Community Detection by Ensemble - Supervised model using Edge, Node and Graph Attributes (E-SEGNA)

Govardana Sachithanandam Ramachandran

Abstract

Major social network sites such as Facebook, Twitter & Google+ offer manual classification of friends in the form of List or Circle. With average node degree on these site increasing & with user's temporal interest evolves the manual classification & re-classification of users become laborious. The project baselines the effectiveness two of more popular existing approaches which uses network structure and nodes attributes to determine number of circles, node membership of these circles and aspects that influences the circle formation. With the baseline number the project analyzes shortcomings of these approaches and tries improvements on the existing approaches. The project is an adaption of problem "Learning Social Circles in Networks" competition in kaggle.com

Dataset

The dataset used for this project is published @ https://www.kaggle.com/c/learning-social-circles/data . It contains labeled social network data in Facebook, Google+ and Twitter.

Properties	Values
Ego Networks	1,143
Circles	5,541
Users	192,075
Node Attributes	26

[Table:1 The dataset contains]

Existing work:

Following two top models are studied: Circles & CESNA and their performance are used as base model

Summary:

Circles:

Here circle formation is modeled on following properties (1) Nodes within circles should have common properties. (2) Different circles should be formed by different aspects (3) Circles are allowed to overlap. (4) Both profile information and ego network structure is used to identify the circles.

It uses a generative unsupervised learning model for finding communities, their membership and aspect that defines the community.

Here log-likelihood of ego-network G & Circle C is given by

$$l_\Theta(G;\mathcal{C}) = \sum_{e\in E} \Phi(e) - \sum_{e\in V\times V} \log\left(1+e^{\Phi(e)}\right)$$

Where

$$\Phi(e) = \sum_{C_k\in\mathcal{C}} d_k(e)\,\langle\phi(e),\theta_k\rangle$$

$\phi(x,y)$ - encodes the similarity between the profiles attributes of two users x and y

θ_k - encodes what dimensions of profile similarity caused the circle to formation (defines the circle aspect)

$d_k(e) = \delta(e \in C_k) - \alpha_k\delta(e \notin C_k)$ - Defines the edge community membership

Here parameters are optimized by unsupervised learning

$$\hat{\Theta},\hat{\mathcal{C}} = \underset{\Theta,\mathcal{C}}{\operatorname{argmax}}\, l_\Theta(G;\mathcal{C}) - \lambda\Omega(\theta).$$

Here

$\lambda\Omega(\theta).$ - for regularization

Similar to that of Expectation Maximization $\mathcal{C}^t$ & Θ^{t+1} are updated alternatively

Θ^{t+1} by gradient Ascent with below partial derivatives

$$\frac{\partial l}{\partial\theta_k} = \sum_{e\in V\times V} -d_e(k)\phi(e)_k\frac{e^{\Phi(e)}}{1+e^{\Phi(e)}} + \sum_{e\in E} d_k(e)\phi(e)_k - \frac{\partial\Omega}{\partial\theta_k}$$

$$\frac{\partial l}{\partial\alpha_k} = \sum_{e\in V\times V} \delta(e\notin C_k)\,\langle\phi(e),\theta_k\rangle\frac{e^{\Phi(e)}}{1+e^{\Phi(e)}} - \sum_{e\in E}\delta(e\notin C_k)\,\langle\phi(e),\theta_k\rangle.$$

While $\mathcal{C}^t$ is maximized by expressing it as pseudo-boolean optimization in a pairwise model, by maximizing the Energy of edge

$$C_k = \underset{C}{\operatorname{argmax}} \sum_{(x,y)\in V\times V} E_{(x,y)}(\delta(x\in C),\delta(y\in C)).$$

Number of communities is compute by minimizing Bayesian Information Criterion

$$\hat{K} = \underset{K}{\operatorname{argmin}}\, BIC(K;\Theta^K)$$

Where BIC is approximated as

$$BIC(K;\Theta^K) \simeq -2l_{\Theta^K}(G;\mathcal{C}) + |\Theta^K|\log|E|$$

With best performing feature vector being defined as

$$\phi^1(x,y) = (1;-\sigma_{x,y}).$$

Where

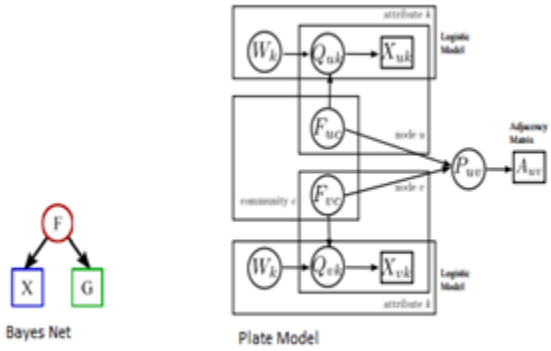
$\sigma_{x,y}[l] = \delta((l\in\mathcal{T}_x)\neq(l\in\mathcal{T}_y)).$ defining the homophilic feature function of the alter-alter nodes, Here 1 extra feature is added to with value 1 in order for the

model to generalize better in case of lack of member profile information based on just the network connectivity

CESNA:

The other model that was profiled was "Communities from Edge Structure and Node Attributes (CESNA)" (Jaewon Yang, Julian McAuley & Jure Leskovec).

It is based on the premise that Graph structure & Node Attributes are formed by latent Communities. The same is depicted by the below plate model.



Fig[1]: BayeNet & Plate Model of CESNA

It makes the below assumptions:

• Nodes that belong to the same communities are likely to be connected to each other.

• Communities can overlap, as individual nodes may belong to multiple communities.

• If two nodes belong to multiple common communities, they are more likely to be connected than if they share only a single common community

• Nodes in the same community are likely to share common attributes

Modeling alters links in ego network:

Probability of nodes u & v are in the same community C is given by

$$P_{uv}(c) = 1 - \exp(-F_{uc} \cdot F_{vc})$$

Where F_{uc} - membership of node u for community C

The adjacency matrix is $A_{uv} \sim \text{Bernoulli}(P_{uv})$. by assuming alter link formation is a generative process

Modeling alter node attribute :

For k[th] attribute of node u could be represented as

$$X_{uk} \sim \text{Bernoulli}(Q_{uk})$$

where,

Q_{uk} is represented as logistic form:

$$Q_{uk} = \frac{1}{1 + \exp(-\sum_c W_{kc} \cdot F_{uc})}$$

W_{kc} is a real-valued weights of logistic model on the membership of a node u parameter for community c

Parameter optimization:

Parameter's are optimized as dual optimization on $\hat{F}$, node's membership and $\hat{W}$ weight vector of the influencing node attributes

$$\hat{F}, \hat{W} = \underset{F \geq 0, W}{\text{argmax}} \log P(G, X | F, W).$$

Node u takes membership of community C when

$$F_{uc} > \delta$$

Where threshold

$$\delta = \sqrt{-\log(1 - 1/N)}$$

Number of community C is determined such that it maximizes likelihood of CESNA on a held out crossvalidation data set

Base Line:

Method	Info	Jaccard similarity					
		Phil	Flickr	Facebook	Google+	Twitter	Avg.
Demon	Net	0.143*	0.098*	0.283*	0.234	0.186*	0.235*
BigCLAM	Net	0.156*	0.092*	**0.347**	0.231	0.246*	0.267*
MAC	Attr	0.069*	N/A	0.190*	0.101*	0.154*	0.133*
Block-LDA	Both	0.082*	N/A	0.241*	0.204*	0.173*	0.178*
CODICIL	Both	0.167*	0.079*	0.263*	0.166*	0.190*	0.218*
EDCAR	Both	0.157*	0.051*	0.222*	0.081*	0.165*	0.177*
Circles	Both	N/A	N/A	0.265*	**0.254**	0.211*	0.183*
CESNA	Both	**0.192**	**0.106**	**0.347**	0.249	**0.249**	**0.282**

[Table:2] Performance of existing model compared against Circles & CESNA on a dataset at http://snap.stanford.edu

It s clear from the above metrics CESNA perform better than the rest. Henceforth CESNA would be used as the base model to compare

Proposed Model: - Community Detection by Ensemble - Supervised model using Edge, Node and Graph Attributes (E-SEGNA)

Unlike the earlier generative model, we propose Ensemble Supervised learning model which uses richer features such as node attributes, edge attributes, edge triads, graph density, connected components and many more, Since we use a supervised learning model, the model tends to capture the user 's behavior in a domain, Hence provide a personalized community membership

Folded community membership graph

Consider a ego network $G(V, E)$, now for a pair of nodes alter nodes $(x, y) \in V \times V$, Let $F = \{f_1, f_2 \dots f_n\}$ is a set of features that the user(or an agent) is influenced to assign these nodes to a shared community.

Then the Probability $P_{xy}(c|F)$, that a pair of alter nodes $(x, y) \in V \times V$, belongs to a community $c \in C$ can be represented as a logistic model

$$P_{xy}(c|F) = \frac{1}{1 + \exp(-\theta^T F)}$$

Here, θ^T is the weight vector corresponding to feature vector F that determines their influence.

Now let, P_{xy}, be the probability that the pair of alter nodes belong to at least one community. Probability that $(x, y) \in V \times V$ doesn't belong to any community can be expressed as

$$1 - P_{xy} = \prod_c (1 - P_{xy}(c|F))$$

Then the Probability that the pair of nodes belong to at least one community is given by

$$P_{xy} = 1 - \prod_c (1 - P_{xy}(c|F))$$

In summary, we use supervised logistic model to predict each entry $C_{xy} \in \{0,1\}$ of the folded community membership adjacency matrix :

$$C_{xy} \sim Bernoulli(P_{xy})$$

Inference:

Now $P_{xy}(c|F)$, the Probability that a pair of alter nodes $(x, y) \in V \times V$, belongs to a community $c \in C$ is expressed a logistic model

$$P_{xy}(c|F) = \frac{1}{1 + \exp(-\theta^T F)}$$

We can find θ^T can be determined by maximizing the likelihood $L(\theta^T) = P_{xy}(C|\theta)$

$$\theta = \text{argmax}_{\theta^T}(1 - \prod_{c \in C}(1 - P_{xy}(c|F)))$$

This could be reduced to

$$\theta = \text{argmax}_{\theta^T}(\prod_{c \in C} P_{xy}(c|F))$$

For log-likelihood above optimization, the above could be represented as

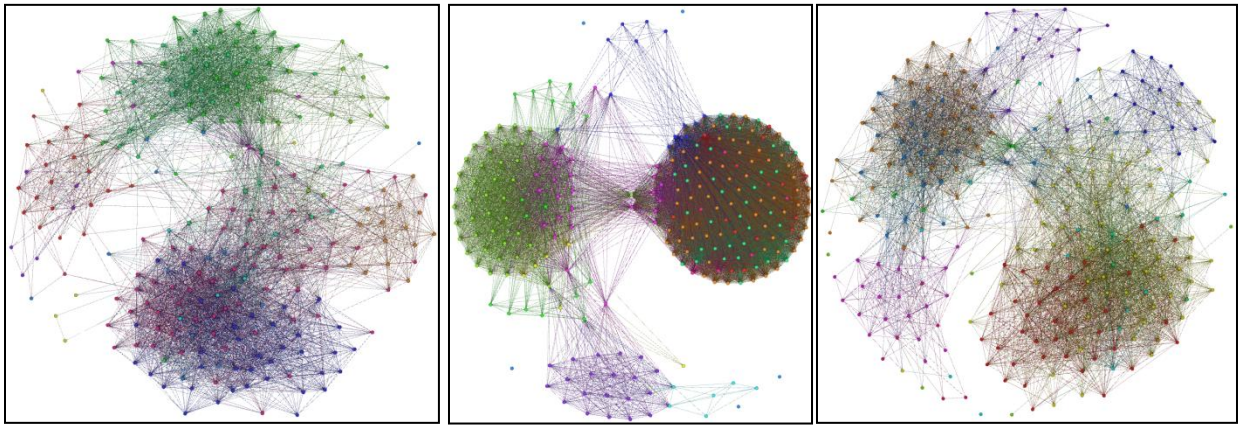
$$\theta = \underset{\theta^T}{\text{argmax}}(\sum_{c \in C} \log(P_{xy}(c|F))$$

If there are m training examples, This could be written as

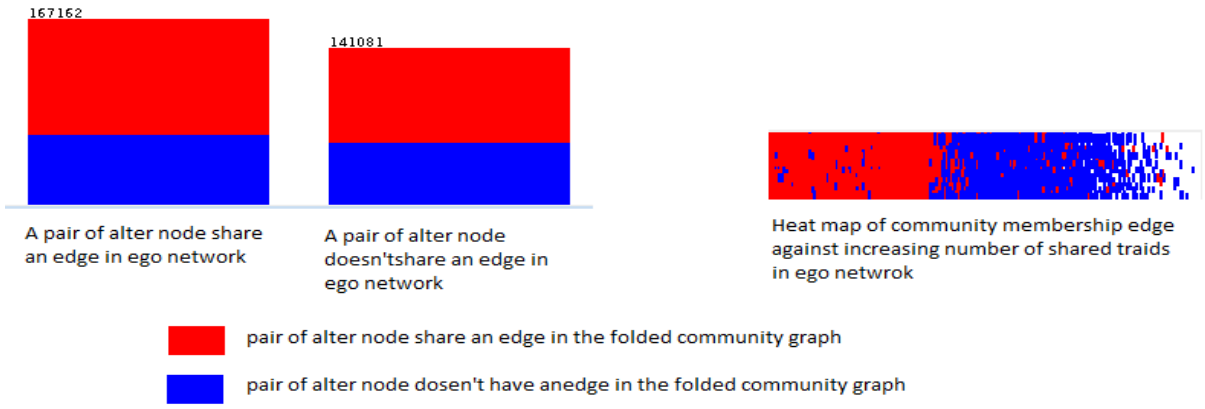
$$\theta = \sum_i^m \underset{\theta^T}{\text{argmax}}(\sum_{c \in C} \log(P_{x^i y^i}(c|F))$$

This is a convex function & can optimized to a global maxima by using algorithm such as Scholastic Gradient Decent. Or Higher order model like SVM

Features & Training Heuristics



Fig[2] is ego network, Fig[3] is actual folded community membership graph, Fig[4] is the predicted folded community membership graph(It could be noted that denser areas is where overlap occurs). The color gradient represent the node circle membership



Fig[5] Distribution of presence of alter-alter edge & shared triads against presence of edge in folded community graph

It would be interesting to note that each of the communities in the folded community sub-graph form a fully connected graph. Hence there are $\frac{N_c(N_c-1)}{2}$ edges in the sub graph.
If we remove edges from this sub-graph . In the worst case the sub graph could loose $(N_c - 2)$ edges & still be strongly connected. While in best case it could loose all but (N_{c-1}) edges & be strongly connected. So the $\frac{N_c(N_c-1)-2}{4}$ is expected number of edge which the sub-graph could loose & still be strongly connected. This is half the original edges. This gives lot of leave-way in choosing the node pair that we want to predict

This is true in non-overlapping communities. For overlapping communities it is assumed the loss of edges is uniform across all nodes and hence the node & edge properties unique to overlapping nodes are maintained relative to others.

As Seen in Fig[5], It was apparent the presence of an edge is not a strong signal for a pair of node to share a community. More richer features such as number of shared triad carries more signal.

Edge selection

We assume all communities have some pairs of alter edge which forms seed for our edge selection. Instead selecting all $\frac{N(N-1)}{2}$ pairs of alter edges, we start with all alter edge nodes and new pair of nodes are added with one node incident on an existing alter edge & the other incident on the a random node which is not in the same connected component, if the connected component is beyond a fixed size. Beside this nodes with less than half of the feature attributes are missing is dropped, since the are very few signals to infer from. This done to reduce the time complexity as well as improve the prediction of the model.

.	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.871	0.254	0.8	0.871	0.834	0.868	0
	0.746	0.129	0.831	0.746	0.786	0.868	1
Weighted Avg.	0.813	0.197	0.815	0.813	0.812	0.868	

[Table:3] Performance on 30% Hold-out set, The test set – has separate set of users & their circle with no overlap of either with the training set, Classifier used: Random Forest

a	B	<-- classified as
30694	4556	| a = 0
7661	22476	| b = 1

[Table:4] Confusion Matrix (before threshold optimization)

.	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.925	0.057	0.911	0.925	0.918	0.866	0.981	0.965	0
	0.943	0.075	0.952	0.943	0.948	0.866	0.981	0.987	1
Weighted Avg.	0.936	0.068	0.936	0.936	0.936	0.866	0.981	0.979	

[Table:5] Performance on 10 fold cross validation with some of test users initial behavior are infered

The threshold is so optimized that False Positive (highlighted in amber) is set to minimum acceptable value while maximizing for True Positive (highlighted in green) . This so done because, as explained above the folded community sub-graph would still be strongly connected with lose of few edges, while we can't afford to have non community edges in the graph.

It was observed that model performs way better if the model , gets to infer initial few community membership mapping for a users it tries to predict, this is clear seen in [Table:5]

Number of Circles

We use supervised regression model to predict the number of circles,

Suppose the user/agent is in influenced by $x = \{x_1, x_2 \dots x_n\}$ signals to decide on the number of circles. We can formulate prediction of number of circles h_θ as regression form as how below

$$h(x) = \sum_{i=0}^{n} \theta_i x_i = \theta^T x,$$

Here θ^T corresponds to the weight of each $x = \{x_1, x_2 \dots x_n\}$ signal in influencing the number of circle selection. Then the cost function $J(\theta)$ that measures the error in prediction with the ground truth is given by

$$J(\theta) = \frac{1}{2} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2.$$

The above could be optimized by solving for optimal θ_j, by using gradient decent as given below

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta).$$

Expanding

$$\theta_j := \theta_j + \alpha \left(y^{(i)} - h_\theta(x^{(i)})\right) x_j^{(i)}.$$

If there are m training examples optimal θ_j could be repeating below until convergence {

$$\theta_j := \theta_j + \alpha \sum_{i=1}^{m} \left(y^{(i)} - h_\theta(x^{(i)})\right) x_j^{(i)} \qquad (\text{for every } j).$$

}

This Linear regression could be replaced with higher order model such as SVR over a ~ Poisson distribution

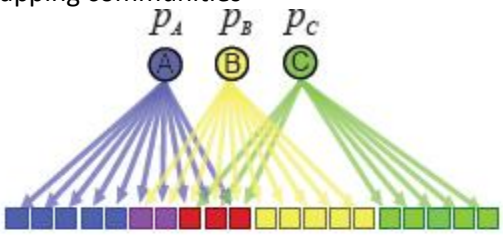
Features to predict number of circles	
Ego Graph	Graph Density
	Number of Edges
	Number of Nodes
	Number of Strong & weak Components
	Number of triads
	Degree distribution type
	Number of connected components
	Average size of connected components
	Mean number of alter node attributes
	Variance in the number of alter node attributes
	Mean matching between alter node attributes
	Ego node attribute features
Folded Community Membership Graph	Graph Density
	Number of Edges
	Number of Nodes
	Number of Strong & weak Components
	Number of triads
	Degree distribution type
	Number of connected components
	Average size of connected components

[Table:6] Features to predict number of circles

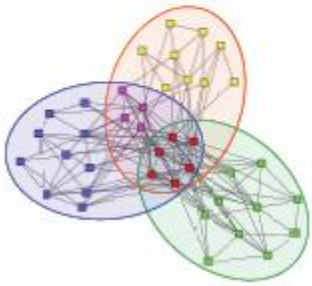
Node, Circle Membership

With the number of circles & folded circle membership is predicted the next step is to predict node circle membership. Since the nature of this graph that overlapping nodes ones that have highest density of connections hence models such as minimum cut or spectral clustering cannot be used.

For this graph we use Community – Affiliation Graph Model for node & community membership, which works well for dense overlapping communities



(a) Community Affiliation Network



(b) Overlap

[Fig:6](a) Bipartite community affiliation graph. $B(V, C, M)$ (b) Predicted Folded Community Membership Graph. $G_{fcm}(V, C_{fcm})$

Given a bipartite graph B(V,C,M), where C is a set of communities, V is a set of nodes, and $(u, c) \in M$ is that, the node $u \in V$ belongs to the community $c \in C$. In our case an edge in the folded community membership graph belong to a particular Community, i.e $C_{uv} \in C$. Also let $\{p_c\}$ be a set of probabilities with which affiliation happens for all $c \in C$. Then the Community-Affiliation Graph Model generates a folded community membership graph $G_{fcm}(V, C_{fcm})$ by creating edge (u, v) between a pair of nodes $(u, v) \in V$ with probability p(u,v)

$$p(u, v) = 1 - \prod_{k \in C_{uv}} (1 - p_k),$$

Community association with Community-Affiliation Graph Model

Given the folded community membership graph $G_{fcm}(V, C_{fcm})$, we aim to detect communities by fitting the AGM (i.e., finding affiliation graph B and parameters $\{p_c\}$ to the underlying $G_{fcm}(V, C_{fcm})$ by maximizing the likelihood $L(B, \{p_c\}) = P(G_{fcm}|B, \{p_c\})$

$$\underset{B,\{p_c\}}{\operatorname{argmax}} L(B, \{p_c\}) = \prod_{(u,v)\in E} p(u, v) \prod_{(u,v)\notin E} (1 - p(u, v))$$

Updating $\{p_c\}$. Keeping the community affiliation network B fixed, we find $\{p_c\}$ by solving the following optimization problem:

$$\arg\max_{\{p_c\}} \prod_{(u,v)\in E} (1 - \prod_{k\in C_{uv}} (1 - p_k)) \prod_{(u,v)\notin E} (\prod_{k\in C_{uv}} (1 - p_k))$$

with the constraints $0 \le p_c \le 1$.By taking log of the above equation we can it a convex form as shown below

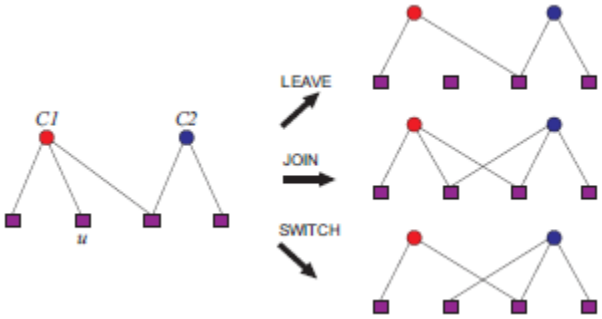
. We maximize the log-likelihood and change $e^{-x_k} = 1 - p_k$ and constrains from $0 \le p_c \le 1$ to $x_c \ge 0$

$$\arg\max_{\{x_c\}} \sum_{(u,v)\in E} \log(1 - e^{-\sum_{k\in C_{uv}} x_k}) - \sum_{(u,v)\notin E} \sum_{k\in C_{uv}} x_k$$

This could be solved by using Stochastic Gradient Decent.

Updating B. For updating B, Metropolis-Hastings is used to stochastically update B using a set of 'transitions'. Given the community affiliation graph B(V,C,M), three kinds of transitions are considered to generate a new community affiliation graph $B'(V, C, M')$

As shown in below figure.

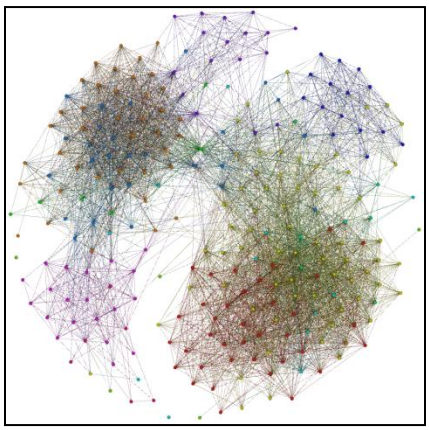
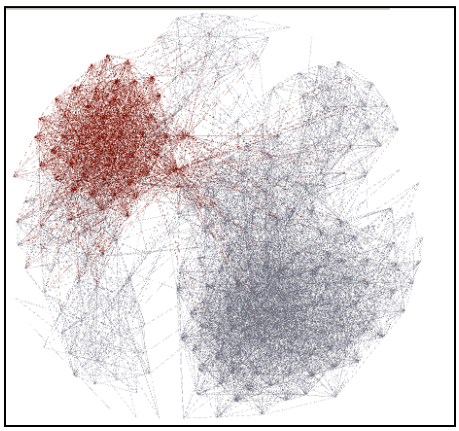
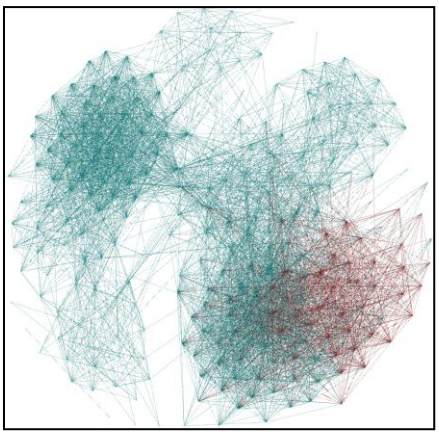


[Fig:7]The 3 'transitions' for updating the community affiliation graph

- LEAVE – randomly a node u, is dropped from the membership of community c.
- JOIN - randomly a node-community pair $(u, c) not\ in\ M$ is chosen and add to M'.
- SWITCH – randomly membership of u is switched from community c1 to c2.

The new community affiliation B'accepted with probability $\max(1, B', \{p_c\})/L(B, \{p_c\})$

By solving these two optimization steps alternatively the optimal parameter $B', \{p_c\}$ can be obtained.

Fig[8] is the predicted folded community membership graph Fig[9] & Fig[10] Generate membership mapping using AGM(red nodes indicate membership to a particular circle) Fig[10] clearly shows overlapping community seperation

Evaluation metrics. To quantify the performance of the predicted set communities' C against the ground truth communities C^*.We adopt an evaluation procedure previously used, Every predicted community is matched with its most similar groundtruth
Community and so is every groundthruth community is matched with its most similar predicted
Community and the score is computed. The final performance is the average of these two metrics. This is formulated as.

$$\frac{1}{2|C^*|}\sum_{C_i^* \in C^*} \max_{C_j \in C} \delta(C_i^*, C_j) + \frac{1}{2|C^*|}\sum_{C_j \in C} \max_{C_i^* \in C^*} \delta(C_i^*, C_j)$$

We use Jaccard similarity as $\delta(.)$., we obtain a score between 0 and 1, where 1 indicates the perfect recovery of ground-truth communities.

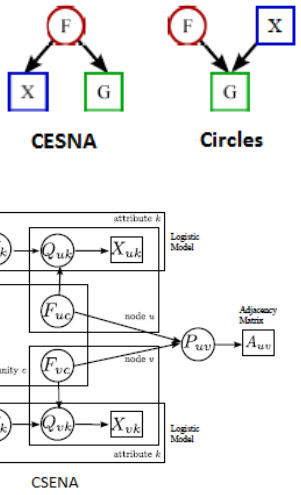
Performance:

.	CESNA	E-SGNA
Cummulative Avg Score:	0.402703	0.433874

[Table:7] Performance of E-SEGNA against CESNA

The results are on 20% of users whose circle data are held out as test set. It clearly shows E-SEGNA performs better than CESNA & has large scope for improvement. As shown earlier if the model can have inference on initial behavior of a user the precision & recall of the model is lot higher for future prediction on the assignment. The model hasn't reached its fullest potential with more refinement & more features engineering the model could perform lot better

Concluding thoughts: CESNA & Circles Vs E-SEGNA

Properties	CESNA & Circles	The proposed model
Latent Structure	Formalizes the Latent Structure 	Model learns the latent structure
Model Type	Un Supervised Generative Model	Supervised Discriminatory Model (Could be extended to semi supervised model if there are not enough training instance to go by)
Model Order	Linear	Could potentially use higher order model like Random Forest or SVM
Model Weights & Membership	Formalizes a dual maximization problem between influence of attributes & community membership	Supervised logistic model to determine the folded community membership between pair of alter nodes, which determines influence of attributes Community – Affiliation Graph Model is used to determine node community membership on the predicted folded community membership graph
Meta parameters	The models picks the number of circle that maximizes the likelihood of the model on a held out set	A supervised regression model on graph global feature and ego attributes to determine the number of circles
Features	Node Attributes & Edge Structure	Local and global Node Attributes &edge properties, global graph properties
	Node Attribute: either alter-alter or alter-ego attributes	Node Attributes: Uses both alter-alter & alter-ego attributes, page rank score etc.,.
	Edge Attribute: alter-alter edge	Edge Attributes: alter-alter edge, number of shared triads, size of the connected component, in-between's, centrality etc.,
		Global Attributes: • Statistics such as mean & variance of number of shared

		attributes between a pair of alters • Number of connected component • Graph density • Degree distribution etc.,
Personalization	Generalized	Personalized. With the expanded set of feature the model tries to capture user or similar user behavior while creating communities and adding membership

[Table:10] Comparision of Circle & CESNA Vs E-SEGNA

Reference:

1. "Learning to Discover Social Circles in Ego Networks" - Julian McAuley & Jure Leskovec -
2. "Community Detection in Networks with Node Attributes" - Jaewon Yang, Julian McAuley & Jure Leskovec
3. J. Yang and J. Leskovec. Overlapping community detection at scale: Anon-negative factorization approach. In WSDM '13, 2013
4. Community-Affiliation Graph Model for Overlapping Network Community Detection - Jaewon Yang & Jure Leskovec
5. S. Lattanzi and D. Sivakumar. Affiliation networks. In STOC '09, 2009.
6. J. Yang and J. Leskovec. Defining and evaluating network communities based on ground-truth. In ICDM '12, 2012.
7. J. Yang and J. Leskovec. Structure and Overlaps of Communities in Networks In SNAKDD '12, 2012
8. K. Miller, T. Griffiths, and M. Jordan. Nonparametric Latent Feature Models for Link Prediction. In NIPS '09, 2009.
9. Y. Ruan, D. Fuhry, and S. Parthasarathy. Efficient community detection in large networks using content and links. In WWW '13, 2013.
10. Y. Sun, C. Aggarwal, and J. Han. Relation Strength-Aware Clustering of Heterogeneous Information Networks with Incomplete Attributes. In VLDB '12, 2012.
11. J. Xie, S. Kelley, and B. K. Szymanski. Overlapping community detection in networks: the state of the art and comparative study. ACM Computing Surveys, 2013.
12. Z. Xu, Y. Ke, Y. Wang, H. Cheng, and J. Cheng. A model-based approach to attributed graph clustering In SIGMOD '12, 2012.