FEATURE BASED COMMUNITY DETECTION BY EXTRACTING FACEBOOK PROFILE DETAILS

Rajeswari Sridhar, Akshaya Kumar, S. Bagawathi Roshini, Ramya Kumar Sundaresan and Suganthini Chinnasamy
Department of Computer Science and Engineering, Anna University, Chennai, India

Abstract

The rise of social networks had marked the revolution and transformation of human relationships and the information age. Social networks, Facebook in specific, have more than a billion daily active users which means petabytes of data are generated every second and there are so many social interactions occurring simultaneously. Community detection revolves around the study of these social interactions and common interests to derive the most efficient method of communication to specialized groups. Considering a preferred set of features such as the posts, likes, education background and the location of users for an optimal data structure, the selection of significant users for community analysis is implemented with the unique approach to investment score and dynamic threshold allocations for the graph creation. The community detection process focuses on the analysis of cliques and map-overlay. The emphasis on the detection of overlapping communities enhances the analysis of community relationships.

Keywords

Community Detection, Data Structure, Link Weights, Influence Metric, Cliques, Map Overlay

1. INTRODUCTION

Community detection revolves around the discovery of groups of interacting people based on common interests. There are numerous groups on Facebook which promote the ideas of like-minded people who share common interests, goals, jobs, profession, education backgrounds, location, etc. With a carefully calculated approach to communicate with a targeted audience, there will surely be more acceptances and understanding of ideas and opinions. Our method highlights the importance of similarity in thoughts and interests in determining social communities. Considering communities on Facebook involves a huge user base and key features such as likes, posts, location, and education backgrounds are essential for analysis. Community detection plays a key role in promoting great ideas within interest groups.

Real-time data from Facebook involves communities of various sizes and it is necessary to detect the overlapping nature of communities [1]. A graph-oriented solution requires the usage of link-weights which is crucial to determine communities of high importance [2].

The aim of this work is to successfully determine communities for user networks based on key elements of Facebook such as posts, likes, location, and education backgrounds. On Facebook, users share opinions, interests, messages and pages dedicated to topics varying from movies, TV shows, music and food to books, sports and corporations, promoting their latest product lines. We obtain data sets which contain crawled profile information from Facebook's dense network through the website interface, with acquired user permissions. The cleaned data is stored in an optimal data structure, for efficient graph creation and access.

This paper is organized as follows: Section 2 discusses on related work to community detection, section 3 describes the proposed system design in detail, section 4 discusses the results and evaluation of the experiments conducted on our system for identifying its accuracy, section 5 concludes the work with possible extensions.

2. RELATED WORK

The clique-based algorithm [1] incorporates pruning strategies, which is up to orders of magnitude faster on large, sparse graphs and of comparable runtime on denser graphs. The algorithm and the new heuristic are well suited for parallelization and are applicable for detecting overlapping communities in networks. Another work presents a novel approach to distinguish the internal links of the community and external links between connected communities based on link weights [2]. The method was used to convert an un-weighted network to a weighted network by assigning link weights and later was used to detect communities based on weak and strong links.

Hangal et al. [3] explores the hypothesis that social searches can be made more effective by taking into consideration the influence a person has over another and improve social search. The proposed analysis gives an insight to asymmetric relationship that could possibly exist between people. Arab et al. [4] states a bottom up approach for community detection which is initiated by finding the fine grained community in order to find the real community of a network. This approach can be used to merge multiple sub-community structures to identify the accurate communities in a network. A novel algorithm, based on the Max-Flow Min-Cut theorem proposed by Qi et al. [5] is validated through a variety of data sets ranging from synthetic graphs to real-world benchmark data sets, outputs an optimal set of local communities. The output of the Quasi Clique Merger which is a hierarchical clustering algorithm is used to initiate the new community detection algorithm. The existing algorithm by Girvan and Newman Algorithm forms the foundation of community detection.

Based on the lessons learnt from the studies, we propose to identify communities for a large social network based on the preferred set of features of each user. The process is done by constructing a strict undirected weighted graph for categories such as books, music, food, sports, TV shows, organizations and movies preferred by the users. The level of intersection of similar interests is determined by the Influential Metric computed as link weights across the edges. Further, a layered approach is used to perform maximum clique detection among the sub-graphs to detect communities for users. The cliques and connected component methods are essential to detect the overlapping communities present in the networks. Our approach differs from the common community detection methodology which depends mainly on the users' friendship networks [7]. With the focus on users' locations and education backgrounds shared on Facebook, community detection is implemented with the map-overlay algorithm [9] to handle overlapping communities. The doubly connected edge-list [10] representation of overlapping communities enhances the understanding of dominating nodes of communities.

3. SYSTEM DESIGN

Our proposed system which is feature based community detection on Facebook presents a novel approach in grouping like-minded users with the help of three phases. They are data extraction phase, graph creation phase and community detection phase. The block diagram is depicted in Fig.1.

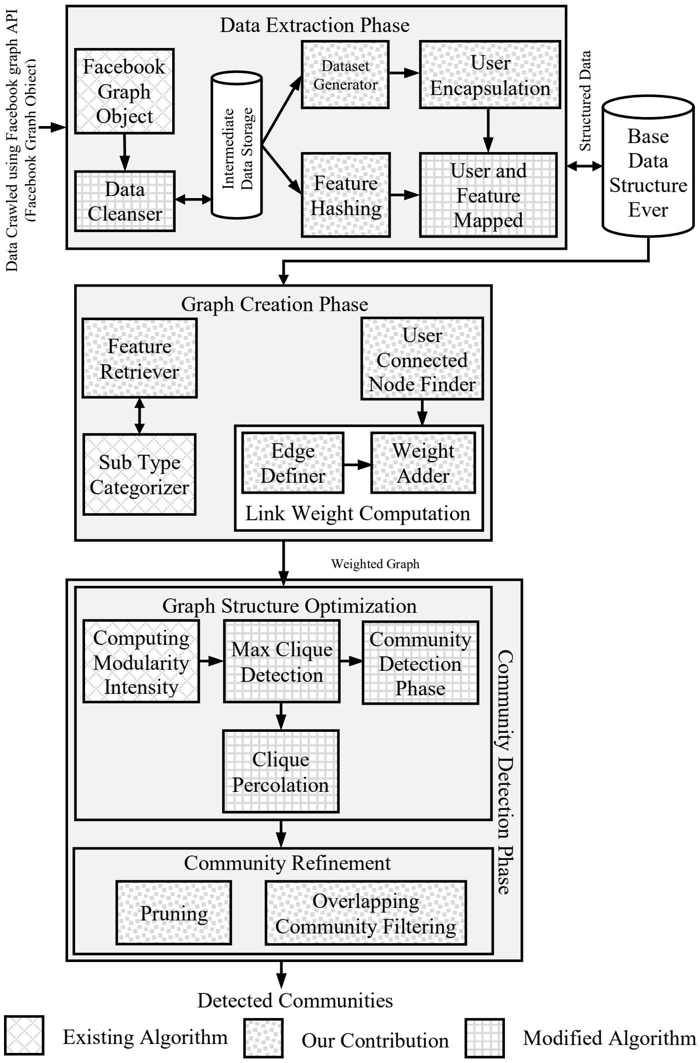
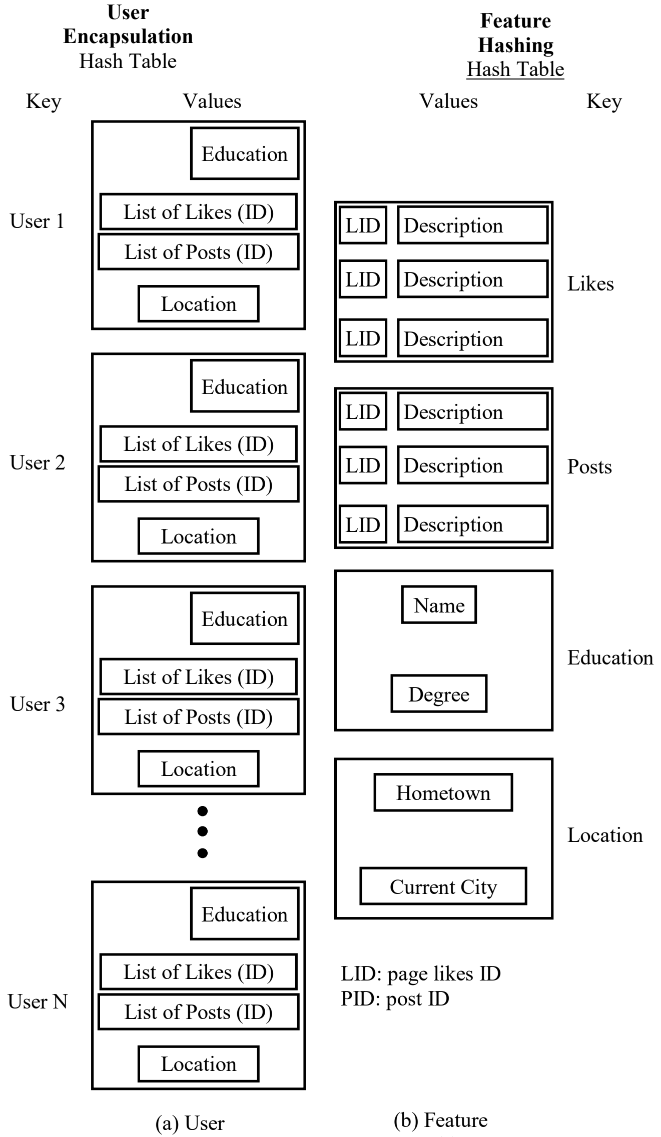


Fig.1. Block diagram for Feature based community detection on Facebook (The three phases comprises of multiple modules)

The crucial factor involved in handling huge Facebook data urges the need for quick access. Therefore, our work proposes the use of an optimal data structure with a fast retrieval mechanism. The category-based sub-graph creation concentrates on Facebook's unique features such as likes, posts, etc. For selecting a set of users who share an affinity towards the sub-types such as books, music, etc., our contribution of the Investment Score and dynamic threshold allocations proves to be accurate. The concept of map-overlay is commonly utilized for Geographical Information Systems and the adaptation of the same concept for the purpose of analyzing overlapping communities, gives excellent results for the community detection process. The details of the block diagram are explained in the following sub-sections.



(a) User Encapsulation

(b) Feature Hashing

Fig. 2. Data Structure representation

3.1 DATA EXTRACTION AND REPRESENTATION

Using the Facebook Graph API, a website was created and we obtained user permissions using a single click and collected features from each user like their page likes, posts, education history and current location. The data was obtained as Facebook graph objects which then had to be parsed into arrays. The data was cleaned from redundant values, stray characters, and null values, and put into an intermediate storage. As information from Facebook could be obtained only from two hundred people, a synthetic network had to be created. The dataset generator module generates a user id. For each user id, a randomized set of page-like ids, post ids, a single education institute and location was assigned. All this data was then aggregated and wrapped into a single unit and stored in the data structure. All unique Facebook features collected from the users initially as well as extra real world Facebook data was collected and hashed accordingly. It is each of these feature ids that have been assigned to synthetic users. The diagrammatic representation of the proposed data structures, user encapsulation and feature hashing is depicted in Fig.2(a) and Fig.2(b) respectively.

In the final module, we map each feature to its set of users who have preferred it. This allows us to not only retrieve features preferred by a user, but also all the users who have preferred the same features. The efficiency of this data structure has been evaluated and is explained in detail in the experiments section. The information organized in a structured manner in this phase is what forms the basis of the next phase, the Graph Creation Phase. The pseudo code for the phase, representing the extraction of Facebook data and the creation of the data structure is indicated in Algorithm 1.

Algorithm 1 Data Extraction Algorithm

```
procedure Data–Extraction
    Object ← User graph object from Facebook Graph API
    Array ← Parse($Object$)
    data ← Clean($Array$)
    Store($data$)
    $n$ ← No of users
    for $i$ ← 1 to $n$ do
        $U$ ← user Id
        $R$ ← Generate($U$)
        $C$ ← Encapsulate($U$)
    end for
    for $j$ ← 1 to $n$ do
        $F$ ← Unique Feature
        Hash($F$, $type$)
    end for
    mappedData ← Map($U$,$F$)
    DataStructureStore($mappedData$)
end procedure
```

3.2 GRAPH CREATION

In order to detect all possible communities, we have retrieved a set of features that are preferred commonly and created weighted sub-graphs based on each feature, consisting of likely users represented as vertices and affinity towards the feature as edges. The features are page likes, posts, education history and the location. The retrieved features such as page likes and posts will be classified in to sub-types such as books, music, movies, sports etc. by the application of Naive Bayes classification technique. We have trained the classifier with a considerable amount of data for accurate classification. We have analyzed the frequency of posts and page likes for every sub-type for a particular user. The computed frequency corresponds to the investment made by the user on a particular sub-type. The proposed investment score by our work is calculated for all the users and we eliminated certain users based on the threshold values decided dynamically for each sub-type as per the size of the dataset. The accuracy of threshold values is tested and it is explained in the experiments section. For the purpose of clear explanation refer to pseudo code presented for edge definer module in Algorithm 2.

3.2.1 Influence Score:

We propose the influence score based on the number of edges between nodes. The set of users whose values of investment score was beyond the threshold value were represented as nodes and the weighted edges were added to connect them based on the influence score calculated as given by the Eq.(1), where *a* indicates the particular sub-type such as books, company, music etc. and *b* indicates the user. The number of sub-types is represented by *n* which takes the value 7 in our work.

$$Influence\left(a,b\right)=\frac{Invests(b,a)}{\sum_{i=1} Invets(b,i)} \tag{1}$$

The influence of a particular sub-type on a user is compared with the sum of influences of other sub-types has on the same user was used to determine the link weights. The weights were tuned with the help of cosine similarity between the users. Each of the sub-type was represented as a community sub-graph that could be further analyzed to detect fine grained communities in the following phase.

Algorithm 2: Edge Definer Algorithm

```
procedure EdgeDefiner($U$,$F$)
    $PL$ ← count of similar pages likes under subtype
    $P$ ← number of posts shared for the subtype
    $n$ ← number of users
    $S$ ← number of subtypes
    for $i$ ← 1 to $n$ do
        investmentScore← calcInvestmentScore($PL$,$P$,$Edu$,$Loc$)
    end for
    for $i$ ← 1 to $n$ do
        for $j$ ← 1 to $n$ do
            if investmentScore ≥ threshold then
                $e_{ij}$ ← DefineEdge()
                $g_i$ ← constructGraph($e_i$)
            end if
        end for
    end for
    return {$g_1$,$g_2$,$g_n$}
end procedure
```

3.3 COMMUNITY DETECTION

The first step for detecting the communities, the Modularity Intensity [2] Computation was performed to determine the cohesiveness between the users, thereby processing the weighted graph input to determine the individual and then, the total modularity intensity. The identification of the optimal cliques is done on sub-type oriented sub-communities such as music, movies, books, food, TV shows, company and sports. In order to verify the correctness of the cliques, we have computed the connected components. The detection of overlapping communities in the Clique Percolation step [1] was executed with the identification of adjacent k-clique communities.

Pruning strategies were suggested to produce sub-graphs of reduced and optimized size as this step can prevent the re-computations of the maximum clique by considering key features of the education and location of the users. We have used the map overlay approach [9] which is typically used for geographic information systems, to filter the overlapping community sub-structures with the pseudo code explained in Algorithm 3.

Algorithm 3: Overlapping Community Filtering

```
procedure OverlappingCommunityFiltering($G$)
$G \leftarrow$ Given Weighted Graph $G$ ($V$,$E$)
$V \leftarrow$ Set of vertices
$E \leftarrow$ Set of edges
$FA \leftarrow$ Mapping of Location Ids
$CO \leftarrow$ HashingLocations($V$, $LocInfo$)
$LS \leftarrow$ FindIntersections($CO$)
for $i \leftarrow 1$ to $n$ do
    $EL \leftarrow$ FindCommunities($LS[i]$)
end for
$FA \leftarrow$ VisualRepresentation($EL$)
return $finalG$
end procedure
```

4. EXPERIMENTS

In this section, we will go into detail about the evaluation of the system and its results. The types of community are Predefined; Books, Company, Food, Movies, Music, Sports, TV Shows. For a detailed analysis of our system, we split our dataset of 25000 nodes into 25 batches of 1000 nodes each. The split up is such that batch one has user's ids from 1 to 1000; batch two has ids from 1001 to 2000 and so on. This dataset has been used for most of the evaluation while certain evaluations required 5 batches of 5000 nodes.

Our data structure had been designed such that the search operation for each node is very quick. As each user id has been hashed, we can obtain the details in $O(1)$ efficiency. Retrieving each feature was also done in $O(1)$ access time since a set of feature ids is encapsulated for each user and obtaining the feature details could be done by accessing the feature table. Insertion time for creating the entire dataset was what proved to be a hassle initially. It proved to be $O(logn)$ where n is the number of users. However, using the concept of bulk loading, the process was sped.

We evaluated the system on three properties. The first, considering each node in a community individually. Reference communities had to be constructed. It was created with the help of our training dataset where each feature was tagged with a predetermined category. This data then used for creating reference communities, while the original dataset was used to create estimated communities. The corresponding batches were then evaluated. The metrics used for this property are Rand Index Fraction of Correctly Classified Nodes, and Normalized Mutual Information [8]. The second property is based on the topological nature of each community.

Reference communities are not required for the following metrics; Scaled Density, Community Size, and Hub Dominance [8]. The final property is based on the links. We evaluate the most optimal threshold values used in the graph creation phase.

Rand Index, also termed *RI*, corresponds to the proportion of node pairs for which both the estimated and reference community structures agree. *RI* lies between 0 to 1, with 0 indicating that the algorithm failed to estimate the community structure and 1.0 indicating that the algorithm correctly estimated the community structure. The node pairs for each community in the reference and estimated communities were counted. Values for all communities in each batch were totalled and then divided by the number of nodes present in each batch. The calculated RI tabulated in Table.1, was found to be ranged from 0.9 to 1.0 as denoted in Fig.3.

Table.1. Rand Index

Dataset Batch	A	B	Rand Index
1	373	387	0.927648
2	238	259	0.969111
3	294	321	0.928348
4	334	389	0.958868
5	411	435	0.988505
6	438	470	0.919148
7	272	296	0.898648
8	400	421	0.928741
9	225	244	0.897540
10	203	247	0.919028
11	409	438	0.908675
12	369	407	0.968058
13	228	266	0.917293
14	415	437	0.919908
15	242	274	0.967153
16	423	469	0.989339
17	202	223	0.919282
18	422	459	0.989106
19	349	376	0.928191
20	178	206	0.907766
21	247	289	0.989619
22	174	192	0.90625
23	367	395	0.939240
24	355	387	0.979328
25	382	384	0.947916

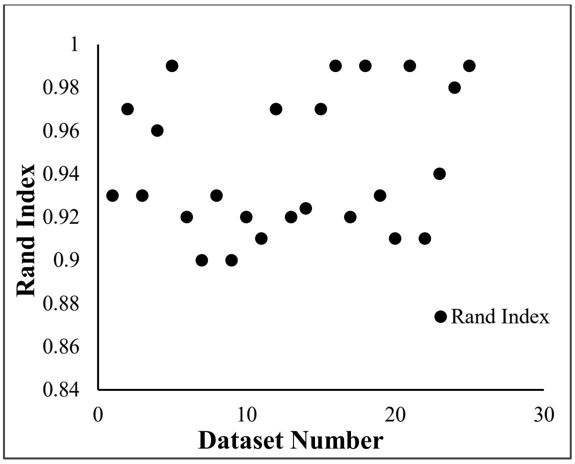


Fig.3. Results from Rand Index Values should be 0 and 1

Fraction of correctly classified nodes [8] is the value of number of nodes correctly classified against the total number of nodes existing. A node is correctly classified if its estimated community is the same than for the majority of nodes present in its reference community. Moreover, if an estimated community corresponds to a fusion of several reference communities, all the concerned nodes are considered as misclassified. The value of Fraction of Correctly classified Nodes, FCC measure, will lie between 0 and 1, with 1 being the most correctly classified. We have used the FCC measure to test our system against the reference communities under each of the category which includes as books, company, food, movie, music, sports and TV shows. The number of nodes classified by our system is recorded.

The corresponding results are given in Table.2. From the results shown in Fig.4, we observe that the category music, movies and food has lesser value than the others. The reason being, the misclassification by the naive Bayes classifier that failed to differentiate music albums from movie names and lesser training data available for food. Since categories books, company, sports, and TV shows have a large training data, they are indicating higher values. The fraction of correctly classified nodes measure gave us commendable results indicating our system has near optimal performance.

Table.2. FCC values comparison for various Community Category

Community Category	Reference	Our System	FCC Measure
Books	14826	14602	0.98
Company	15483	15326	0.99
Food	14636	14003	0.95
Movies	12806	12025	0.93
Music	12850	12206	0.94
Sports	9215	9169	0.98
TV shows	14871	14620	0.99

To further enhance the evaluation of our proposed system, we compared our results with four other existing community detection methods using the same dataset consisting of twenty-five thousand nodes. The chosen evaluation metrics for this purpose are rand index and Fraction of correctly classified nodes [8]. Out of the five methods including our own proposed method, we found that Louvain top the Rand index measure with a whopping 98% followed closely by InfoMod, the proposed system, fast greedy and COPRA at 97%, 94%, 91% and 80% respectively. With the fraction of correctly classified nodes measure however, we found that our system topped the metric measure with a 97%, with others following behind at COPRA 90%, fast greedy 80%, Louvain 42.5% and InfoMod behind by a landslide 25%. From these results, we find that our system has consistently been performing well. The results of this evaluation are tabulated in Table.3 and represented graphically in Fig.5.

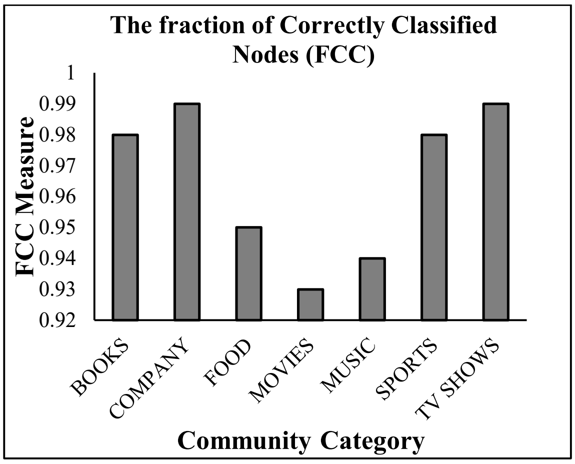


Fig.4. Fraction of Correctly Classified Nodes Computations

Table.3. RI and FCC values comparison for various Community Detection Methods

Detection Method	Rand Index	FCC Measure
COPRA	0.8	0.9
Fast Greedy	0.91	0.8
InfoMod	0.97	0.425
Louvain	0.98	0.25
Our system	0.94	0.97

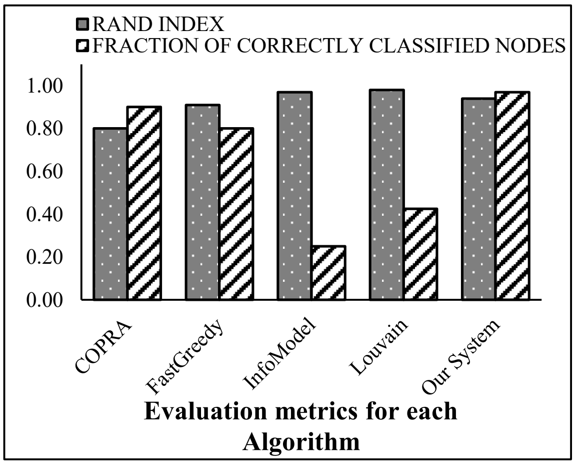


Fig.5. Comparison of Community Detection Methods

Normalized Mutual Information [8] is a measure to compare two different partitions of one data set, by measuring how much information they have in common. If the estimated communities correspond perfectly to the reference ones, the measure takes value 1, whereas it is 0 when they are independent. We have divided our users into five different groups and randomly categorize them under the seven communities. By using this as a reference we have tested our system to check if the nodes belong to the same set or not. We have given a value of one if they belong to the same set up to a certain extent else we have given a value of zero. The results are given in able 4 and depicted in Fig.6.

After further analysis we decided when the users of the reference set matches up to 80 with the test set, the system awards NMI score of 1 else it takes the value of 0. The reason why there were a few misses where our system got a zero value is because of eliminating certain users by the threshold value that gave importance only to users having large value for investment score. But the number of ones in the result is greater than the zeros indicating a fair performance in detecting the correct users for the appropriate category.

Table.4. Normalized Mutual Information

Data Subsets	REF 1	REF 2	REF 3	REF 4	REF 5
SET 1	1	1	1	0	0
SET 2	1	1	0	0	1
SET 3	0	1	0	1	1
SET 4	0	1	0	1	1
SET 5	1	0	1	1	0

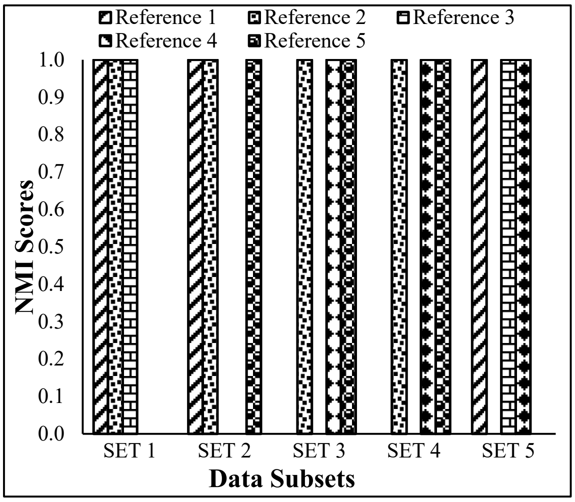


Fig.6. Normalized Mutual Information Analysis

Community sizes [8] come in various numbers and it is essential to know the number of people in a community to be able to assign them a purpose. For educational purposes, the ideal community size for a project would be three, four, and five. On evaluating the system, we have found that in each batch, the number of communities of size three was much higher, with size four next in line, and size five trailing close behind. Size one and two were found to be many in number and hence eliminated, keeping the threshold value for communities at 3. In Fig.7, each colour denotes a batch with the number of communities appended at the end. We can clearly see that communities of size three are large in number compared to size four and size five in Table.5.

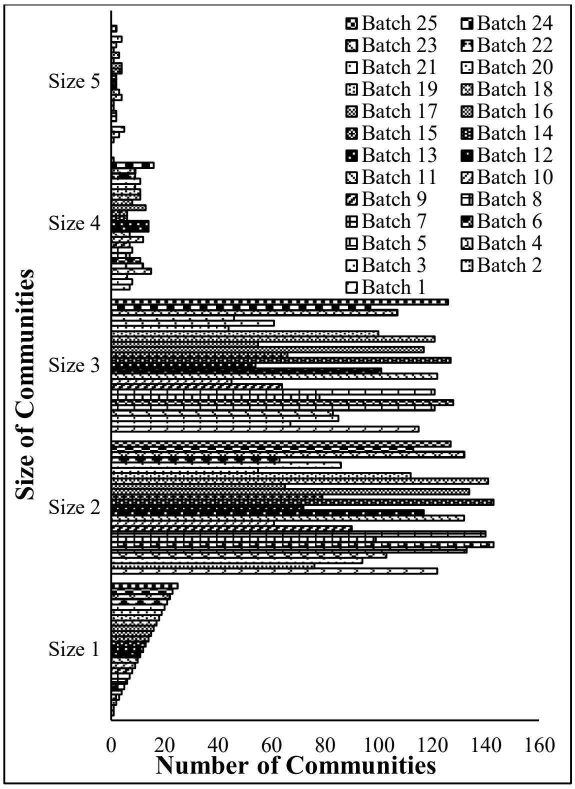


Fig.7. Results from Community Size

Table.5. Community Size

Data Subsets	Total	Size 3	Size 4
1	122	115	7
2	76	67	8
3	94	85	6
4	103	83	15
5	133	121	12
6	143	128	11
7	99	78	7
8	140	121	8
9	90	64	7
10	61	45	12
11	132	122	7
12	117	101	14
13	72	54	14
14	143	127	6
15	79	66	6
16	134	117	13
17	65	55	8
18	141	121	11
19	112	100	11
20	55	44	9
21	86	61	11
22	63	46	9
23	132	107	9
24	113	97	16
25	127	126	1

Scaled density [8] for a community is defined as the ratio of links it actually contains to the number of links it could contain if all its nodes were connected. While most of the communities formed were cliques, there were some nodes that were connected to more nodes than others in its community. For each community in a batch, the existing links were counted, along with the number of edges in community if it were complete. The numbers were totalled for each batch. In Fig.8, red lines denote the number of existing links in a community and yellow as the number of possible links. We can see that the lines are almost of the same length. As most of the batches have their scaled density lie between the ranges 0.8 to 0.9, we can confidently say many of the communities are cliques, thus showing that our system has detected communities roughly 80% accurately.

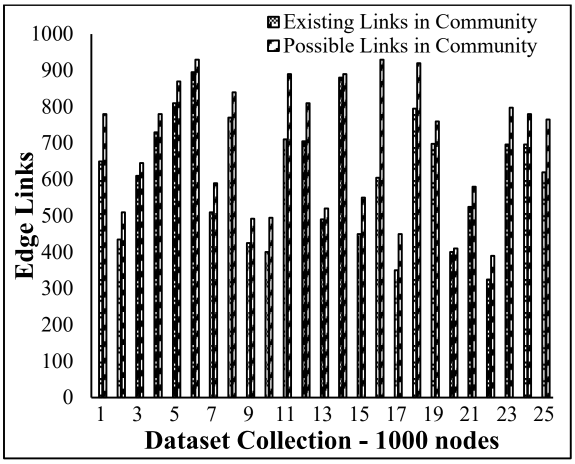


Fig.8. Results from Scaled density

Hub Dominance [8] determines the interconnected nature within communities, that is, identifying if there are some nodes connected to more nodes in its community than the others. For each estimated community, the hub dominance for each community is calculated. If the community is a clique, the hub dominance will have a value equivalent to the *n*-1 where n is the number of nodes in that community. For our system, we checked if the number of hub dominant nodes was equivalent to the number of nodes present in that community. If it was equal, then the community is not dominated. If not equal, the community is dominated by the hub dominant nodes. From our observations we find that many communities have not been dominated. This is another validation that most of the communities are cliques and accurately predicted. The observations are graphically represented in Fig.9.

The final eval.metric we used for our system is determining the threshold accuracy. In order to test the extent to which the particular set of users has affinity towards a category, we adjusted the threshold value for investment and checked the accuracy in detecting communities. The dataset containing 25,000 users implied a fixed community count of 5000, including overlapping structures. The results as tabulated in Table.6, show that the threshold value that takes sum of average and half of the same gives best results. Also the threshold value, sum of maximum and 20% of the same, as represented in Fig.10 gives good results. The computation of cliques is optimal in the above cases.

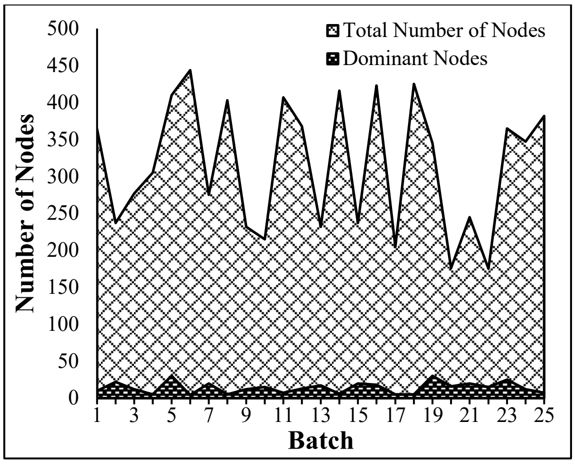


Fig.9. Results from Hub Dominance

Table.6. Threshold Accuracy

Threshold limit	All Communities	Overlapping
Average × 1.5	5276	1564
Max × 1.2	4701	167
Average × 1.3	6336	3751
Mode	2831	150

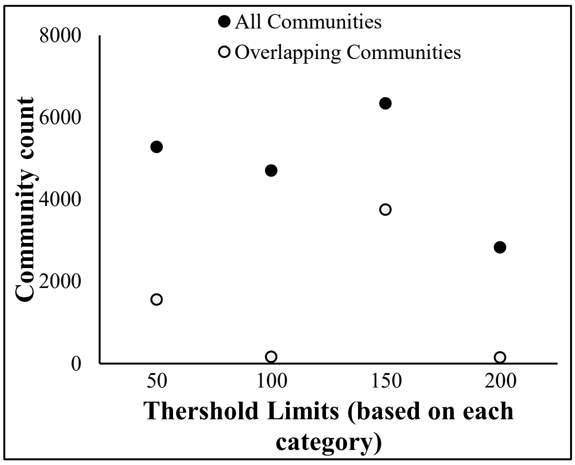


Fig.10. Threshold Accuracy Computations

The drawbacks observed in the experimental analysis can be attributed to the limited access to Facebook data, ineffective classification into pre-defined community categories, variations in threshold value allocations, long execution times of clique detection and map-overlay implementation.

5. CONCLUSION AND FUTURE WORK

Our work delves on community detection for a set of users using the feature data provided by Facebook. For this system, we have used page likes, posts, location, and education history, obtaining these details from Facebook. As only limited of number of users' data could be obtained using the Facebook Graph API, a synthetic network of twenty-five thousand users was created from those existing features, each user with a unique identity number and a set of features and each feature identified by its own unique number as well. The role of organizing the large dataset is crucial to improve the execution of feature retrieval and that of our system. The data structure used by us helped in efficient retrieval of data. The graph creation with the threshold values in each category on the influence metrics eased the detection of cliques and further, the overlapping communities. The overlapping nature of communities on social networks is exhibited with the map-overlay algorithm and is crucial to analysis of user relationships. On combining the results of all the evaluation metrics, we find that most of the communities detected are cliques, which is the sole purpose of our community detection. The users in each community are hence connected not only through their shared interests, but also by either their common location or education, assuming only undergraduate education institutes. Although the information for each user has been syndicated using existing real world data from Facebook, this system will work for feature data of users obtained directly from Facebook as well.

The future work and potential of community detection leads to applications revolving around the recommendations of user-specific groups depending on results from community detection analyses. Facebook imposes restrictions on the amount and type of user data that can be retrieved. With user permissions, more features could be used to enhance the detection and utilize the available Facebook information. For example, visited places could be an additional feature, or even events that people have attended. The overlapping of communities paves way for future development. All three map overlay conditions, namely, node-node overlap, edge-edge overlap, and edge-node overlap, could be used to solve the overlapping community's problem. Instead of pair-wise merging, another method could be used to find the complete overlapping of communities, giving rise to common groups for all seven predetermined communities, to even a single one. The system achieves high accuracy in detecting communities, but with the future work incorporated, the results could be more accurate, detailed, and used for many more purposes, for example, targeting a particular community for various reasons.

REFERENCES

[1] B. Pattabiraman et al., "Fast Algorithms for the Maximum Clique Problem on Massive Graphs with Applications to Overlapping Community Detection", *Journal of Internet Mathematics*, Vol. 37, No. 1, pp. 156-169 2014.

[2] Peng Gang Sun, "Weighting Links based on Edge Centrality for Community Detection", *Physica A: Statistical Mechanics and its Applications*, Vol. 394, pp. 346-357, 2014.

[3] Sudheendra Hangal, Diana MacLean, Monica S. Lam and Jeffrey Heer: "All Friends are Not Equal: using Weights in Social Graphs to Improve Search", *Proceedings of International Conference on Social Network Analysis Knowledge Discovery and Data Mining*, pp. 356-371, 2010.

[4] Mohsen Arab and Mohsen Afsharchi, "Community Detection in Social Networks using Hybrid Merging of Sub Communities", *Journal of Network and Computer Applications*, Vol. 40, pp. 73-84, 2014.

[5] Xingqin Qi, Wenliang Tang, Yezhou Wu, Guodong Guo, Eddie Fuller and Cun-Quan Zhang, "Optimal Local Community Detection in Social Networks Based on Density Drop of Subgraphs", *Pattern Recognition Letters*, Vol. 36, pp. 46-53, 2014.

[6] Joseph E. Gonzalez, Reynold S. Xin, Ankur Dave, Daniel Crankshaw, Michael J. Franklin and Ion Stoica, "Graphx: Graph Processing in a Distributed Dataflow Framework", *Proceedings of 11th Usenix Symposium on Operating Systems Design and Implementation*, pp. 599-613. 2014.

[7] W. Fan and A. Yeung, "Similarity between Community Structures of Different Online Social Networks and Its Impact on Underlying Community Detection", *Journal of Communications in Nonlinear Science and Numerical Simulation*, Vol. 20, No. 3, pp. 1015-1025, 2015.

[8] Gnce Orman, Vincent Labatut and Hocine Cherifi, "Comparative Evaluation of Community Detection Algorithms: A Topological Approach", *Journal of Statistical Mechanics: Theory and Experiment*, Vol. 2, pp. 802-809, 2012.

[9] Hans-Peter Kriegel, Thomas Brinkhoff and Ralf Schneider, "An Efficient Map Overlay Algorithm based on Spatial Access Methods and Computational Geometry", *Proceedings of International Workshop on Database Management Systems for Geographical Applications*, pp. 194-211, 1991.

[10] Mark De Berg, Marc Van Kreveld, Otfried Cheong and Mark Overmars, "*Computational Geometry: Algorithms and Applications*", 3rd Edition, Springer, 2008.