First Invites System - First few invites are precious !

Aravind Senguttuvan
University Of Utah

Arun Alamsetty
University Of Utah

Chitradeep Dutta Roy
University Of Utah

Abstract

Finding the major players with similar business interests in a community is a challenge in various domains.

We propose to find the major players in a community by triangle counting and cluster coefficient calculation on social graph. To find the players with similar interests, we assign weights to the edges of the social graph as jaccard similarity of vertices' interests.

For instance, distributing the first invites in a referral invite system can be solved using cluster coefficient calculation as mentioned above.

Classification Keywords

referral Invite; referral Incentive; referral system; triangle counting; jaccard Similarity; referral discounts;

1 Introduction

Reviews about businesses has changed from word of mouth to sharing posts in social media. Towards the end of 2014, any new system is made to look exclusive and lucrative by inviting few members to a referral invite system. Thereby the trend doesnt fade away and gains more attention in the community.

Limited time discount coupons (or) freebies are given by businesses to spread brand name to major players in any community. Discounts in the coupons are based on the number of people referred (or) number of people who used the product. Examples include OnePlusone invite, Google Inbox invite and Earhoox new brand referral invite. Oneplusone uses the concept of number of people who used the product whereas earhoox model works on the basis of number of people referred. Refer figure 1 for this system.

Lets take the Earhoox example and analyse it further. Earhoox announced free earhoox for people, who refer more than 20 people to visit their page within 3 days. Earhoox sent the invites to major reviewers like TechCrunch, CNET. By this method, Within 3 days, the Earhoox was able to advertise its brand name.

Problem statement: In a referral system, it is essential to give the first invites or discount coupons to major players in the society. In a social graph, these major players have followers or friends crowded around them.These referral discounts are available only for a limited time to reduce loss for the business. So the referrals should reach maximum amount of people in minimum time.

Word of mouth and sharing facebook posts are alike. They die over time even in the referral invite system. Most people might not receive the invites because the invites reached uninterested parties.

Real life research challenges include finding the major players in a diverse community. We can address that situation by picking communities by diverse locations or languages or interests.

Many might argue that giving the first invites to the highest degree nodes in a cluster is the best tactic but this bias leads to relying only on the major players to reach out to everyone in a cluster. Highest degree doesnt guarantee that people the information is spread to have high degree as well. When we count

triangles for the first invites system, we evaluate and offer better guarantee that such spread to almost all nodes in the community is acheived.

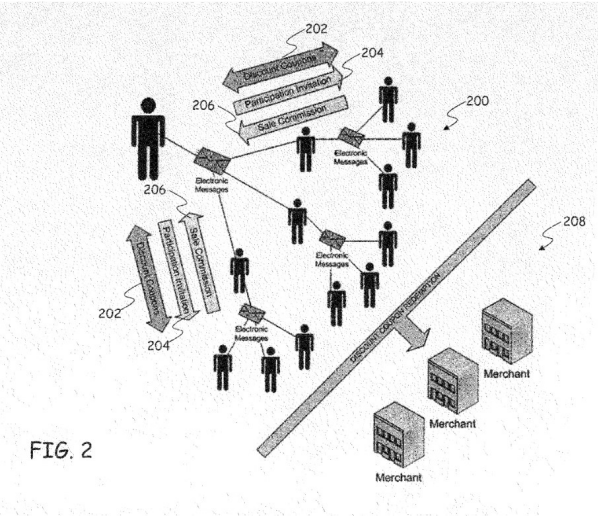

Figure 1: Sample referral Invites System

2 Definitions

- **First invites:** Exclusive invites which were initially sent out by the business.
- **Major players:** The users who received the first invites from the business in question certainly people who would can spread referral invites to more interested people quickest.

3 Scope

1. We concentrate on picking the *major players* within a connected graph, assuming we are able to separate the totally disjoint clusters by centrality or clustering mechanism. (ie) There are no completely disjoint clusters in the dataset.
2. We also assume that our problem is not to find the optimal number of invites to be sent out to a community.
3. Our approach doesnt offer solution to find the probabilty with which a person in the community would spread their referral invites to connections so that the invites reach almost all nodes in the community.
4. Our solution deals only with first invites sent out by the business in question. We perform evaluation under this assumption.

4 Our Contributions

4.1 Presenting First Invites System

We show how First Invites System can be developed based on triangle counts and cluster coefficients.

4.2 Implementing First Invite System

First Invites System is a set of 7 MR Phases to extract data, finding jaccard similarity, counting triangles, calculate cluster coefficients and arriving at top k major players in the community.

4.3 Evaluating First Invite System

We evaluate our First Invite System by a breadth first search from all major players and calculating minimum threshold of the probabilities of an invite reaching any node in the community.

5 Assumptions

1. The dataset has a connected graph. i.e. there is no completely disjoint clusters.
2. Assume the optimal number of invites to be sent out to a community be k.
3. Assume that a person sends out his/her referral invites to any friends with equal probability, therefore $P = l/d$ where l is maximum number of invites that can be sent by any invitee and d is the degree (no. of friends) of the invitee.
4. Let business B be the business requesting the major players in the community to decide the recipients of their first few seed invites.
5. Top k major players are selected by cluster coefficients with highest degree then review counts as the tie breaker.

6 Dataset and attributes used

6.1 Major Dataset used for results

Yelp Dataset: link

6.2 Attributes

Refer figure 2 for the attributes in the dataset.

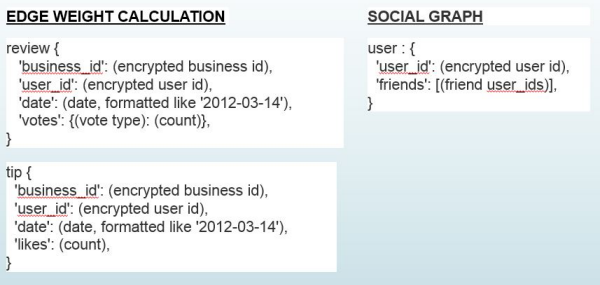

Figure 2: Dataset Attributes

7 Triangle counting inference

7.1 What does Triangle counting mean in real world?

In a social graph, Communities crowdedness is related to density of the triangles. Clustering Coefficient cc(v) which shows that how much community around one user is present;[5] For each vertex v, clustering coefficient is the number of triangles having one vertex/user v above maximum number of triangles that v can be a part of. From Suri et al[7], See the figure 3 formula for clustering coefficient. We also refered Park et al[4] for implementation.

$$cc(v) = \frac{|\{(u,w) \in E \mid u \in \Gamma(v) \text{ and } w \in \Gamma(v)\}|}{\binom{d_v}{2}}.$$

Figure 3: Clustering coefficient

7.2 Jaccard similarity in social graphs.

Triangle counting is done on social graph with edges indicating similar interest between users. We assign Edge weight as the similarity in interest between two friends in the social graph calculated from two users' checkins overlap over total no. of checkins. Refer the figure 4

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}.$$

Figure 4: Jaccard similarity

In the above example, $JS(A,B) = e(A,B) = 1/3$ Refer the figure 5

Figure 5: Example for Jaccard similarity

8 Methodology

We needed 7 MR phases to arrive at our output

1. Finding Jaccard Similarity MR phase
2. Friends graph extraction MR phase
3. NodeIterator MR phase 1
4. NodeIterator MR phase 2
5. Triangle Sum MR phase
6. Weighted Cluster Coefficient MR phase
7. Top k select MR phase

8.1 Finding Jaccard Similarity MR phase

- We filter businesses $B_1, B_2, \ldots B_n$ which have restaurant categories similar to B.
- We find similar check-ins between two users U_1 and U_2 for those filtered restaurant $B_1..B_n$.
- Assign edge weight $e(U_1, U_2)$ as Jaccard similarity, percentage of overlapping check-ins.
- The edge weights are normalized to $[0-1]$ for ignoring absolute bias during triangle counting.

8.2 Friends graph extraction MR phase

- We get degree of every user in the social graph for use in next phase
- Mapper gets the users and their friends from json of yelp dataset
- The reducer produces the key-value pairs $\{(u,v) : (degree(u), degree(v))\}$

We are indebted for the next two section implementation to Suri et al[7].

8.3 NodeIterator MR phase 1

- Here, the mapper emits only edges only when the first vertex of the edge has smaller degree, tie is broken by lexicographical ordering of vertex ids.
- Therefore every edge is only accounted for by the vertex of lower degree.
- In reducer, all keys are shuffled and received in key-value format $\{u : v\}$ where u, v are the lower and higher degree vertices respectively.
- The reducer emits all possible two pairs from the list of neighbors of u.

8.4 NodeIterator MR phase 2

- Mapper emits actual edges as present in the original adjacency list that describes the friends network as $\{u, v : \$\}$
- Other virtual edges from previous reducer (all two pairs) as $\{u, v : w\}$.
- If the reducer finds a virtual edge as well as real edge, in that case a triangle is successfully counted towards the vertex with $\min(degree(u), degree(v), degree(w))$.
- We slightly modified this by emitting $u : 1$, $v : 1$ and $w : 1$ i.e. accounting the triangle for each participating vertices.

8.5 Sum MR phase

- For every vertex(v) the map phase simply emits the key-value pair $\{v : (1, degree(v)\}$.
- Reducer sums up the values and thereby counts the triangles for the a vertex and calculates the cluster coefficient for the vertex dividing the degree. Refer figure 3 for cluster coefficient formula.

8.6 Weighted Cluster Coefficient phase

- Two factors affect the review of users in a business need to be accounted for the cluster coefficient.
- $Factor_{visit}$ decides whether a user has visited the particular business or not.
- If a user has not visited the business B (or) B is a new restaurant, $Factor_{visit} = 0.9$, else 1.
- Similarly, $Factor_{time} = 1$ for recent (within last 1 year) time of visit of $B_1..B_n$ else 0.9.
- We calculate new cluster coefficients for each vertex v based on equation below.

$$cc_{weighted}(v) = Factor_{visit} * Factor_{time} * cc(v) \quad (1)$$

8.7 Top k select MR phase

- We have the cluster coefficients for the all the vertices from previous phase
- In the mapper, We emit cluster coefficient as keys and vertices as values

- Since shuffle phase sorts all keys, we force the MR job to have only one reducer to get top k coefficients.

9 Disjoint Cluster scenario

With two regular disconnected graph partitions, we have left 20,000 nodes which never get selected for invites. For this reason, we can use betweeness centrality measures to find the different centrality nodes connecting the two disconnected graph partitions. We send referrals to top k centrality nodes to solve this problem. Refer the figure 6

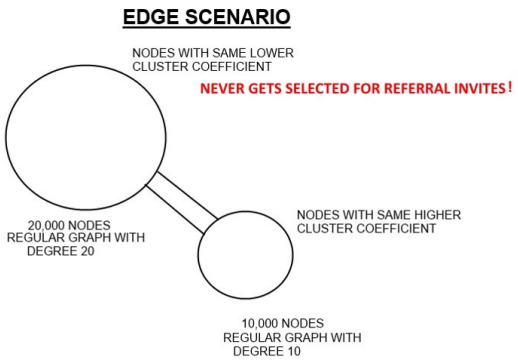

Figure 6: Disjoint cluster scenario

10 Results and Evaluation

10.1 Replication rate and reducer size

For a graph with m edges and n nodes, find the reducer size and replication rate for each MR phase in table 1.

10.2 Algorithm complexity

On a graph with m edges, Our algorithm reaches its highest computation complexity only at triangle counting and hence algorithm takes O($m^{\frac{3}{2}}$)

10.3 Evaluation algorithm and criterion

10.3.1 Evaluation algorithm

We do a bread first search starting from the first few invitee nodes selected as the top k major players. The breadth first search ends after i level (or) hops when all nodes in the network are reached atleast once.

The first seed invitee nodes are assigned probability of 1. Now we calculate the conditional probability of receiving an invite by every other node in each hop according to formula below.

Say α is an invitee node, β is any neighbor of the invitee, d(α) is degree of the invitee node, l is the maximum number of invites an invitee can send.

$$P(\beta) = \begin{cases} \frac{l}{d(\alpha)} * P(\alpha) + P(\beta) & \text{if } P(\beta) \neq 0, \text{ visited already} \\ \frac{l}{d(\alpha)} * P(\alpha) & \text{if } P(\beta) = 0, \text{ not visited} \end{cases} \quad (2)$$

Note that, $P(\alpha)$ is 1 for the seed nodes, but could have different values for deeper nodes as the BFS progresses.

10.3.2 Evaluation criterion 1 - Percentage of nodes reached with a certain Threshold probability

We find the number of nodes which have a probability of receiving an invite greater than a threshold value say 0.5. This gives us an approximate count of nodes that can be reached with a particular probability and we also note how many steps does it take to reach there.

10.3.3 Evaluation criterion 2 - Expected number of nodes receiving an invite

Summing up the probabilities of all nodes in the graph, we get the expected number of nodes that can be reached after i hops where i is the no. of hops required to reach every node in the graph with some positive probability greater than zero.

10.3.4 Selection of Evaluation criteria

We selected Expectation criterion 2 rather than the first criterion because providing absolute threshold to an algorithm varies from one dataset to another.

MR Phase	Reducer size	Replication rate
Finding Jaccard Similarity MR phase	$\mathcal{O}(1)$	$\frac{2*m}{m}=\mathcal{O}(2)$
Friends graph extraction MR phase	$\mathcal{O}(1)$	$\frac{2*m}{m}=\mathcal{O}(2)$
NodeIterator MR phase 1	$\mathcal{O}(\sqrt{m})$	$\mathcal{O}(m^{\frac{3}{2}})$
NodeIterator MR phase 2	$\mathcal{O}(\sqrt{m})$	$\mathcal{O}(m^{\frac{3}{2}})$
Triangle Sum MR phase	$\mathcal{O}(degree^2_{max})$	$\mathcal{O}(1)$
Weighted Cluster Coefficient MR phase	$\mathcal{O}(1)$	$\mathcal{O}(1)$
Top k select MR phase	$\mathcal{O}(cc(v)_{max})$	$\mathcal{O}(1)$

Table 1: Replication rate and reducer size

10.4 Evaluation of Yelp dataset

10.4.1 Precision and scale

We used *BigDecimal* class of java for achieving a scale of 100 i.e. 100 digits after decimal point.

10.4.2 Inputs to our algorithm

1. No of users in yelp dataset: 252898
2. Category Keywords for Business B: [”food”, ”restaurant”, ”breakfast”, ”lunch”, ”dining”, ”dessert”]
3. k, Number of first invites was varied and run with different constant values.
4. l, Maximum number of invites sent out by any invitee = 10

10.4.3 Intermediate outputs

l,d,k are defined in the assumption section 5 We found the major players in the yelp dataset to whom we need to send referral invites for a particular business.

1. We filtered out the users based on business B's category, No of users who have reviewed at least a business of Bs category (restaurants): 192225
2. File with filtered users who reviewed businesses from restaurant category and their reviews count: *Users Interested*
3. File with shrunken adjacency list graph of friends with only above users: *Modified Adjacency List*
4. File with triangles for such users: Triangle Counts
5. Top 10 major players (sorted by cluster coefficients breaking ties with maximum degree). Refer table 2
6. Top 10 major players (sorted by outdegree). Refer table 3

10.4.4 Outputs from our algorithm

By choosing the value of k from $\{100, 500, 1000\}$ and $l = 10$, we found the expected number of useres to be reached in $[1 \ldots 9]$ hops. The whole network in yelp dataset was covered by BFS in 8 (or) 9 hops (high-degree and cluster-coefficient approaches respectively) starting with the first seed invitees. Refer the figures 7, 8 and 9.

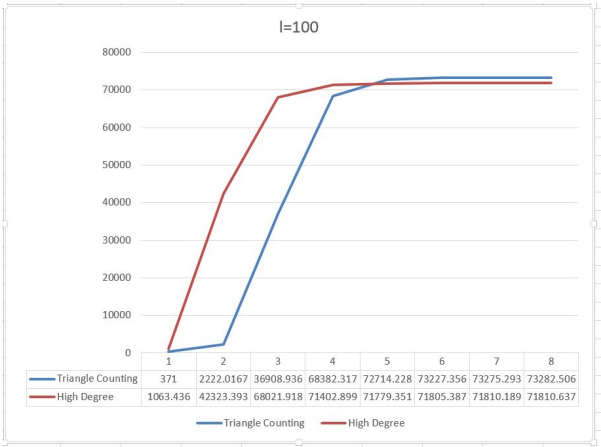

Figure 7: Graph for k=100

User names	Clustering coefficient
kelvin	1.0
Aaron	1.0
elizabeth	1.0
Bella	1.0
Hapi	1.0
Ellen	1.0
Chante	1.0
Genny	1.0
Tom	1.0
David	1.0
Irene	1.0

Table 2: Users with top 10 Clustering coefficient

User names	Outdegree
Kimquyen	2672
Philip	2442
Gabi	2432
Katie	2310
Hazel	2007
Carol	1993
Julie	1833
Daniel	1771
Connie	1745
Abby	1730

Table 3: Users with top 10 outdegrees

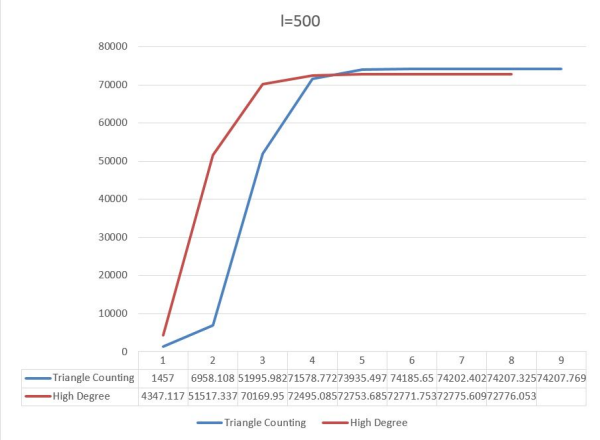

Figure 8: Graph for k=500

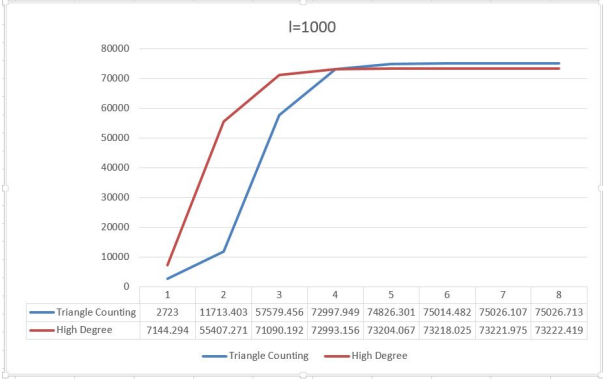

Figure 9: Graph for k=1000

10.4.5 Inference about the result

From the yelp dataset, we found that although high degree approach leads to higher expected no of users early but eventually triangle counting approach led to more even distribtuion and therefore a higher final expectation value.

10.4.6 Evaluating our method with other existing methods

We also tested picking nodes at random for evaluation. But it did not seem like a useful approach. We understood that it would require testing with large number of samples. Some results that we got are shown here, 4 for $k = 1000$ first invites and $l = 10$.

10.5 Evaluating other datasets

Results from different datasets indicated the truth that density of the similar interests mattered over only degree of nodes. We ran triangle counting on *Enron* dataset webgraph initially and some of the results that we have found are the following.

1. Total No. of Users: 36692
2. No. of Triangles: 727044
3. Average Cluster Coefficient: .497

We found that the community in Yelp dataset is much stronger because the percentage of closed triangles is much higher in Yelp than in Enron.

11 Related work

Some of the closely related work are noted below.

11.1 Hierarchical referral system[6]

This patent discusses just about using a user interface configured to enable a referrer to create an electronic message to spread through one's community. This is base patent for all citations involving hierarchical system of referral system.

11.2 System and method including referral processing[2]

This patent gives more idea on how and who is rewarded in a referral system.

11.3 Networked referral system[3]

This patent discusses more about referral rules and how referals can be used to attack more specific crowds of interest.

11.4 On-line referral[1]

This patent explains the use of referrals instead of spamming the consumers.

11.5 Uniqueness of our work

Our work for referral system gives an idea to the business whom they should target to spread their product through a referal invite system. This was not discussed in the previous patents or papers involving counting triangles and applications.

12 Availability

The source code is available for use at the site below for two reasons. First, Our application doesnt need obscurity to provide security and privacy. Secondly, Proclaiming it open source, the work can be approved by any privacy fundamentalist or pragmatist.

Project Github link : First Invite System

Evaluation Part Github link : Evaluation of First Invite System

13 Acknowledgments

We were captivated by the referral invite system provided by Earhoox and we created an effective model to get the most out of any referral system.

14 Obstacles and Future Work

14.1 Implementation Goals

- It would lead to much more even distribution, therefore better results if we can detect almost disjoint clusters (i.e. clusters that are densely connected among themselves but connected with each other via very small no. of edges.). Therefore we would like to further

Major player picking Techniques	Expected number of users who have invites after 8 hops
Triangle Counting and cluster coefficient based	75026
Highest degree based	73222
Random picking	34000

Table 4: Evaluation of Technique

this by introducing *Betweenness* or *Centrality* in our approach.

- Filtering similar businesses is currently based on simple keywords matching from business categories as labelled by Yelp. Some better measure could be developed to make this process suitable for random dataset, where such labelling is missing. It's very much a clustering problem on its own though.

- In Weighted Cluster Coefficient MR phase, Vote factor = Number of votes for a user review for $R_1 \dots R_k$ (if a user was voted more for his reviews his cluster coefficient is given more preference)

- In JaccardSimilarity MR phase, We can make use of transitive relations in the triangle to calculate a variant which takes into account the structure of a network.

$$P_{total}(a|c) = P_{direct}(a|c) + \sum_{i=0}^{d} P_{direct}(a|i) * P_{direct}(i|c) \tag{3}$$

where i is the intermediate node in any triangle $\{a,i,c\}$ for nodes a and c.
where d is number of triangles with a and c.

14.2 Future directions for our idea

- We will consider other applications apart from First Invite System for triangle counting in social graph with similarity based edges.

- We also plan to design algorithms for diversity based first invites distribution.

15 Discussion of merits

15.1 Similarity in interests

We give invites based on jaccard similarity measure. Hence this guarantees that being a friend in the circle doesnt guarantee that a person should get an invite. Both the user and the invitee should have more common interests than other neighbours, only then the user has a probability to receive the invite from the invitee.

15.2 Disjoint clusters

Disjoint cluster problem can be avoided by finding betweenness centrality for all nodes in the graph. The number of shortest paths through the gateway nodes will be significantly higher. These are the centrality nodes which have higher betweenness centrality. Inviting them guarantees that disjoint clusters in the community are not left out. Refer figure 6 for disjoing clusters problem.

15.3 Centrality nodes from Disjoint clusters and clustering coefficient from counting triangles

Nodes with higher cluster coefficient reach more nodes than others. But in addition giving invites to centrality nodes also improves our chances of reaching a larger crowd.

16 Conclusion

"Referral invite system" was introduced so that the product doesn't fade over time. But now "Referral invite system" trend itself is fading away. Example, Google Inbox invites were at first were very scarce and everyone was going berserk to use the

product but now everyone is getting multiple invites for google Inbox. referral invite systems should be used when any business wants to invite more customers. But certain referral systems die over time through spider traps and businesses come with the tactic of randomly introducing a single day of free invites for everyone.

Business as the name implies aspires to gain not lose money because of these referral discounts. And we assure maximum coverage!

References

[1] ARUMUGAM, A., BERCY, Y., BHORKAR, M., AND FABRIZIO, E. On-line referral, Dec. 2 2010. US Patent App. 12/475,306.

[2] CANETTO, M. System and method including referral processing, Aug. 23 2012. US Patent App. 13/402,180.

[3] CHOPRA, B., AND TISSERA, J. Networked referral system, Nov. 17 2011. US Patent App. 12/778,856.

[4] PARK, H.-M., AND CHUNG, C.-W. An efficient mapreduce algorithm for counting triangles in a very large graph. In *Proceedings of the 22Nd ACM International Conference on Conference on Information & Knowledge Management* (New York, NY, USA, 2013), CIKM '13, ACM, pp. 539–548.

[5] PHILLIPS, J. M. Jeff's notes on triangle counting, Aug. 23 2013. Notes.

[6] REMPE, N., AND MAJOR, B. Hierarchical referral system, Nov. 15 2007. US Patent App. 11/796,678.

[7] SURI, S., AND VASSILVITSKII, S. Counting triangles and the curse of the last reducer. In *Proceedings of the 20th International Conference on World Wide Web* (New York, NY, USA, 2011), WWW '11, ACM, pp. 607–614.