A Novel Approach for Secure Hidden Community Mining in Social Networks using Data Mining Techniques

R. Renuga Devi
Research Scholar
Department of Computer Science
Karpagam University
Coimbatore, India-641021

M. Hemalatha
Department of Computer Science
Karpagam University
Coimbatore, India-641021

ABSTRACT
Social network community contains a group of nodes connected on the basis of certain relationships or same properties. Sometimes it refers to the special kind of network arrangement where the Community Mining discovers all communities hidden in distributed networks based on their important similarities. Different methods and algorithms have been employed to carry out the task of community mining. Conversely, in the real world, many applications entail distributed and dynamically evolving networks. This leads a problem of finding all communities from a given network. Detecting evolutionary communities in these networks can help the user for better understanding the structural evolution of the networks. In this research, first a new bipartisan scheme using k- Dimensional (KD) –Tree to deal with the recursive bisection method is proposed; next an Improved KD-Tree algorithm to deal with the multidimensional problem is put forward. The security issue such as a Sybil attack (Multiple fake Identities attack) arises in these network structures. It can be mitigated by fixing the target time by using SICTF (Sybil Identification using Connectivity Threshold and Frequency of visit) algorithm. The problem faced by the mining community of heterogeneous network can be addressed by using Convergence aware Dirichlet Process Mixture Model (CADPM).

Keywords
Community Mining, Links, Nodes, Social Networks, Sybil,

1. INTRODUCTION
A social network is a social structure between actors, mostly an individual person or an entire organization. Entire network represents the relationship between the actors. Actors are taken as nodes, and the relationship between the nodes are represented as edges. Each node is connected through various social relationships, such as friendship, professional relationship, or organizational relationship, etc. In a social network, a community is represented as a sub-graph, which is present in the network that indicates the connections among nodes. The relationships within the community members are much greater than the relationship among other communities in a network.

Most of the Social communities allow users to freely share information, services, and other useful resources anytime and anywhere. Users on the Internet can create multiple identities to access the system without any limitations. There is a lack in strong user identity, so it makes the open access systems easily vulnerable to Sybil attacks. An attacker can easily create a number of duplicate or fake identities (called as Sybil) to corrupt the system with fake information and affect the accurate performance of the system. Security plays an important role in community mining. In recent years, community mining in a social network is a rising, exciting area of research that has forced it to go a long way, with the involvement of many research fields.

Huge volumes of user generated information are produced on social networking sites every day. This development will be probable to continue with exponentially large information in the future. It is difficult for consumers, service providers, and producers to figure out the administration and consumption of large data and it is difficult to provide their services to the users. Based on the social networking sites, information can be very noisy. Removing the noise from the data is important before starting effective community mining. Most of the Social networking sites are continually evolving and dynamic. Social networking information is unstructured. Getting important information based on this unstructured information from different data sources is a challenging task. Community mining can help the advertisers to find the important people to increase the product advertisement within the budget. Also, it helps various researchers and product sellers to uncover the human activities such as in-group and out-group activities of Users.

This paper is organized as follows: Section 2: Proposed Methodologies, Section 3: Related Works, Section 4: Bipartition method using k- dimensional (KD) Tree, Section 5: Improved KD-Tree algorithm, Section 6: Sybil Identification using Connectivity Threshold and Frequency of visit algorithm, Section 7: Convergence Aware Dirichlet Process Mixture Model, Section 8: Experimental results and Discussions and finally in Section 9: Conclusion.

2. PROPOSED METHODOLOGIES
In this paper a new community bipartisan scheme by using KD-Tree (K Dimensional – Tree) is proposed. KD-Tree overcomes the problem of communication performance optimization and partitioning complexity which are all the throwbacks produced from the recursive bisection strategy [1]. Since the basic KD-Tree does not agree to balancing, the entire tree has to be manually rearranged periodically to improve the balancing. It is not well-situated for dynamically evolving social network communities; those also lead to multidimensional nearest neighbour problems. To overcome this problem an Improved KD-tree with LM algorithm is proposed where the KD-Tree combined with the joint encoding scheme is used to reduce the memory limitations. Though the links in the communities are dense there is a possibility of existence of some security related problems.

Distributed systems from these communities without trusting identities in this community are particularly vulnerable to Sybil (Multiple identities) attacks. This is due to the creation of multiple fake identities from an adversary. Sybil attacks can be avoided by assuming the existence of a trusted authority, which limits the rate of introduction of bogus identities by prompting the users to provide some credentials, like social security number, or by requiring payment. However, such requirements of the network will prevent users from accepting these systems.

In this research, two novel algorithms to avoid the Sybil attack are proposed. First proposed SICT (Sybil Identification using Connectivity Threshold), the connectivity established between each and every node counted in a frequent time interval. The connection threshold is compared with the connection count of each node. If the connection establishment exceeds the threshold, then the node is identified as Sybil node. SICT only considers the hitting time of the nodes. It is not suitable for large scale networks. This problem can be mitigated by using SICTF (Sybil Identification using Connectivity Threshold and Frequency of visit, or hitting of neighbors) algorithm. The maximum variance of connectivity, frequency, and length of a node can be calculated for a particular time interval. The alteration in the variance of the connectivity, length and frequency for a respective time interval can be noted. The maximum variance with respect to connectivity, frequency, and length is said to be Sybil nodes.

On the other hand, many previous researches are related to homogeneous network. In this network, community mining assumes that there is only one kind of relation existing between the nodes in the network, and furthermore, the mining results are independent of the preferences or users' needs. This type of network is said to be homogeneous network, which is shown in the Figure 1. A node or a person indicated as a red color circle may have a friendship with other nodes in a network. No other relationship between the nodes in a network.

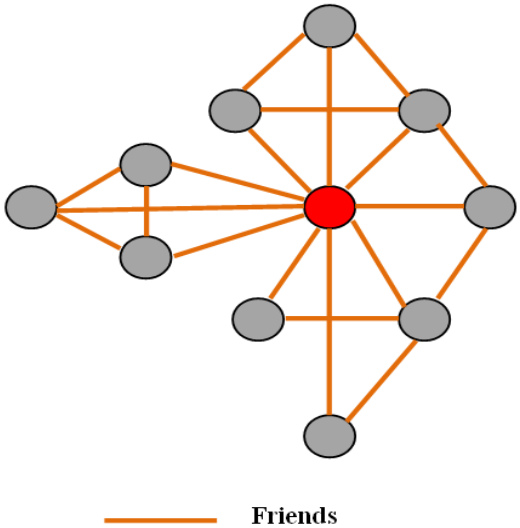

Figure 1: Homogeneous Relationship

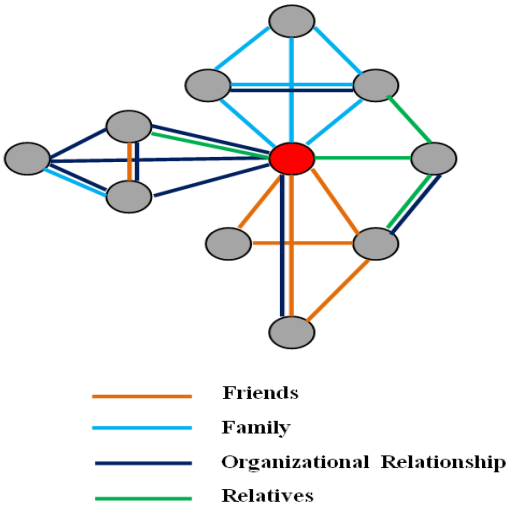

Figure 2: Heterogeneous Relationships

However, there exist multiple heterogeneous social networks, which are shown in Figure 2, each link representing a certain kind of relationship. These relationships may play a distinct role in a particular task. The red colored node or person may have different kind of relationship with other nodes in the same network. So mining such communities is a challenging task. To achieve this goal, this research presents Convergence Aware Dirichlet Process Mixture model (CADPM). Earlier researchers used Dirichlet Process (DP) mixture models which are promising numbers for clustering applications where the amount of clusters is not priorily known [2]. To address this problem CADPM is suggested which can routinely handle millions of data cases.

3. RELATED WORKS

Several methods have been proposed to analyze the social networks. But most of the methods consider the homogeneous networks. Heterogeneous networks are not much considered. Most of the existing methods are not efficient enough to handle heterogeneous information. The data collection is difficult in large scale networks so that the existing methods mainly focused on small networks. In recent days, the Internet usage has grown because of the social media. Many researchers started analyzing the heterogeneous information because a huge volume of information is available in the Internet. Community mining, detection, and clustering in social networks have been considered for a long time. The existing methods are trying to segregate the huge network into several independent parts and combine similar nodes into the same clusters.

Bo Yang., et al, proposed LM (Local Mixing Properties) algorithm to analyze the social network communities by understanding the dynamics of the network. The authors put forward a common framework for analyzing, characterizing, and mining communities in a Social network. LM has been developed which will exactly mine hidden communities in large networks. But LM runs not as efficiently as spectral methods in that ordering time is comparable with networks' scale. Recursive bisection methods are used to identify the actual number of communities in a network. The stopping criterion values are predefined. It does not increase communication performance and also the network partitioning became complex [1].

Several existing Sybil defense methods include a lot of useful and reasonable optimizations that advance the system performance in a particular application scenario. The best examples of Sybil detection methods are SybilGuard method [3] and SybilLimit [4] which have a number of design facilities that make their use in decentralized network environment easy. Similarly, Sum Up method [5] has an optimization strategy to online secure voting systems. On the other hand, the major goal of the Sybil protection method is to find out the center graph partitioning algorithm. Generally, open access systems supported a simple authority like CAPTCHA or process puzzles to take the edge off the Sybil attack [6], [7], [8]. But despondently, these solutions can only limit the speed with that the assailant will commence Sybil identities into the system rather than the full range of such identities.

In most cases, community mining problem has some similar properties to the graph cut problem. Network community mining and analyzing problems are considered as a graph mining. Community mining problem is called as sub graph identification [9]. First community mining and detection research started with the homogeneous social networks. Next few existing methods, such as modularity based methods [10, 11], spectral clustering algorithms, based on probabilistic models, bipartite networks, and recently on heterogeneous networks [12, 13] has been proposed. The heterogeneous network along with star network method was proposed for real world networks. It was differentiated from other existing methods and other community detection and mining methods. It mainly focused on the model of the dynamic development of the net cluster based multi relational communities. In general at every time new nodes may join the dynamic network, some nodes may leave from the network. Finding evolutionary clusters of such dynamic network sequences will help people to understand the development of communities in the network [13].

Dirichlet method provides the simplest way to feature priors to the cluster range of mixture models, and this is often terribly useful to make a decision the cluster range mechanically. In recent times, few works have extended the existing Dirichlet method into allowing for time data. A different DP-based extension considered to model organic mode cluster [14, 15]. The proposed algorithm provides a particular answer to net-cluster evolution in heterogeneous networks and author outlined a unique generative model for net-cluster evolution. This may model the evolution of constant cluster in several timestamps; whereas several existing works need constant clusters do not change among totally dissimilar timestamps. Ending model doesn't claim a worldwide reasoning of the model, however greedy reasoning at every time stamp, is additionally sensible for timely change of the evolution [16].

4. BIPARTITION METHOD USING K- DIMENSIONAL (KD) TREE

A social network can be described as a graph $G = (V, E)$, where $V = \{V_1, V_2, V_3, ...V_n\}$ the set of vertices and E is the set of edges concerning pairs of vertices. Each edge symbolizes the social relationships between two nodes representing individuals. A network community contains a group of nodes connected based on certain relationships or same properties sometimes that refers to a special sort of network arrangement where the mining community is discovering all communities hidden in distributed networks based on their relevant local outlooks. To this point, different methods and algorithms have been employed to carry out the task of community mining. Conversely, in the real world, many applications entail distributed and dynamically evolving networks. This leads to a problem of finding all communities from a given network.

To avoid this problem, the existing work presented a novel model for characterizing network communities via introducing a stochastic process on networks and analyzing its dynamics based on the large deviation theory. Using the basic properties of local mixing, then proposed an efficient implementation for that framework, called the LM (Network community mining based on Local Mixing properties) algorithm, to practically solve large-scale NCMPs. There are also some drawbacks identified that are: By combining with a predefined stopping criterion, the actual number of communities is estimated by using a recursive bisection method. The main disadvantage of the recursive bisection method is one in which it does not optimize communication performance and the complexity of performing the partitioning. To solve these problems in proposed work a new community bipartisan scheme is developed by using KD-Tree. Also, the stopping criterion is calculated automatically by efficiently determining the minimum Eigen-gap without explicitly computing eigenvalues.

Generally, KD-Tree is a multidimensional binary search tree, assumed for managing spatial data. On the other hand, it is also helpful in various applications like graph partitioning, database applications, and n - body replications. The KD-Tree is used to optimize the finding of the closest centroid for all patterns. The basic idea is to group patterns with associated coordinates to attain group assignments, whenever possible, devoid of the explicit distance computations for each of the patterns. Through a pre-processing step the input patterns are prearranged in a KD-Tree. On each iteration the tree is traversed and the patterns are assigned to their closest centroid. The stopping criterion is computed based on the Eigen gap. The proposed LM with KD-Tree algorithm is given in Table 1. The experimental result shows that the proposed scheme is more effective and scalable when compared with the existing scheme.

Table 1: LM with KD-Tree Algorithm

Step 1: Select attractor (Column Selection)
Step 2:

$$c = \arg_i \max \pi_i = \arg_i \max\{\frac{d_i}{\sum_k d_k}\} = \arg_i \max\{d_i\}$$

Step 3: Calculate OTD $p_{i,c}^{(t)} = \frac{1}{d_i} \sum_{(i,j)\in E} A_{ij}, p_{i,c}^{(t-1)}$

Step 4: Bipartition of $p_{i,c}^{(t)}$

Step 5: Stopping Criterion $Q = \sum_i e_{ij} - a_i^2$

Step 6: If a stopping criterion is satisfied
Step 7: Return result
Step 8: Else Goto step3.
Step 9: procedure call build KD-Tree
Step 10: Build KD-Tree (ReferencePts, []);
Neighbour point (D-dimensional Euclidean, 2-norm distance)
Step 11: initial value of centroids
Step 12: calculate point to cluster centroid distances
Step 13: Euclidean $d(p,q) = \sqrt{\sum_{i=1}^{n}(q_{i-} p_i)^2}$
Step 14: cl=cluster d (p, q)
Step 15: plot all the points in the KD-Tree
Step 16: calculate eigen-gap

$$\Delta_K = (\lambda_K(p(\theta) - \lambda_{K+1}(p(\theta))$$

Step 17: if large eigen-gap for $p(\theta)$ is met
Step 18: End process
Step 19: Else
Step 20: go to Step 11.

5. IMPROVED KD-TREE ALGORITHM

The network community mining problem is the foremost limitation in social network communities that should be avoided. To avoid this problem earlier a novel model for characterizing network communities is suggested by introducing a stochastic process on networks and analyzed its dynamics based on the large deviation theory that method called as local mixing (LM) algorithm where it performs Network community mining based on Local Mixing properties. But this also leads the problems like the actual number of communities is estimated by using a recursive bisection strategy. To solve these problems a new community bipartisan scheme is developed by using KD-Tree that works quite well for small dimensions. When the number of the searched nodes grows with the space dimension K-D tree may turn out to be too time-consuming. One more drawback is that the basic K-D tree does not have the same opinion to balancing; it has to physically rearrange the whole tree periodically to get better balancing, which is not well situated for dynamic evolving social network communities. That also leads to multidimensional nearest neighbour problems. To overcome all the above mentioned problems an improved KD-tree with LM algorithm is proposed where the KD-Tree is combined with the joint encoding scheme to decrease the memory usage and limitations. Also, the stopping condition is calculated automatically by efficiently determining the minimum eigen-gap without explicitly computing eigenvalues. The experimental results show that the improved KD-tree method is more effective and scalable than the classical KD-tree method. The proposed Improved KD-Tree algorithm is given below in Table 2.

Table 2: Improved *k* – Dimensional (KD) Tree algorithm

Step 1: Create an Improve KD-Tree for the given data $x_i, i = 1,.....n.$
Step 2: For $j = 1,.....q.$ calculating the rank density p_j of each leaf bucket L_j.
// Where q is the leaf buckets in KD-Tree.
Step 3: Calculate mean value, m_j
Step 4: Choose $c_1 = m_z, where\ z = \arg_j \max\ p_j$
Step 5: For $t = 2,.....k,$ and For $j = 1,.....q$
Evaluate $g_j = \{\min_k = 1,.....t[d(c_k, m_j]), P_j\}$
$c_t = m_2\ where\ z = \arg_j \max(g_j).$
Step 6: go to step 3, until the convergence criteria is met and calculate a second possible list of K initial centuries $(c_1, c_2, ...c_K).$
Step 7: Return $(c_1, c_2, ...c_K).$
Step 8: End

6. SYBIL IDENTIFICATION USING CONNECTIVITY THRESHOLD AND FREQUENCY OF VISIT

Community mining in a network has also been aware of some certain types of attacks. Mostly on Distributed system attacks create some bogus identities and pollute the system with fake information can be done. These types of attacks are known as Sybil attacks. In the previous method Sybil identity has been found by using various methods which consider trusted agency certified identities. However, this method applicability is neither justifiable nor practically realized in large scale distributed systems. Recently in social networks, defending against Sybil attack has been one of the increasing interest of research. Some works used Sybil Infer for detecting Sybil nodes in social networks.

In this research, first the researcher proposed Sybil Identification using a Connectivity Threshold algorithm (SICT algorithm), to analyze the nodes in the given network, the connection Cn established for each node in the network. Every network community consists of honest h nodes as well as Sybil or attacker node s1. The connections created between each and every node is counted in a regular time interval. The connection threshold is compared with the connection count of each node. If the connection establishment is exceeding the threshold then the node is identified as Sybil node. But it has few limitations. In order to overcome the proposed algorithms drawbacks again an algorithm was offered. Sybil identities can be identified by setting a target time interval by using SICTF (Sybil Identification using Connectivity Threshold and Frequency of visit, or hitting of neighbors) algorithm.

In network community the linkage and relationship among each node dynamically increase. The security issue such as Sybil attack arises in these network structures. This attack can be mitigated by fixing the target time. With this time limit the connection made by each user on a network community can be evaluated. If a particular user creates a higher connection in a given time limit, then that user is identified as Sybil and mitigated. Because SybilGuard suffers from high false negatives, it may introduce the nodes of Sybil without being detected. Comparing the proposed SICTF with existing Sybil guard, the SICTF detects the Sybil users accurately in the network. The proposed SICTF algorithm is given below in Table 3.

SICTF can detect Sybil attacks by observing hitting event distribution of a suspect node over a period of time intervals. It randomly selects nodes from the originator node with length and observing the hitting event of each node with its neighbors in the periodical time. The neighboring nodes of observed nodes within their length also observed in terms of hitting event value with their neighbor lists. It repeatedly selects the random nodes until the stopping criterion is met. The measurement of average hitting event all time interval is calculated for each node based on hitting count with its neighbors.

After the measurement is over, the collected measurements from all nodes, the node can locally compute an estimated hitting event of all neighbor nodes, if hitting event exceeds a predefined threshold for any nodes, that node is considered as Sybil node. If identifying Sybil through hitting event, the false positive is reduced than the existing random walk algorithm.

Table 3: The Proposed Sybil Identification Using Connectivity Threshold and Frequency of visit (SICTF) Algorithm

Step1: Consider the established connection C_n
Step2: Set $t_i = 0$ // Initial time
Step 3: Do $t_i = t_{i-1} + \Delta t$ where i>0
//Changing time interval
Calculate $C_{n(i)}$// follow algorithm 1
Calculate l $(C_{n(i)})$l$(C_{n(i)}) = \frac{\sum \forall (X_{i,j}=1)}{Cn(i))}$
//Length of a established connection in a given time interval
Calculate f $(C_{n(i)})$
F (n') = $\sum v(n_i)$ //frequency of the node
f(n)→ $\sum f(n')$
f $(C_{n(i)}) = \frac{f(n)}{Cn(i))}$
While(t_i = x) // x is target time
Step4: If
$t_i \rightarrow \forall$ (max(($C_{n(i)}$) ,l ($C_{n(i)}$) , f ($C_{n(i)}$))) then
$S_i \leftarrow \forall$ (max (($C_{n(i)}$) ,l ($C_{n(i)}$) , f ($C_{n(i)}$)))
// Connection made by Sybil is found.
Step 5: End

7. CONVERGENCE AWARE DIRICHLET PROCESS MIXTURE MODEL

Recently the social network has attracted much attention. Community mining in social networks is one of the major research areas in social network analysis. Therefore, Network community mining is still a major problem that should be avoided. Most of the existing methods of community mining assume that there is only one kind of relation in the network, and moreover, the mining results obtained from such kind of networks are independent of the users' needs or preferences. However, there exist multiple heterogeneous social networks, each one representing a specific type of relationship. These relationships may play a distinct role in a particular task. So mining such communities is a challenging task.

To achieve this goal, the current work presents Convergence aware Dirichlet Process Mixture Model (CADPM). Earlier research used Dirichlet Process (DP) mixture models which are promising candidates for clustering applications where the number of clusters is unknown a priori. These models are unluckily not appropriate for large scale data mining applications because of few computational considerations. To overcome this problem the CADPM algorithm is proposed which can routinely handle millions of data. The proposed CADPM algorithm is given below in Table 4.

Table 4: The Proposed Convergence Aware Dirichlet Process Mixture Model (CADPM)

Step 1: Find the log-likelihood of all the observations to calculate similarity for clustering
Step 2: Initialize object Oi
Step 3: Construct mixture model

$$O_i \sim \sum_{K=1}^{K} \Pi_K P(O_i | Z_i = K \text{ Oi}$$

Step 4: Define the DPM model

$$O_i \mid \phi_i f(\phi_i)$$

$$O_i \mid G \sim G \rightarrow (2)$$

$$G \sim DP(G0, \infty)$$

Step 5: Finite infinite number of clusters with cluster number k

$$O_i \mid Z_i \{\phi_K\}_{K=1}^{K} \sim f(\theta_{zi})$$

$$Z_i \mid \pi \sim Drichlet(\pi_1, \ldots\ldots \pi_k)$$

$$\phi_k \sim G0$$

$$\pi \sim Dirichlet(\alpha \mid K, \ldots . \alpha \mid K)$$

Step 6: $Dirichlet(v, q, z)$

Step 7: End

8. EXPERIMENTAL RESULTS AND DISCUSSIONS

The proposed algorithm's performance is evaluated using two datasets, namely, Dolphin and Wiki. With the aim of analyzing and comparing the performance of the above algorithms, the metrics like clustering accuracy, Precision, Recall, F-Measure, ROC of True Positive Rate, False Positive rate were used. In heterogeneous community mining we have taken three large scale data sets. Those are Amazon data set, Gnutella data set, and Stanford data set. The Amazon data set was collected from the Amazon website. This contains information about the customers and their purchase information. The Gnutella data set consists of a series of snapshots of the peer-to-peer file sharing network information

from the year of 2002 August. The nodes represent the hosts in the Gnutella network topology and the edges between the nodes represent the connections between the hosts. The Stanford data set consists of the page information of the

Stanford University (i.e. stanford.edu). Nodes represent the pages and edges represent hyperlinks between them.

The result obtained for the proposed system based on performance evaluation parameters such as accuracy, precision, recall, f-measure, True and false positive rates is discussed below:

The advocated KD-tree algorithm shows 0.82 of precision rates, 0.67 of recall values for wiki dataset and 0.67 of precision, 0.61 of recall values for dolphin datasets. The evaluated values of the proposed method are higher than the LM algorithm. The overall accuracy of the proposed KD - Tree provides 86.2% for wiki dataset and 82% for dolphin dataset. The result obtained from this shows that KD-tree for community mining shows higher precision, recall, the accuracy value than with LM algorithm.

The proposed Improved KD-tree shows 0.78 of precision, 0.62 of recall value of wiki dataset and 0.96 of precision, 0.69 of recall values for dolphin datasets. The result obtained for proposed Improved KD-tree is higher than LM algorithm and KD-tree algorithm. The overall accuracy of proposed improved KD-Tree provides 88.77% for wiki dataset and 91.94% for dolphin dataset. The result obtained from this shows that improved KD-tree for community mining shows higher precision, recall, and accuracy values than with LM algorithm and KD-Tree algorithm.

The proposed improved KD tree with SICFT algorithm shows 0.97 of precision, 0.75 of recall for wiki dataset and 0.98 of precision, 0.80 of recall for dolphin datasets. The result obtained from a proposed improved KD tree with SICFT algorithm higher than an Improved KD tree with SICT algorithm and Sybil Guard algorithm. The overall accuracy of proposed improved KD tree with SICFT algorithm provides 95.16% for wiki dataset and 96.77% for dolphin dataset. The result obtained from this shows that improved KD-tree with a SICFT algorithm for community mining under Sybil attack can be mitigated by showing higher precision, recall, the accuracy value than an Improved KD tree with SICT algorithm. The comparative results are given in Table 5 and the graphical representation of accuracy value is given in Figure 3, Precision Value is given in Figure 4 and the Recall value is given in Figure 4.

Table 5: Comparative results of proposed algorithms in terms of Accuracy, Precision, and Recall values for Wiki and Dolphin Datasets

Performance Measurement	Wiki Dataset			Dolphin Data set		
	LM with KD-Tree Algorithm	LM with Improved KD-Tree Algorithm	Improved KD-Tree with SICTF Algorithm	LM with KD-Tree Algorithm	LM with Improved KD-Tree Algorithm	Improved KD-Tree with SICTF Algorithm
Accuracy	86.2	88.77	95.16	82	91.94	96.77
Precision	0.67	0.78	0.97	0.82	0.96	0.98
Recall	0.61	0.62	0.75	0.67	0.69	0.80

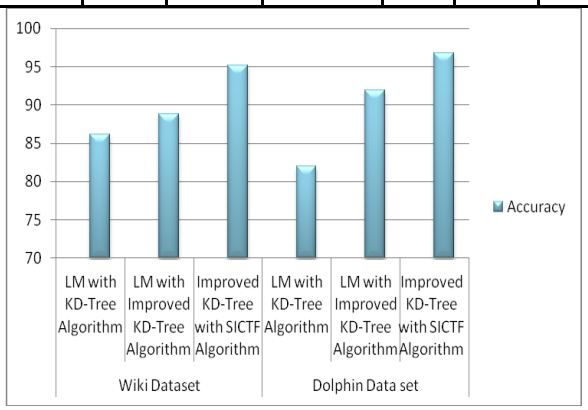

Figure 3: Accuracy value Comparison

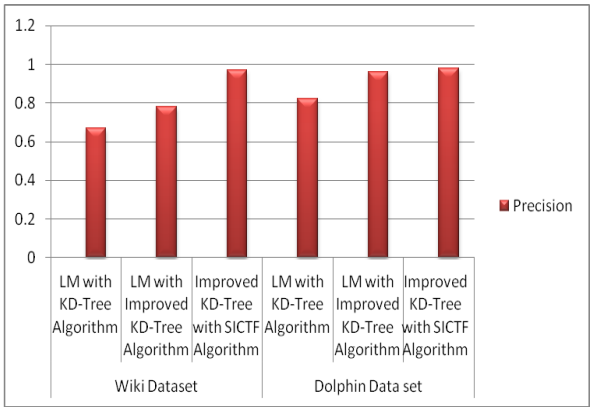

Figure 4: Precision Rate Comparison

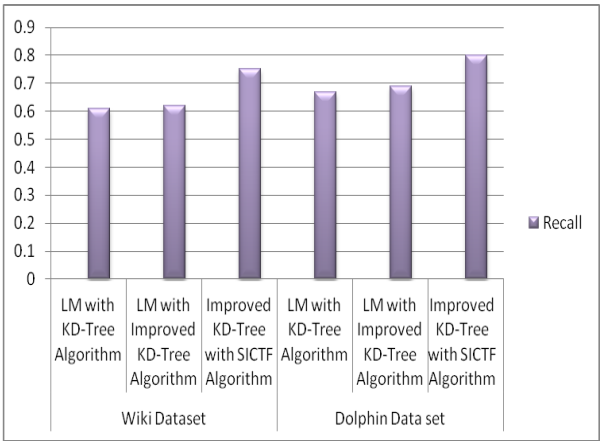

Figure 5: Recall Rate Comparison

The proposed Convergence aware Dirichlet Process Mixture Model shows precision rate 0.95 for Amazon dataset, 0.96 for Gnutella dataset and 0.95 for Stanford dataset. The recall rate for Amazon dataset is 0.95, Gnutella dataset is 0.96 and 0.95 for Stanford dataset. The proposed Convergence aware Dirichlet Process Mixture Model provides 94.5 % of accuracy for the Amazon data set, 95.75% for the Gnutella dataset, and 94.6% of accuracy for the Stanford dataset. The result obtained from this shows that Convergence aware Dirichlet Process Mixture Model for heterogeneous network community mining shows higher precision, recall and accuracy rate than the rate of Dirichlet Process. The comparative results are given in Table 6 and the graphical representation of accuracy value is given in Figure 6, Precision Value is given in Figure 7 and the Recall value is given in Figure 8.

Table 6: Comparative results of proposed CADPM Algorithm in terms of Accuracy, Precision, and Recall values for Amazon, Gnutella and Stanford Datasets

Performance Measurement	Amazon Dataset		Gnutella Dataset		Stanford Dataset	
	Dirichlet Process Model	CADPM Algorithm	Dirichlet Process Model	CADPM Algorithm	Dirichlet Process Model	CADPM Algorithm
Accuracy	87.6	94.5	92.45	95.75	89.5	94.6
Precision	0.88	0.95	0.92	0.96	0.9	0.95
Recall	0.88	0.95	0.92	0.96	0.9	0.95

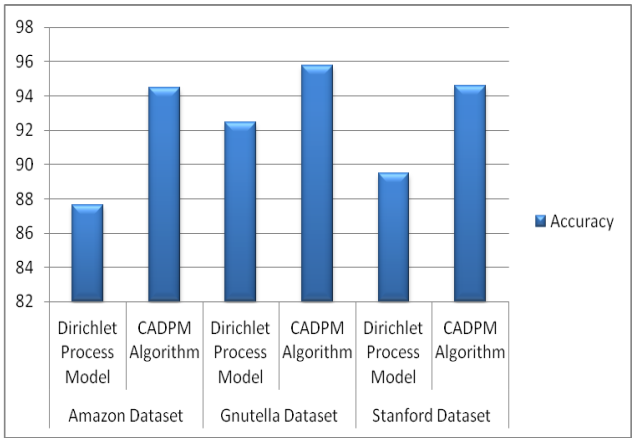

Figure 6: Accuracy Value Comparison

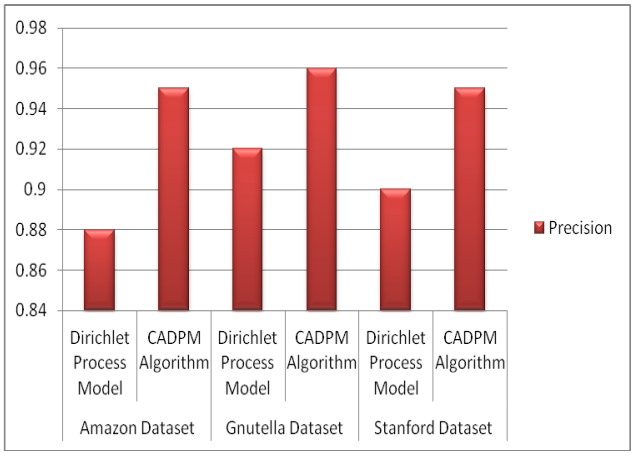

Figure 7: Precision Rate Comparison

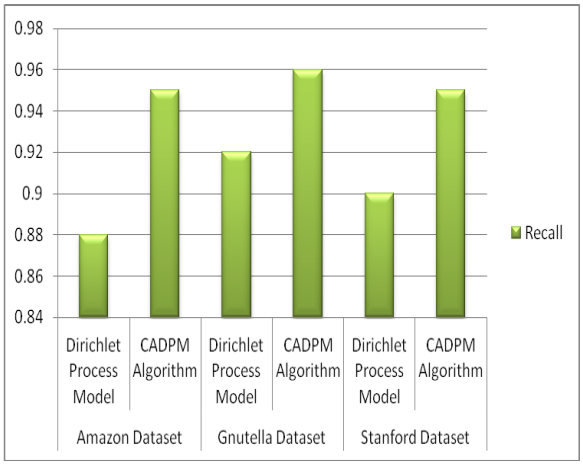

Figure 8: Recall Rate Comparison

9. CONCLUSION

In this work we have given four main contributions. First, the KD-Tree based clustering that produce good clustering results. In that a new criterion to avoid the complexity of the clustering process is established. The criterion optimizes the quality of the target clustering. Then best partition-based speedup scheme of bidirectional search accelerated in both

directions with KD-tree. In Second contribution of work improved KD-Tree based clustering for producing good clustering results has been discussed. This is accomplished by choosing the gap as the clustering quality, by adding the eigengap, as regularization term. The best partition-based speedup scheme is a bidirectional search, accelerated in both directions with improved KD-tree. The memory constraints are reduced by using the join coding scheme. In third contribution of work the Sybil attack in community network has been considered. This has been mitigated by setting the target time. The community can be generated and nodes of Sybil can be found by determining the maximum connections of nodes created by the user. The final contribution of work can be done by the mining community of heterogeneous network can be addressed by using Convergence aware Dirichlet Process Mixture Model (CADPM). This solves the problem by routinely handling millions of data cases.

10. REFERENCES

[1] Yang, B., Liu, J., and Feng, J. 2012. On the spectral characterization and scalable mining of network communities. Knowledge and Data Engineering, IEEE Transactions On, 24(2), 326-337.

[2] Tang, L., Liu, H., Zhang, J., and Nazeri, Z. 2008. Community evolution in dynamic multi-mode networks. In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM. 677-685.

[3] Yu, H., Kaminsky, M., Gibbons, P. B., and Flaxman, A. 2006. Sybilguard: defending against Sybil attacks via social networks. ACM SIGCOMM Computer Communication Review, 36 (4), 267-278.

[4] Yu, H., Gibbons, P. B., Kaminsky, M., and Xiao, F. 2008. Sybillimit: A near-optimal social network defense against Sybil attacks. In Security and Privacy, 2008. SP 2008. IEEE Symposium on IEEE. 3-17.

[5] Tran, D. N., Min, B., Li, J., & Subramanian, L. (2009, April). Sybil-Resilient Online Content Voting. In NSDI. 9(1): 15-28.

[6] Walsh, K., and Sirer, E. G. 2006. Experience with an object reputation system for peer-to-peer file sharing. NSDI. Proceedings of the 3rd conference on 3rd Symposium on Networked Systems Design & Implementation.

[7] Peterson, R., and Sirer, E. G. 2009. AntFarm: Efficient Content Distribution with Managed Swarms. In NSDI. 9(1): 107-122.

[8] Piatek, M., Isdal, T., Krishnamurthy, A., and Anderson, T. E. 2008. One Hop Reputations for Peer to Peer File Sharing Workloads. In NSDI. 8(1): 1-14.

[9] Cai, D., Shao, Z., He, X., Yan, X., and Han, J. 2005. Mining hidden community in heterogeneous social networks. In Proceedings of the 3rd international workshop on Link discovery. ACM. 58-65.

[10] Newman, M. E., and Girvan, M. 2004. Finding and evaluating community structure in networks. Physical review E, 69(2), 026113.

[11] Newman, M. E. 2006. Finding community structure in networks using the eigenvectors of matrices. Physical review E, 74(3), 036104.

[12] Sun, Yizhou, and Jiawei Han, 2010. Integrating Clustering with Ranking in Heterogeneous Information Networks Analysis. Link Mining: Models, Algorithms, and Applications. Springer New York, 439-473.

[13] Sun, Y., Yu, Y., and Han, J. 2009. Ranking-based clustering of heterogeneous information networks with star network schema. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM. 797-806.

[14] Xu, T., Zhang, Z., Yu, P. S., and Long, B. 2008. Dirichlet process based evolutionary clustering. In Data Mining, 2008. ICDM'08. Eighth International Conference on IEEE. 648-657.

[15] Xu, T., Zhang, Z., Yu, P. S., & Long, B. 2008. Evolutionary clustering by a hierarchical Dirichlet process with the hidden Markov state. In Data Mining, 2008. ICDM'08. Eighth International Conference on IEEE. 658-667.

[16] Sun, Y., Tang, J., Han, J., Gupta, M., and Zhao, B. 2010. Community evolution detection in dynamic heterogeneous information networks. In Proceedings of the Eighth Workshop on Mining and Learning with Graphs. ACM. 137-146.