Impact Factor (SJIF): 3.632

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444

Volume 3, Issue 4, April-2016

Enhancement Network Security On Centralized Social Networks by Increment Clustering

Mr.Mrunalkumar Khairnar[1],Ms.Dipali Hadiyel[2], Mr.Sunny Thakur[3]

Information Technology, Sigma Institute of Technology, Vadodara, India[1 2 3]

Mr.Romil Patel[4]

Assistant Professor, IT, Sigma Institute Of Technology, Vadodara, India[4]

Abstract: -In this 21st century, rapid growth of internet peoples uses too much social network sites to communicate, share data. Social networks like Instagram, YouTube, Pinterest, Flickr, Reddit, LinkedIn etc. This all site store private and public data like their IDs, Date of Birth, ph_ no, address, comment, share data information To provide security in networks for secure communication is major problem. Networks are divided into different users and find annoying users and make cluster. Our main aim is to provide better security and performance. In centralized our incremental clustering algorithm is better than sequential clustering algorithm which is based on the SaNGreeA algorithm. Security is provide using checksum algorithm. Different algorithm for clustering are implement on this paper.

Keywords: - Social Networking, Sequential Clustering, Incremental clustering, Security.

I. INTRODUCTION

Clustering is unsupervised learning. Clustering is process to divided whole datasets into small part based on their category, similarity, distance. The social network is describe as a set of different attribute and their relations between the different users like users ids relation etc. A social network provides the personal or private information for different attributes to the users and connect them related to their relation, which may be relations of transaction, communication. Networks are shown by the different graphs, where the nodes of the graph have structural information. It represent relations between them [1].The social networks may be more difficult than the network. For example, a bank transaction network [12], the graph would be directed .If the interaction contains more than two users there interaction which are in form of labels and nodes in graph such as age, gender, location, or Jobs which could enrich and shed light on the structure of the network [1][13].Using clustering they divide into different cluster based on their age, gender, zip code. For hiding the details of users use k-anonymizing properties. In which user id are create in bunch of group so particular user not find easily. Like we have user_id 12, 16, 10, 22, 43, 56, 67, 44, 65, 43, 27, 20, 15.Using this id anyone can easily get details, So using k-anonymizing properties they group in[10-30], [31-50], [50-70] so finding particular user id and there details not easily found.

II. EXISTING SYSTEM

In the existing system provide security is difficult.it is not suitable to large large network or tabular data. In Sequential clustering time to build cluster is very high and find user details is easy because of no security and any user(hacker) can hack the data and misuse other person personal details. So this major part are in existing system [15] [1].

III. PROPOSED SYSTEM

In this project our main aim is to represent the escalate method for improve security of Centralized Social Networks by incremental Clustering with improved dependency and performance [15].This is implementation paper which is based on our review paper [15]:

- Block anonymizing users using security
- To show the new framework and methods.
- To show high reliable and easy system performances related to algorithm.
- Provide less time complexity
- To find comparative analysis of current and proposed (incremental) algorithm in order to claim of efficiency and time complexity.

IV. SYSTEM IMPLEMATION

System flow:-

- First select proper dataset which is in .xml file format convert into readable form using Microsoft excel.
- This selected input dataset file take and give cluster size as input and make cluster.
- Perform partition ,add or delete node and modified clustering
- Perform sequential clustering algorithm
- Find the information loss and computation sum
- Same flow for incremental clustering algorithm
- Compare both sequential clustering algorithm and incremental clustering algorithm in manner of time complexity.

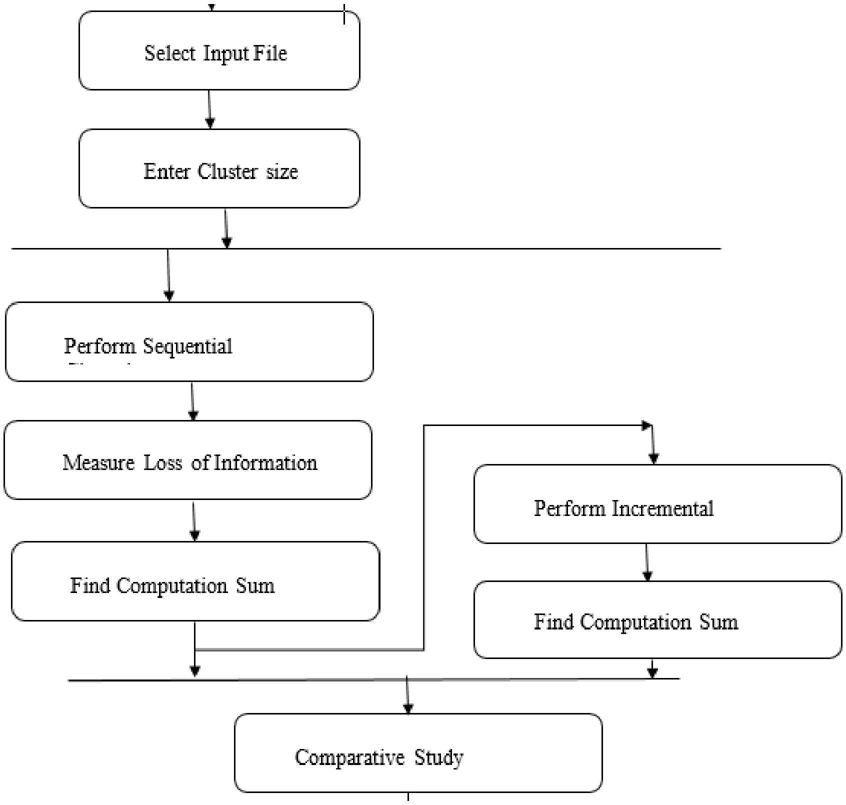


Fig 1. System Flow

A. Sequential Algorithm:-

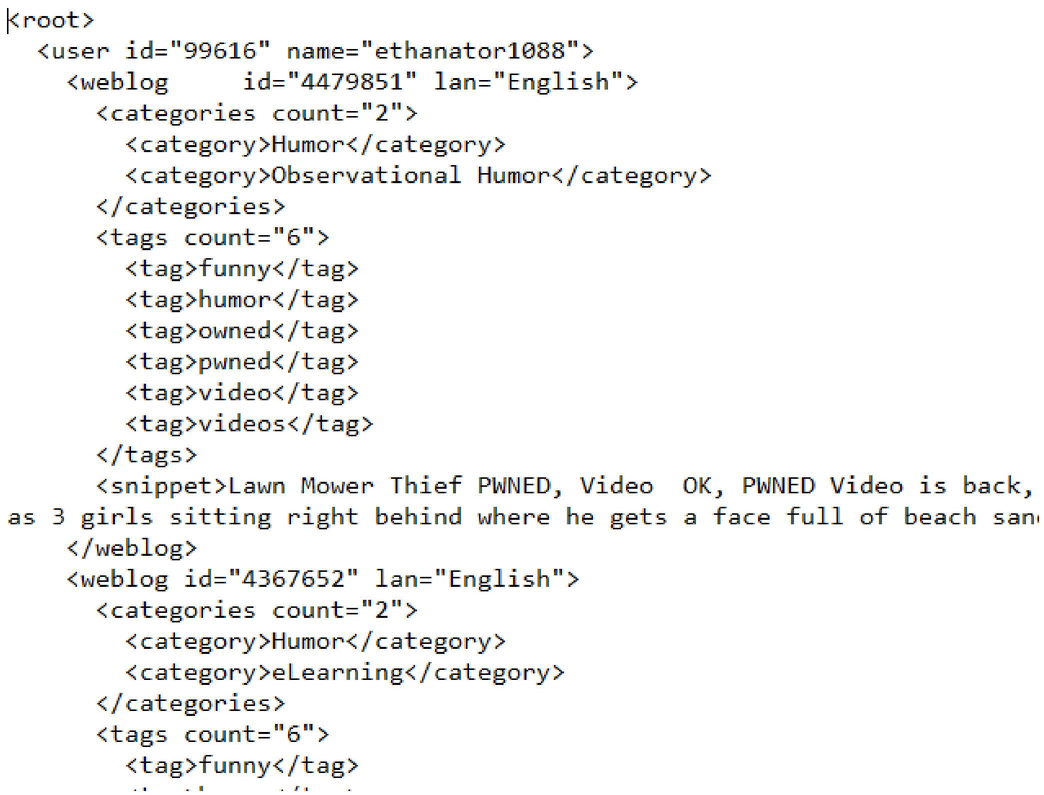
Input: Nodes of social network, an numeric m
Output: Size of cluster $\geq$ m
Step 1. Elect a approx partition of data cluster of vertices
Step 2.
For n=1,………...N do;
Let cluster C[i] which is belong to current vertices v[n].
Step3. For each cluster C[d], $d \neq i$, find different information loss if vertices V[n] move from one cluster C[i] to another cluster C[d].
Step 4.If one cluster is singleton, move vertices V[n] from one cluster C[i] to another cluster C[d] and delete the first cluster C[i].
Step 5.Else, if information loss is less than zero then move vertices V[n] from one cluster[i] to another C[d].
Step 6.If cluster size is greater than m split each of them randomly into two equally sized clusters
Step 7.If at least one node move during the last loop, go to step 2 to 6.
Step 8.while current cluster size smaller than m, select one of them and bring together it with the cluster which is nearest.
Step 9: Outcome cluster is output.

B. Incremental cluster:

Input: Set of nodes (N), Number of cluster (k) , threshold

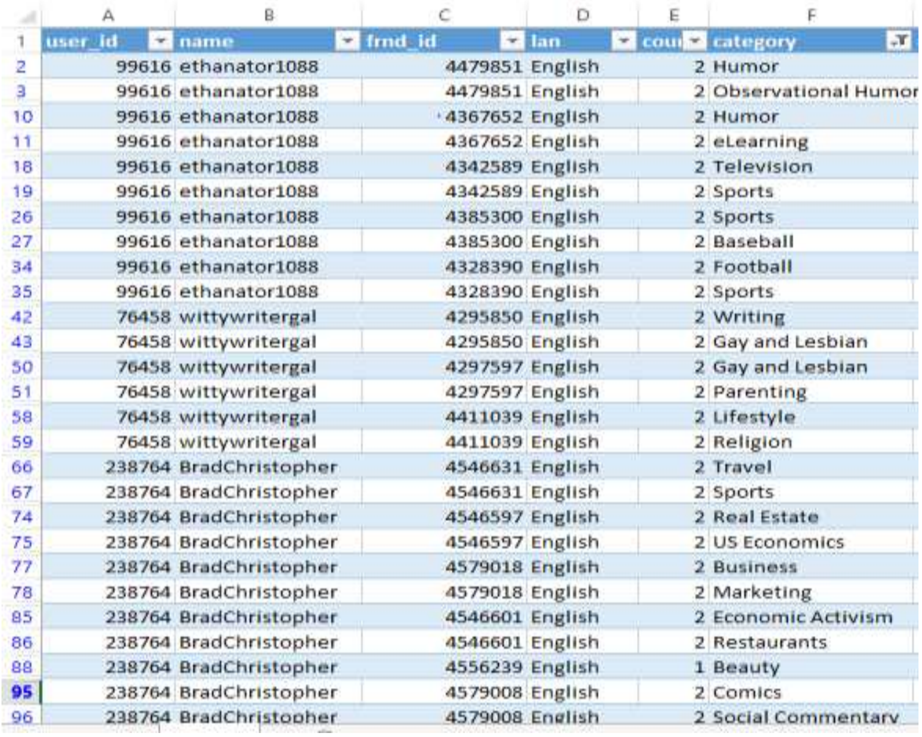
```
Output: Clustering of Social Network
Process:
Step1: Null Cluster
Step2: For all x[i] is part of N do
AS_F=False
For all cluster
IF ||x[i]- centroid(Cluster)||<threshold then
{
Update centroid (Cluster)
Ins_counter (Cluster)++
AS_F=true
}
Exit loop
End if
End for
If (not AS_f)
{
Centroid (new Cluster) =x[i]
Ins_counter (new Cluster=1)
Cluster= old Cluster union new Cluster
}
End if
Step 3.End for
```

We have 2904 data entry and 480 edges and 448 nodes. This .xml datasets are shown in fig 2.Converting this .xml format into readable format using Microsoft excel shown in fig 3.



```
<root>
  <user id="99616" name="ethanator1088">
    <weblog     id="4479851" lan="English">
      <categories count="2">
        <category>Humor</category>
        <category>Observational Humor</category>
      </categories>
      <tags count="6">
        <tag>funny</tag>
        <tag>humor</tag>
        <tag>owned</tag>
        <tag>pwned</tag>
        <tag>video</tag>
        <tag>videos</tag>
      </tags>
      <snippet>Lawn Mower Thief PWNED, Video  OK, PWNED Video is back,
as 3 girls sitting right behind where he gets a face full of beach san
    </weblog>
    <weblog id="4367652" lan="English">
      <categories count="2">
        <category>Humor</category>
        <category>eLearning</category>
      </categories>
      <tags count="6">
        <tag>funny</tag>
```

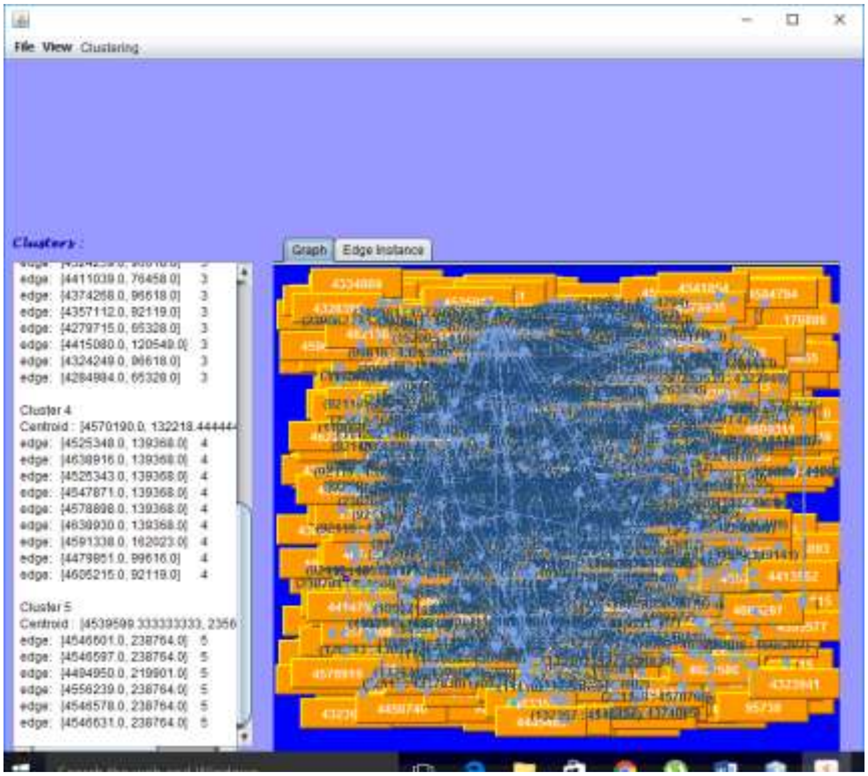
Fig 2. .XML Format Dataset



1	user_id	name	frnd_id	lan	cou	category
2	99616	ethanator1088	4479851	English	2	Humor
3	99616	ethanator1088	4479851	English	2	Observational Humor
10	99616	ethanator1088	4367652	English	2	Humor
11	99616	ethanator1088	4367652	English	2	eLearning
18	99616	ethanator1088	4342589	English	2	Television
19	99616	ethanator1088	4342589	English	2	Sports
26	99616	ethanator1088	4385300	English	2	Sports
27	99616	ethanator1088	4385300	English	2	Baseball
34	99616	ethanator1088	4328390	English	2	Football
35	99616	ethanator1088	4328390	English	2	Sports
42	76458	wittywritergal	4295850	English	2	Writing
43	76458	wittywritergal	4295850	English	2	Gay and Lesbian
50	76458	wittywritergal	4297597	English	2	Gay and Lesbian
51	76458	wittywritergal	4297597	English	2	Parenting
58	76458	wittywritergal	4411039	English	2	Lifestyle
59	76458	wittywritergal	4411039	English	2	Religion
66	238764	BradChristopher	4546631	English	2	Travel
67	238764	BradChristopher	4546631	English	2	Sports
74	238764	BradChristopher	4546597	English	2	Real Estate
75	238764	BradChristopher	4546597	English	2	US Economics
77	238764	BradChristopher	4579018	English	2	Business
78	238764	BradChristopher	4579018	English	2	Marketing
85	238764	BradChristopher	4546601	English	2	Economic Activism
86	238764	BradChristopher	4546601	English	2	Restaurants
88	238764	BradChristopher	4556239	English	1	Beauty
95	238764	BradChristopher	4579008	English	2	Comics
96	238764	BradChristopher	4579008	English	2	Social Commentary

Fig 3. Readable Dataset

In Fig 4 using this datasets create graph and give cluster size as input and generate cluster in this cluster size is 5.



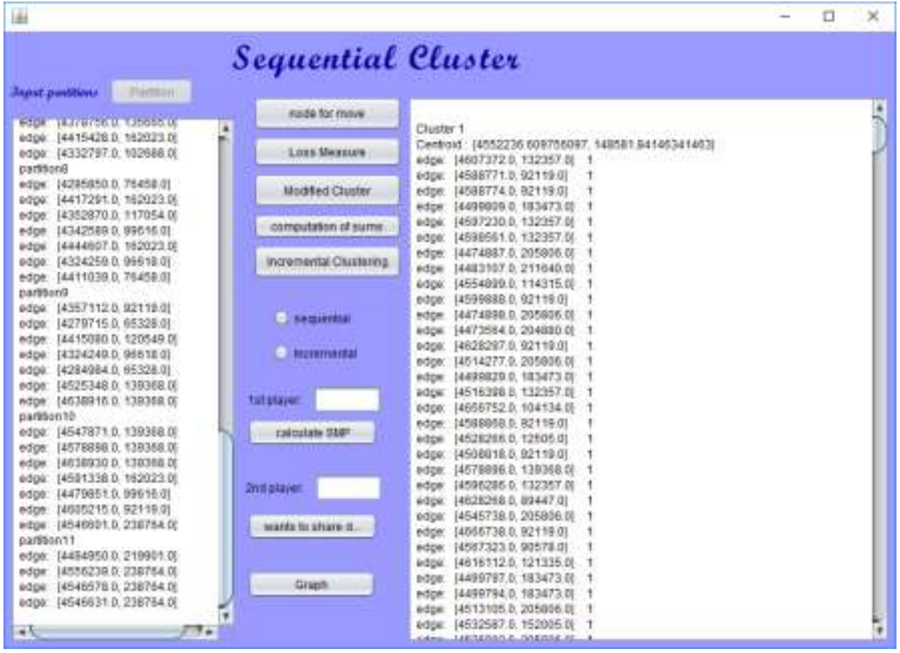
```
edge:  [4411039.0, 76458.0]     3
edge:  [4374268.0, 96618.0]     3
edge:  [4357112.0, 92119.0]     3
edge:  [4279715.0, 65328.0]     3
edge:  [4415080.0, 120549.0]    3
edge:  [4324249.0, 96618.0]     3
edge:  [4284984.0, 65328.0]     3

Cluster 4
Centroid : [4570190.0, 132218.44444
edge:  [4525348.0, 139368.0]    4
edge:  [4638916.0, 139368.0]    4
edge:  [4525343.0, 139368.0]    4
edge:  [4547871.0, 139368.0]    4
edge:  [4578898.0, 139368.0]    4
edge:  [4638930.0, 139368.0]    4
edge:  [4591338.0, 162023.0]    4
edge:  [4479851.0, 99616.0]     4
edge:  [4605215.0, 92119.0]     4

Cluster 5
Centroid :  [4539599.333333333, 2356
edge:  [4546601.0, 238764.0]    5
edge:  [4546597.0, 238764.0]    5
edge:  [4494950.0, 219901.0]    5
edge:  [4556239.0, 238764.0]    5
edge:  [4546578.0, 238764.0]    5
edge:  [4546631.0, 238764.0]    5
```

Fig 4. Cluster and Graph

In fig 5 right side part is sequential cluster partition are shown. On this partition we perform move node, find measure loss, modified cluster are perform. Right side part is incremental cluster in this cluster size are 5.

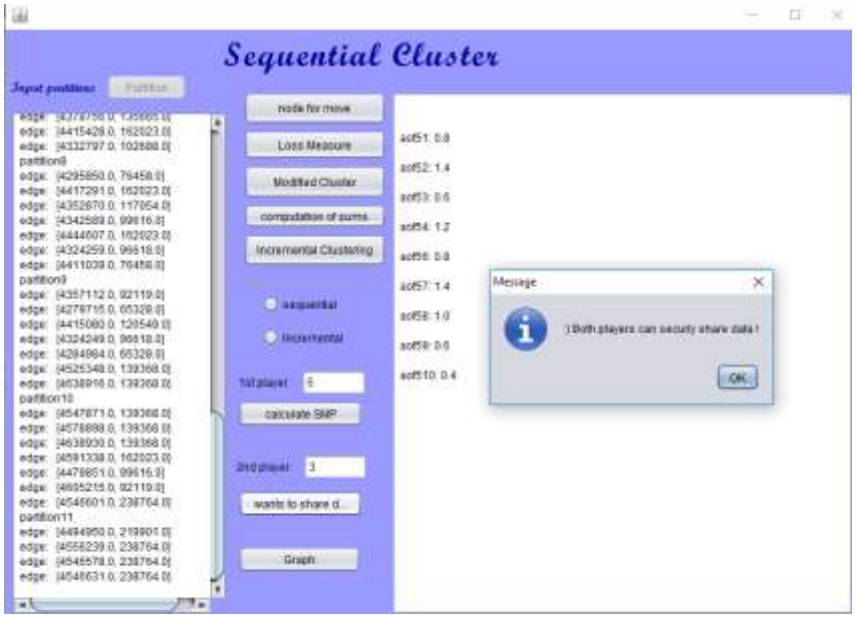


```
edge:  [4415428.0, 162023.0]
edge:  [4332797.0, 102688.0]
partition8
edge:  [4295850.0, 76458.0]
edge:  [4417291.0, 162023.0]
edge:  [4352870.0, 117054.0]
edge:  [4342589.0, 99616.0]
edge:  [4444607.0, 162023.0]
edge:  [4324259.0, 96618.0]
edge:  [4411039.0, 76458.0]
partition9
edge:  [4357112.0, 92119.0]
edge:  [4279715.0, 65328.0]
edge:  [4415080.0, 120549.0]
edge:  [4324249.0, 96618.0]
edge:  [4284984.0, 65328.0]
edge:  [4525348.0, 139368.0]
edge:  [4638916.0, 139368.0]
partition10
edge:  [4547871.0, 139368.0]
edge:  [4578898.0, 139368.0]
edge:  [4638930.0, 139368.0]
edge:  [4591338.0, 162023.0]
edge:  [4479851.0, 99616.0]
edge:  [4605215.0, 92119.0]
edge:  [4546601.0, 238764.0]
partition11
edge:  [4494950.0, 219901.0]
edge:  [4556239.0, 238764.0]
edge:  [4546578.0, 238764.0]
edge:  [4546631.0, 238764.0]
```

```
Cluster 1
Centroid :  [4552236.609756097, 148581.84146341463]
edge:  [4607372.0, 132357.0]    1
edge:  [4588771.0, 92119.0]     1
edge:  [4588774.0, 92119.0]     1
edge:  [4499809.0, 183473.0]    1
edge:  [4597230.0, 132357.0]    1
edge:  [4598561.0, 132357.0]    1
edge:  [4474887.0, 205806.0]    1
edge:  [4483107.0, 211640.0]    1
edge:  [4554899.0, 114315.0]    1
edge:  [4599888.0, 92119.0]     1
edge:  [4474898.0, 205806.0]    1
edge:  [4473584.0, 204880.0]    1
edge:  [4628297.0, 92119.0]     1
edge:  [4514277.0, 205806.0]    1
edge:  [4499829.0, 183473.0]    1
edge:  [4516398.0, 132357.0]    1
edge:  [4656752.0, 104134.0]    1
edge:  [4588868.0, 92119.0]     1
edge:  [4528266.0, 12505.0]     1
edge:  [4508018.0, 92119.0]     1
edge:  [4578896.0, 139368.0]    1
edge:  [4596286.0, 132357.0]    1
edge:  [4628268.0, 89447.0]     1
edge:  [4545738.0, 205806.0]    1
edge:  [4666738.0, 92119.0]     1
edge:  [4587323.0, 90578.0]     1
edge:  [4616112.0, 121335.0]    1
edge:  [4499797.0, 183473.0]    1
edge:  [4499794.0, 183473.0]    1
edge:  [4513105.0, 205806.0]    1
edge:  [4532587.0, 152005.0]    1
```

Fig 5.Incrmental Cluster

In fig 6 security part is shown one player generate the keys this right keys are enter by other player .If both keys match then both can communicate. If keys not match then both can't communicate shown in fig 7.



```
edge:  [4378756.0, 135665.0]
edge:  [4415428.0, 162023.0]
edge:  [4332797.0, 102888.0]
partition8
edge:  [4295850.0, 76458.0]
edge:  [4417291.0, 162023.0]
edge:  [4352870.0, 117054.0]
edge:  [4342589.0, 99616.0]
edge:  [4444607.0, 162023.0]
edge:  [4324259.0, 96618.0]
edge:  [4411039.0, 76458.0]
partition9
edge:  [4357112.0, 92119.0]
edge:  [4279715.0, 65328.0]
edge:  [4415080.0, 120549.0]
edge:  [4324249.0, 96618.0]
edge:  [4284984.0, 65328.0]
edge:  [4525348.0, 139368.0]
edge:  [4638916.0, 139368.0]
partition10
edge:  [4547871.0, 139368.0]
edge:  [4578898.0, 139368.0]
edge:  [4638930.0, 139368.0]
edge:  [4591338.0, 162023.0]
edge:  [4479851.0, 99616.0]
edge:  [4605215.0, 92119.0]
edge:  [4546601.0, 238764.0]
partition11
edge:  [4494950.0, 219901.0]
edge:  [4556239.0, 238764.0]
edge:  [4545578.0, 238764.0]
edge:  [4546631.0, 238764.0]
```

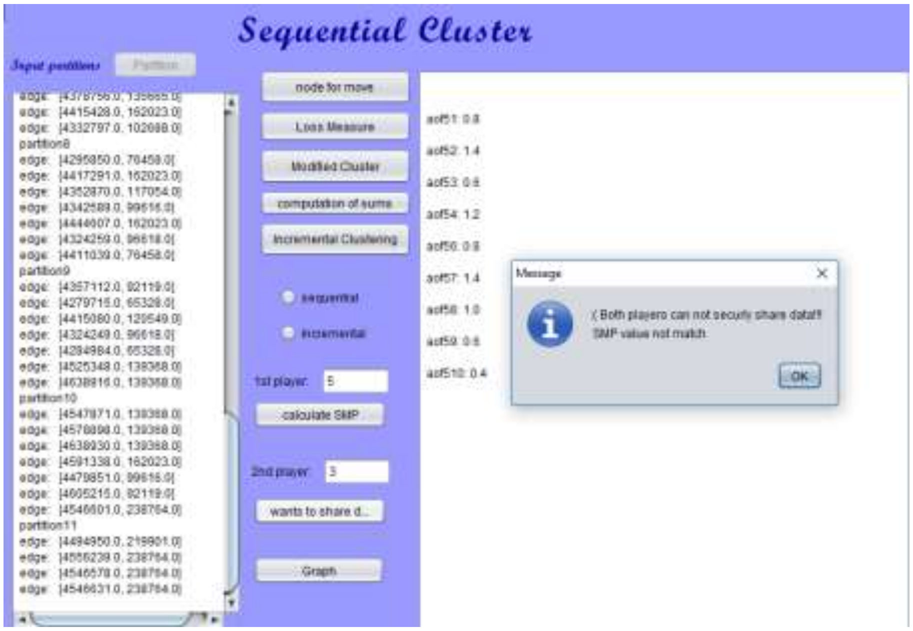
1st player: 5

2nd player: 3

```
aof51: 0.8
aof52: 1.4
aof53: 0.6
aof54: 1.2
aof56: 0.8
aof57: 1.4
aof58: 1.0
aof59: 0.6
aof510: 0.4
```

:) Both players can securly share data !

Fig 6.Both Share Data



```
edge:  [4378756.0, 135665.0]
edge:  [4415428.0, 162023.0]
edge:  [4332797.0, 102888.0]
partition8
edge:  [4295850.0, 76458.0]
edge:  [4417291.0, 162023.0]
edge:  [4352870.0, 117054.0]
edge:  [4342589.0, 99616.0]
edge:  [4444607.0, 162023.0]
edge:  [4324259.0, 96618.0]
edge:  [4411039.0, 76458.0]
partition9
edge:  [4357112.0, 92119.0]
edge:  [4279715.0, 65328.0]
edge:  [4415080.0, 120549.0]
edge:  [4324249.0, 96618.0]
edge:  [4284984.0, 65328.0]
edge:  [4525348.0, 139368.0]
edge:  [4638916.0, 139368.0]
partition10
edge:  [4547871.0, 139368.0]
edge:  [4578898.0, 139368.0]
edge:  [4638930.0, 139368.0]
edge:  [4591338.0, 162023.0]
edge:  [4479851.0, 99616.0]
edge:  [4605215.0, 92119.0]
edge:  [4546601.0, 238764.0]
partition11
edge:  [4494950.0, 219901.0]
edge:  [4556239.0, 238764.0]
edge:  [4545578.0, 238764.0]
edge:  [4546631.0, 238764.0]
```

1st player: 5

2nd player: 3

```
aof51: 0.8
aof52: 1.4
aof53: 0.6
aof54: 1.2
aof56: 0.8
aof57: 1.4
aof58: 1.0
aof59: 0.6
aof510: 0.4
```

:(Both players can not securly share data!!
SMP value not match.

Fig 7.Both Can't Share Data

In table 1 some different cluster size are shown. That time, sequential clustering and incremental clustering algorithm performance are shown in milliseconds. Using this table data make different chart of time complexity are shown column chart are shown in fig 8 and line chart are shown in fig 9.

Cluster Size	Sequential Time(ms)	Incremental Time(ms)
3	5235	2563
4	6949	2614
5	4066	2676
6	4428	2901
7	11932	2587

8	25133	3008
9	29834	2730
10	18437	3133

Table 1. Time Table

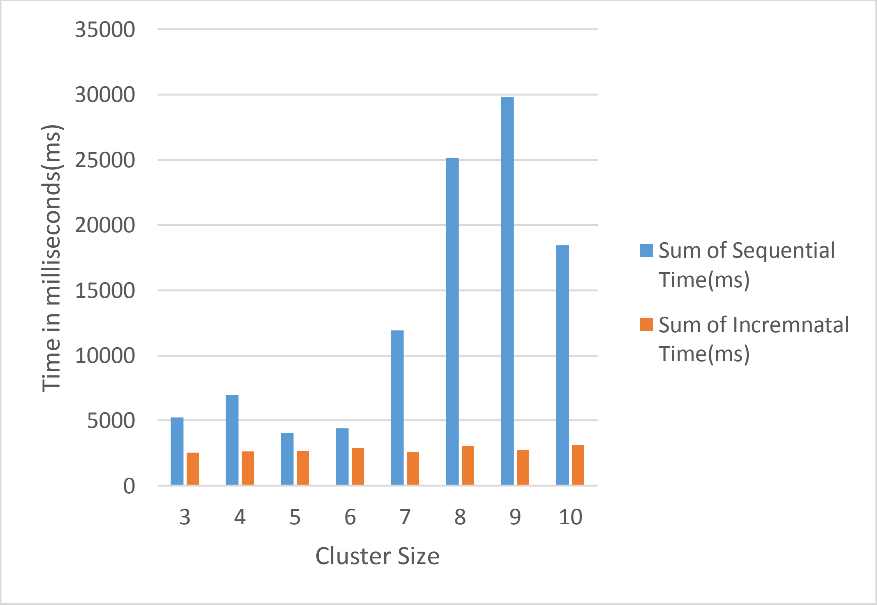


Fig 8. Comparision Column Chart

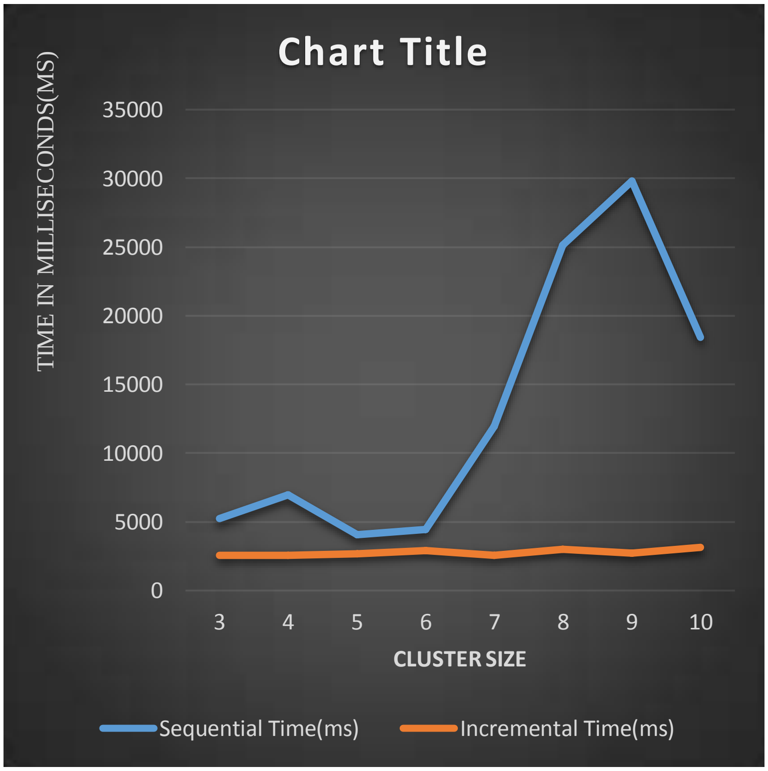


Fig 9. Comparison Line Chart

III.CONCLUSION AND FUTURE WORKS

Proposed system is running on real and sample data set which improves the performance and reliability of the system [15].We show incremental clustering algorithms for secure social networks and anonymization view. Narrative information loss and the users must hide from their allocation of their nodes which are to be clustered from the other player is the main problem in the existing system. So we use incremental clustering algorithm for security and reduce time complexity. As the privacy protection apply we can block the users/attacker for some time of span who are trying to hack the data.

ACKNOWLEDGMENT

We would like to extend our gratitude to our respected teacher – Mr.Romil Patel, for his constant support and for imparting us with the knowledge and helping us throughout the paper.

REFERENCES

1. Tamir Tassa and Dror J. Cohen “Anonymization of Centralized and Distributed Social Networks by Sequential Clustering” IEEE Transactions On Knowledge And Data Engineering, Vol. 25, No. 2, February 2013
2. G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu, “Anonymizing Tables,” Proc. 10th Int’l Conf Database Theory (ICDT), vol. 3363, pp. 246-258, 2005.
3. Baraba´si and R. Albert, “Emergence of Scaling in Random Networks,” Science, vol. 286, pp. 509-512, 1999.
4. J. Benaloh, “Secret Sharing Homomorphisms: Keeping Shares of a Secret Secret,” Proc. Advances in Cryptology (Crypto), pp. 251-260, 1986.
5. F. Bonchi, A. Gionis, and T. Tassa, “Identity Obfuscation in Graphs Through the Information Theoretic Lens,” Proc. IEEE 27th Int’l Conf. Data Eng. (ICDE), pp. 924-935, 2011.
6. Campan and T.M. Truta, “Data and Structural k-Anonymity in Social Networks,” Proc. Second ACM SIGKDD Int’l Workshop Privacy, Security, and Trust in KDD (PinKDD), pp. 33-54, 2008.
7. W. Jiang and C. Clifton, “A Secure Distributed Framework for Achieving k-Anonymity,” The Int’l J. Very Large Data Bases, vol. 15, pp. 316-333, 2006.
8. M. Kantarcioglu and C. Clifton, “Privacy-Preserving Distributed Mining of Association Rules on Horizontally Partitioned Data,” IEEE Trans. Knowledge and Data Eng., vol. 16, no. 9, pp. 1026-1037, Sept. 200410] S. Kirkpatrick, D.G. Jr, and M.P. Vecchi, “Optimization by Simmulated Annealing,” Science, vol. 220, no. 4598, pp. 671-680, 1983.
9. K. Liu and E. Terzi, “Towards Identity Anonymization on Graphs,” Proc. ACM SIGMOD Int’l Conf. Management of Data (SIGMOD), pp. 93-106, 2008.
10. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, “-Diversity: Privacy Beyond k-Anonymity,” ACM Trans. Knowledge Discovery and Data, vol. 1, no. 1, article 3, 2007.
11. M.E. Nergiz and C. Clifton, “Thoughts on k-Anonymization,” Proc. Int’l Conf. Data Eng. (ICDE), p. 96, 2006.
12. P.Mohana Lakshmi,P.Balaji,P.Nirupma “Sequential Clustering Algorithms for Anonymizing Social Networks”.
13. S.Bhagat,G.Coremode,B.Krishnamurthy and D. Srivasta “Class based graph annoymization for social network data”
14. Hongxin Hu, Member, IEEE, Gail-Joon Ahn, Senior Member, IEEE, and Jan Jorgensen” Multiparty Access Control for Online Social Networks: Model And Mechanisms”
15. Network Security On Centralized and Distributed Social Networks by Increment Clustering.Mr.MrunalkumarKhairnar[1],Ms.Dipali Hadiyel[2], Mr.Sunny Thakur[3]