t_u^s|X_s\left(t_u^s\right)\right)\cdot\prod_{t_u^s=\infty}p\left(u\text{ doesn't review }s|X_s\right)\right]\\&=\prod_{s\in D;t_u^s<\infty}\left(1-\prod_{v\in f(u);t_v^s<t_u^s}\left(1-w\left(t_u^s-t_v^s\right)p_{vu}\right)\right)\prod_{s\in D;t_u^s=\infty;v\in f(u);t_v^s<t_u^s}\left(1-p_{vu}\right)\end{aligned}$$

With this formulation, we can compute the optimal value for $P_{:,u}$ by finding the minimal value of $-\log L_u\left(P_{:,u}|D\right)$ which is proved to be convex in [8]. Also, because each node can be inferred independently, the model parameters can be computed in a distributed setting thus provides good scalability for the algorithm. A regularization term can also be added to control the sparsity of the network, which changed the optimization problem to

$$-logL_u\left(p_{:,u}|D\right)+\lambda\sum_{v\in f(u)}|p_{vu}|$$

5.1.1 Self-exploring Model

One assumption in the convex network inference model is that there is only one initiator for each cascade in the network. However, in reality, users are very likely to discover new subjects through various means. Therefore, it can be helpful to include the probability that user discovers subject by themselves rather than merely influenced by their neighbors.

Given this, the probability becomes

$$p\left(u\text{ review }s\text{ at }t_u^s|X_s\left(t_u^s\right)\right)=1-\left(1-w\left(t_u^s\right)p_u^s\right)\prod_{v\in f(u),t_v^s<t_u^s}\left(1-w\left(t_u^s-t_v^s\right)p_{vu}\right)$$

Note that we can treat the self-exploring model as adding a new subject node into the network and then add an edge between the user and their reviewed subject. We can treat $t_s^s=0$ as when the business started. Also, we assume that the r_s^s is always positive. By doing so, we simplified the combined algorithm as adding subject node and edges between the user and reviewed subject, while still applying the original algorithm.

5.1.2 Rating Model

First we propose a joint probability model for influence propagation with rating data. Assumes $(u, v) \in E$ and user u reviewed subject s at time t with rating r. We define a influence propagation from user u to user v as follows: user v reviewed subject s give u's rating r is positive; or user v did not visit subject s given u's rating r is negative. If more than one of user v's friend reviewed subject s, user v only visit s if there is at least one positive information influence and no negative information influence. Following this assumption,

$$p\left(u \text{ visit } s \text{ at } t_u^s | X_s\left(t_u^s\right)\right) = \left(1 - \prod_{v \in f(u), t_v^s < t_u^s} \left(1 - w\left(t_u^s - t_v^s\right) p_{vu}\right) I\left(r_v^s \text{ is positive}\right)\right) \prod_{v \in f(u), t_v^s < t_u^s} \left(1 - w\left(t_u^s - t_v^s\right) p_{vu}\right) I\left(r_v^s \text{ is negative}\right)$$

5.1.3 Combined Model

Combining rating model and self-exploring model, we obtained a new probability formulation:

$$p\left(u \text{ visit } s \text{ at } t_u^s | X_s\left(t_u^s\right)\right) = \left(1 - \left(1 - w\left(t_u^s\right) p_u^s\right) \prod_{v \in f(u), t_v^s < t_u^s} \left(1 - w\left(t_u^s - t_v^s\right) p_{vu}\right) I\left(r_v^s \text{ is positive}\right)\right) \prod_{v \in f(u), t_v^s < t_u^s} \left(1 - w\left(t_u^s - t_v^s\right) p_{vu}\right) I\left(r_v^s \text{ is negative}\right)$$

We then replace the original probability in the maximum likelihood formula for convex network inference method with above probability and solve for $-logL_u\left(p_{:,u}|D\right) + \lambda \sum_{v \in f(u)} |p_{vu}|$ to obtain the optimal incoming edge weight for node u. After finding the optimal solution for all nodes in the network, we estimate the influence of each node by calculating the sum of the weight of its outgoing degrees.

6 Results

In order to evaluate the algorithm, we first used convex network inference model, combined model and credit distribution method [7] to pick influential users in the social network, then judged how they perform in spreading reviews in the graph using Yelp dataset. Also, we tested the performance of proposed combined model under simulated attacks. For all the analysis, the influence of a particular user is estimated based on their following reviews, which can be measured using the dataset, as the actual information propagation probability of the network is unavailable for real world datasets.

6.1 Influential User Selection

We divided our dataset into training and testing data, where training data contains first 80% of the reviews ranked by review date, which means all the reviews made by user before 7/1/2014. Correspondingly the testing dataset contains 20% of the review data after that time. Three methods were used for evaluation, namely the original convex network inference model, a variation of this model with rating and self-exploring and credit distribution model. We use these three algorithms to select 100 seed user which algorithm chose to be the most influential users in the dataset based using training data. After selecting seed users, we measure the number of following reviews between seed user and their friends.

In modified convex network inference model with rating and self-exploring method, we use reviews with rating greater than 3 as positive examples and rating less than 3 as negative examples. For original model, we only used positive samples in the analysis. For both convex network inference model and the modified version, we modeled the influence propagation time distribution $w(t)$ using an exponential distribution with alpha = 0.0022 according to the analysis of the distribution of time passed between original reviews and following reviews. For credit distribute model we used the implementation provided by Goyal et al. [7] with a truncation threshold of 0.001.

Let κ_r denote the number of following reviews in the testing dataset for original review r. Let R_u^{train} and R_u^{test} denote the user u's reviews in training and testing dataset. The *total influence* for user u is defined as $\sum_{r \in R_u^{train} \cup R_u^{test}} \kappa_r$ and total influence per review is defined as $inf_u = \frac{1}{|R_u^{train}|} \sum_{r \in R_u^{train}} \kappa_r + \frac{1}{|R_u^{test}|} \sum_{r \in R_u^{test}} \kappa_r$. The total influence spread for a set of user K_{Seeds} is the sum of their influence score except each following review is only counted once. For example, if the seed set only contains user u and user v, who both reviewed subject s before their mutual friend z made their review for s in the test dataset, then the total influence score for the seed set in this case would be 1 instead of 2. Even though the review z made could be a following review for both u and v, we only count it once when calculating the total influence of a set of users. The legacy influence from reviews in the training set is included due to the sparsity of the review data where more than 30% of the picked users doesn't have any new reviews in the test dataset.

6.2 Attack Tolerance

First, we measured the change of edge weight for both newly added edge and the entire network. Let p_{uv} be the original estimation of weight for edge (u, v) and p'_{uv} be the new estimation under the attack, *change of total graph weight* $= \frac{\sum_{(u,v) \in E} (p_{uv} - p'_{uv})^2}{\sum_{(u,v) \in E} p_{uv}^2}$. Figure 8. shows the change of edge weight in the entire network measured by computing the change of total graph weight compared with original estimation. According to the graph, attacking 55% of the edge changes the total network weight up tp 80%. Figure 9. shows that regardless of the change in total network weight, the MSE of estimated weight for fake edge is less than 0.11 with a change up to 55% of the graph.

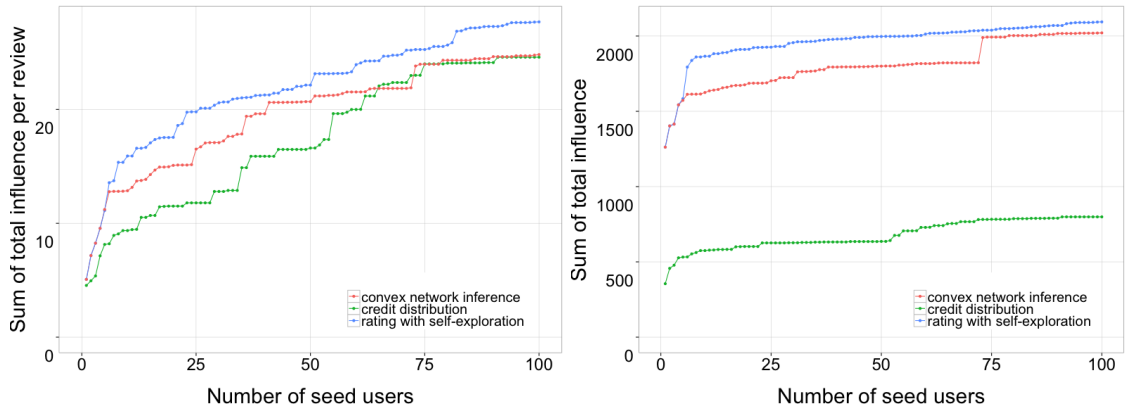

Figure 7: The estimated influence of a selected set of users measured by the sum of total influence per review on the left and the sum of total influence on the right. The influential users are identified using 3 different models: convex network inference model, credit distribution model and proposed rating with self-exploration model.

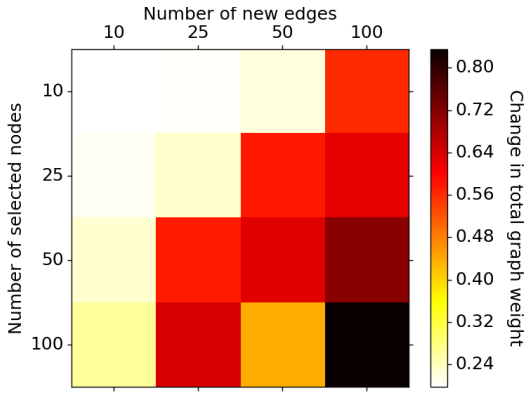

Figure 8: The heatmap represents the change of total graph weight between the estimated information propagation probability with and without the attack. This change indicates the overall impact of the attack for the algorithm.

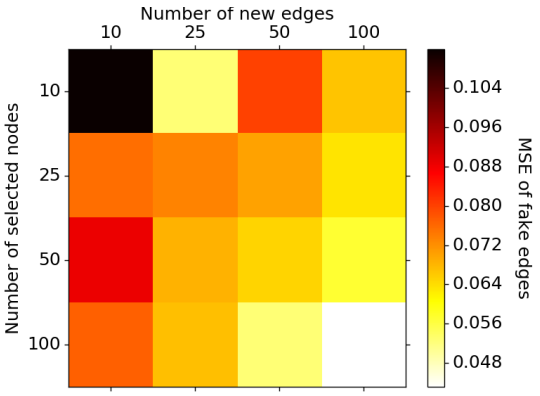

Figure 9: MSE of estimated influence propagation probability of fake edge under the attack. A lower MSE suggests that the algorithm is better at identifying fake social ties created under the attack.

We then calculate the 100 most influential node under the attack and measured their sum of total influence per review. Let K_o and K_a denote the sum of total influence per review for original set of users and set under attacks respectively, the change is measured by $\frac{K_a - K_o}{K_o}$. In addition to randomly selecting nodes and adding fake edges for them, we also experimented with users samples of a larger number of reviews. We randomly selected n users with 10 reviews or more and added m fake edges for them.

Figure 10 shows that attacking random nodes and adding 55% more edges to the network decreases the influence made by 100 seed users only by 18%. When the attack targets users with more reviews, the influence made by 100 seed users decreases by 26.6%.

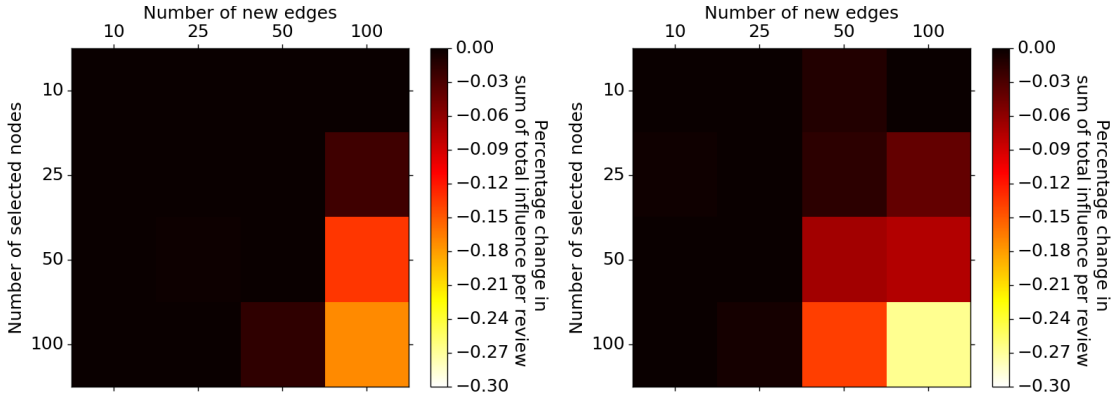

Figure 10: Percentage change of sum of total influence for 100 seed users under attacks targeting random users(left) and targeting users with more than 10 reviews(right). The change is measured by the difference between new result and the old result in percentage changed, which indicates how the performance of selecting influential users was affected by the attack.

7 Discussion

The result of influential user selection in figure 7 shows that the among all three algorithms our proposed method achieved highest total influence and total influence per review past 8 users, where the total influence grows logarithmically with respect to the number of seed users. Convex network inference algorithm performs better than credit distribution model for a small set of seeds, but the differences quickly decreases as the number of seeds increases. We also notice that although the total influence of seed users selected by credit distribution model is much smaller than the other two algorithms, the performance difference in total influence per review is much less. This might suggest that credit distribution model is more likely to select users with fewer reviews in the dataset.

As for attack simulation, our result in figure 8 and 9 indicate that attacking the entire network with a large number of fake edges can have big impact on the inferred propagation probability graph, however, the algorithm can still detect that it's unlikely for information to propagate through those fake edges. In terms of selecting most influential users, the proposed algorithm still outperforms the comparison algorithm unless attackers added 100 fake social ties to more than 50 users. If the attacker picked users with more reviews as their target, the attack would make a bigger impact especially when adding a large number of fake edges to the selected nodes.

7.1 Limitation

In our paper, we defined and measured the influence for a specific user in term of number of following reviews they received for each review they make which we believe is a good estimation for such influence. In reality we do not know the truth of how likely an user is going to be influenced by another user, and such influence also depends on the review

subject. For other influence maximization purpose, alternative definition might be used and the model could be modified to adjust for such definitions. Also, our simulated attacks is fairly simple compared to what attackers could do in real life. In future works we would like to explore the impact of different types of attacks on our model.

The original Yelp data set also contains text information and other attributes of different subjects. We did not include such information in our analysis due to limited time and generalization purpose. In future work, text information could be incorporated in various ways such as understanding whether the user made this review based on a friend's review or solely because they are interested in the subject.

8 Conclusion

We have adopted the convex network inference method in the context of network inference and proposed a variation of the model. The variation combines a self-discovery model which allows users to learn knowledge from outside resources in addition to the information propagation within the social network and rating model which considers the rating of the original review. Our analysis shows that the proposed algorithm performs better in selecting a set of influential users compared with credit distribution model.

We also discussed the possible attacks on such social network and tested the performance of proposed algorithm. Our result shows that attacks might change the total weight of the social network graph dramatically but have very limited impact on the selection of most influential users in the social network. Thus possible real life application of the model such as selecting advertisement and promotions targets would be able to tolerate some potential attacks.

9 Acknowledgments

I would like to thank my committee members Eric Xing and Roy Maxion for their advice and support for this project. Assistance provided by Yaoliang Yu throughout different phrases of the project was also greatly appreciated. I would also like to thank Yelp for making this awesome dataset available for the researchers.

References

[1] M. Cha, H. Haddadi, F. Benevenuto, and P. K. Gummadi, "Measuring user influence in twitter: The million follower fallacy.," *ICWSM*, vol. 10, no. 10-17, p. 30, 2010.

[2] M. Richardson and P. Domingos, "Mining knowledge-sharing sites for viral marketing," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 61–70, ACM, 2002.

[3] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, "Cost-effective outbreak detection in networks," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 420–429, ACM, 2007.

[4] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 199–208, ACM, 2009.

[5] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137–146, ACM, 2003.

[6] A. Goyal, F. Bonchi, and L. V. Lakshmanan, "Learning influence probabilities in social networks," in *Proceedings of the third ACM international conference on Web search and data mining*, pp. 241–250, ACM, 2010.

[7] A. Goyal, F. Bonchi, and L. V. Lakshmanan, "A data-based approach to social influence maximization," *Proceedings of the VLDB Endowment*, vol. 5, no. 1, pp. 73–84, 2011.

[8] S. Myers and J. Leskovec, "On the convexity of latent social network inference," in *Advances in Neural Information Processing Systems*, pp. 1741–1749, 2010.

[9] Yelp.com, "Yelp dataset challenge." `https://www.yelp.com/dataset_challenge`, 2015. [Online; accessed 30-Oct-2015].