IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 17, Issue 2, Ver. VI (Mar – Apr. 2015), PP 61-67
www.iosrjournals.org

Socio Media Connect: A Social Profile based P2P Network

Chirag Jain[1],Hardeepsingh Jethra[2], Rishikesh Pange[3], Vinil Jain[4], S.C.Shrawne[5]
[1](VeermataJijabai Technological Institute, India)
[2](VeermataJijabai Technological Institute, India)
[3](VeermataJijabai Technological Institute, India)
[4](VeermataJijabai Technological Institute, India)
[5](VeermataJijabai Technological Institute, India)

Abstract: *It is observed that users tend to like and follow a particular type music/movies based on their social interests. Users like music of particular artists, album; movies/TV shows of a particular actor, director, genre, etc. Generally, peer-to-peer networks do not use this vital information to perform effective search using the metadata of media files. This information can be used to create social profile of users and cluster the peers based on their interests as reflected in the social profiles. This clustering of users over a peer-to-peer network will help in finding the users their interested files more easily. Also, peer-to-peer networks usually flood the entire network to search a particular file. But, due to clustering of peers based on their social interest, there is no need to flood the entire network to find a particular file, as most of the times the required file will be found in the cluster itself, which will in turn reduce the network overhead. We simulated this system using JXTA to connect the peers after clustering over P2P network. The peers with similar social profiles were connected via a common communication channel and therefore the results were better than the normal P2P network.*
Keywords: *clustering,media, peer-to-peer, sharing, similarity*

I. Introduction

Peer-to-peer networks have become very important in today's world and hence, it is really important to improve the efficiency of the P2P network for finding and receiving files. If a peer wants to search for a file, it requests all its neighbours for the file. If none of the neighbours have the requested file, then all the neighbours ask their neighbours. This process continues till the file is found in the network or all the peers in the network are requested for the file. This results in flooding of the entire P2P network. Because the peers with similar interests are not directly connected in a normal P2P network, they have a broader search window. In SocioMediaConnect, we aim to tackle this problem. The media files such as music, movies and TV shows contain important information such as genre, actors, artists, etc.

Example: Inception (Movie)

Title: Inception- Genre: Action, Mystery, Sci-Fi; Director: Christopher Nolan; Writer: Christopher Nolan; Actors: Leonardo DiCaprio, Joseph Gordon-Levitt, Tom Hardy; Language: English, Japanese, French

This information can be compiled into a file, what we call the social profile of the user. The social profile is built from all the files that the user shares with the peer-to-peer network.

II. Outline of Socio Media Connect

Our proposed work, SocioMediaConnect will have a peer-to-peer network with social profiling of each user, where users will be able to search objects by the semantic attributes of the multimedia files. Peers with similar interests are clustered to form a group. If the similarity quotient of the group and the new peer falls within the threshold then the peer is added to the group with highest similarity quotient. Since, the interested people will be directly connected by the network, the time required to find and retrieve resources will be much lesser. SocioMediaConnect will search for a given resource first amongst the people (neighbours) with matching similar profile and then if the resource is not found our system will flood the entire peer-to-peer network with search query, thus making the search window narrower. Moreover, most of the times, no separate channel is required for transferring file between peers because neighbouring peers are directly connected by common channel. This gives SocioMediaConnect a major advantage over common peer-to-peer networks. In SocioMediaConnect, peers mimic how people form social network and how they query, based on their interests. Whenever a new resource is downloaded from neighbouring peers, its metadata will be generated and integrated with previous metadata. The pipes used for communication between peers use secure cannel for communication, which uses public key encryption, to prevent loss of data or its modification and also prevents against other possible security thefts.

SocioMediaConnect will feature two kinds of peers:

- Member peer
- Admin peer

III. Proposed Work

The system consists of many groups with each group having members with similar interests in music, movies and series. Each member of SMC has its own social profile based on the music, movies and series interests of the user. Since the peers with interested file are directly connected with a common channel, the peer won't have to flood the entire P2P network with the search query. As a result the TTL (Time to Live) value of the packets will be reduced, thereby reducing the overhead of the entire P2P network.

The admin of each group behaves as a dual peer viz. admin and member. Admin peer, in addition to activities performed by the member peer, also performs activities such as maintaining the group vector, member list of all peers in the group, list of different groups present in SMC, sending group vector to new peer added to SMC, periodically maintains the status of all peers within the group. All peers have a unique username in order to avoid conflict. Also group names are unique.

Algorithm 1-To update the group vector after a new peer is added to the group:

Read the social profile of the new peer.
Create a List_Peer<Name, Value> of all the tags in the profile
Create a List_Group<Name, Value> of all the tags in the group vector
For each entry in the List_Peer
If Name exists in List_Group
Add entry.value to corresponding List_Group entry
Else create new entry in List_Group for the current tag and update the attributes
Write List_Group<Name,Value> to group vector

Group Vector:

Each group has its own group vector which is maintained by the admin. Initially, when the only member of the group is the admin, the vector of the admin is itself the group vector. As peers with same interests are added to the group, the group vector keeps on updating. Group Vector is stored in xml format. As a result, the group vector is the essential component in forming the groups of matching profiles. Whenever a peer downloads a new file from SMC, the peer generates new tags corresponding to the new file to temp.xml and periodically sends it to the admin. Admin updates the group vector whenever it receives such a file. Admins runs the same algorithm whenever it receives a new profile or temp.xml

Attribute file:

This file will be generated by the system when the peer joins a group. This file helps the peer to decide which group to j

oin the next time it comes online. It stores the following information: Type of peer, Group Name, Peer ID of Admin

```
<profile>
    <music>
        <tag>
            <Name>Tag_name</Name>
            <Value>No_of_objects by that Tag</Value>
        </tag>
    </music>
    <movies>
        <tag>
            <Name>Tag_name</Name>
            <Value>No_of_objects by that Tag</Value>
        </tag>
    </movies>
    <series>
        <tag>
            <Name>Tag_name</Name>
            <Value>No_of_objects by that Tag</Value>
        </tag>
    </series>
</ profile >
```

Figure 1: social profile format

Working of SMC with an example:

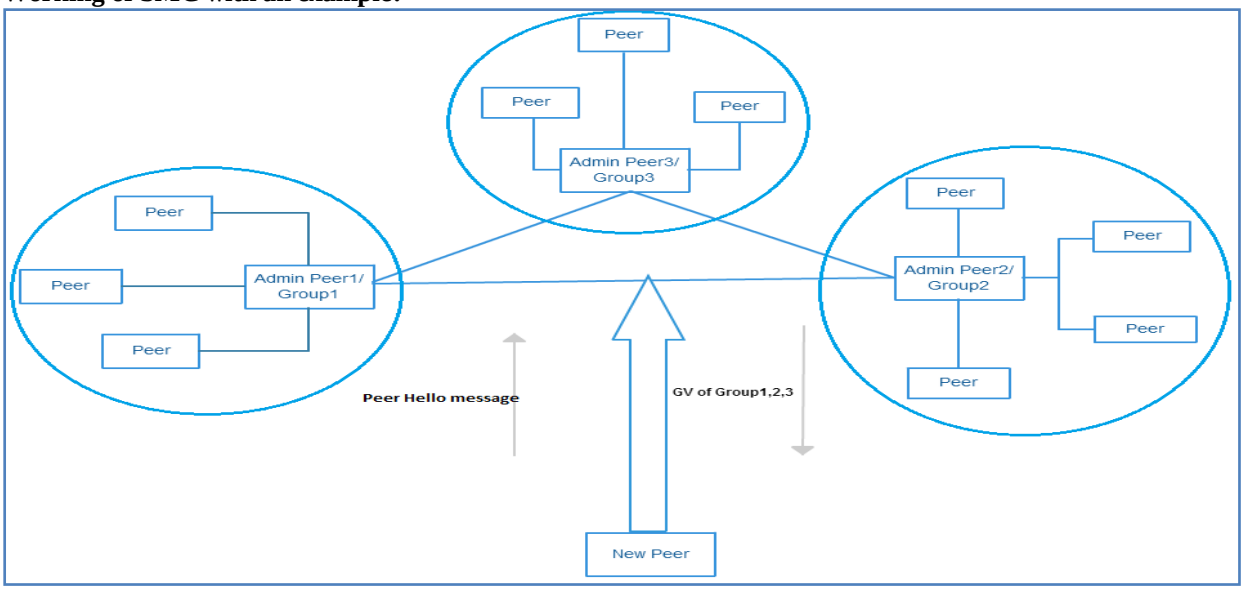

Figure 2: New peer enters system

Consider a P2P network with 3 groups--Group1, Group2, and Group3 (as shown above). Group1 has 3 members, Group2 has 4 members and Group3 has 3 members. All group admins are connected via a common group. When a new peer joins SMC, it first selects a unique username after communicating with the name server. SMC creates 3 folder (Music, Movies and Series) in the peer's home directory. SMC places all the movies, music and series files it wants to share and also wants them in its social profile in these files. SMC starts building the social profile of the peer, after the peer is done adding files to the directories. After building the social profile, SMC initially adds the new peer to a common group containing all the admins. New peer sends a "hello" message to all the admins via the common group channel. New peer also sends some other information necessary for P2P communication. In response, all admins reply by sending their group vectors which will be analysed by the new peer to find matching interest. The new peer will run profile matching algorithm in order to find a group with similar interest. The algorithm returns for each group-no of matches and the average score. New Peer will sort the output. The peer then selects the group with maximum no of matches and least average score.

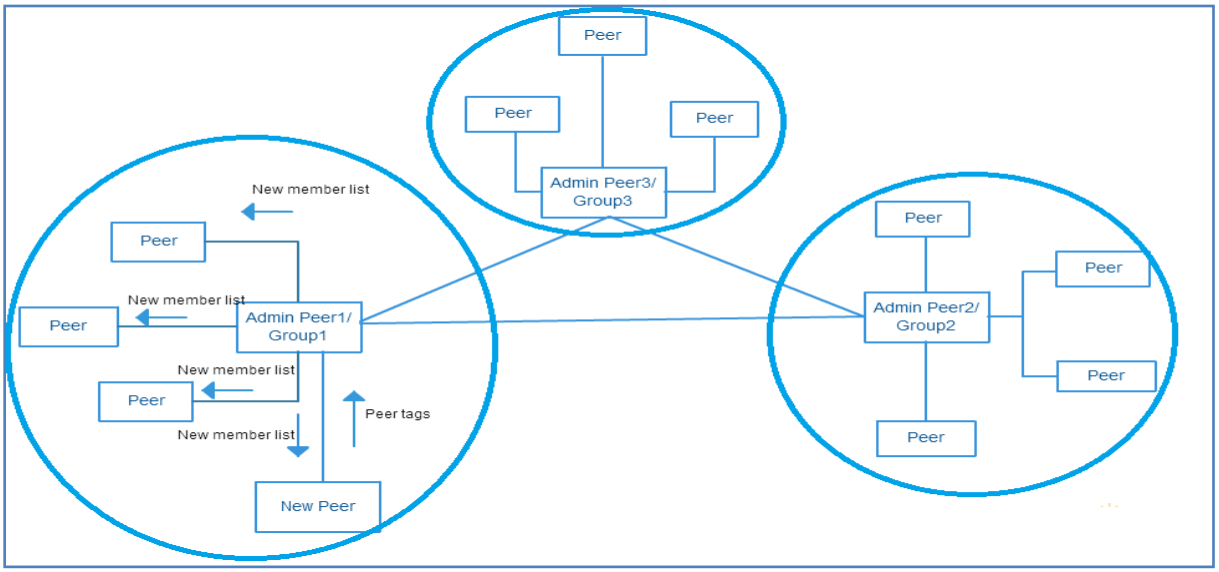

Figure 3: Group formation based on social profiles

New Peer	**Group1**	**Group2**	**Group3**
Music Score	1.41	1.28	1.39
Movie Score	1.31	1.12	1.4
Series Score	1.33	1.5	1.41
(Average Score, No of matches)	(1.35,3)	(1.3,2)	(1.4,3)

If these values fall within the threshold, then the peer will be added to this group. Otherwise the peer will have to form a new group and becomes the admin. According to the profile matching algorithm, the new peer joins the Group1 because all number of matches is maximum for group1 (music, movie and series score is below threshold) and average score is the least. New Peer sends a message to the admin ofgroup1 saying it wants to join group1 along with its own social profile. Admin of group1 then adds the new peer to group1, updates the group vector with the tags of new peer. Admin also sends a new member list to all the members of the group indicating that a new peer has joined. After joining the group, the new peer can request the file list of any member in the group. Since all the members have similar interest the peer can expect to find a file of its interest easily. The search queries of the peer will be first sent only to the members of the group so as to reduce the network overhead.

Important Algorithms designed and implemented for SMC:

Algorithm -To add new peer to SocioMediaConnect:

Peer joins Socio Media Connect
Choose unique peer name
Build social profile after files has been added
Add peer to Admin group and wait for 15 seconds
For each advertisement received
Extract profile
Run matching algorithm
Sort all profiles by similarity
If similarity under threshold
Add peer to group
Add peer tags to group vector
start member peer
Else
Create new peer group
Start Admin peer
Create attr.txt file

After this algorithm finishes, a new peer will either be added to a group or a new group will be created. This algorithm also creates the attribute file.

Algorithm 2. Build Social profile

Create File List
For each file in File List
If FileType == Audio
Extract Tags using JAudioTagger
Else If FileType == Video
Extract file name
Run GuessIt to get qualified name
Query OMDB API to get movies/series information
Extract tags from json array
Add tags to peername_Profile.xml

This algorithm will be initiated once the new peer has selected a unique username and placed the files in shared folders. The output of this algorithm is the social profile of the peer. Social profile of the peer is an xml file where there is a separate tag for each tag extracted from the file and its value is equal to the number of objects by that tag. Initially an empty List<Name, Value> is created. Whenever a new tag is extracted from the file, it is searched inside the list to check if already exists. If found then, its value is incremented by one, else a new entry is created in the list <name, 1>

Algorithm 3. Match Social Profiles

Get all group vectors
For each vector
Extract vector
Apply cosine function between group vector and peer vector
Check movies, music and series weights
If all weights < Threshold
Add to List

Cosine Function:
For each social profile, we initially create 3 vectors each for music, movies and series. These vectors contain the weight of each tag which is calculated as follows:

$$\text{weight of tag} = \frac{\text{no. of objects by that tag}}{\text{Total no. of objects}}$$

For comparing two social profiles, we require two sets of such vectors which are as follows:

$\mathbf{Vector1_{music}} = \mathbf{<wt_1, wt_2, wt_3, \ldots wt_{n1}>}$
$\mathbf{Vector1_{movies}} = \mathbf{<wt_1, wt_2, wt_3, \ldots wt_{n2}>}$
$\mathbf{Vector1_{series}} = \mathbf{<wt_1, wt_2, wt_3, \ldots wt_{n3}>}$

$\mathbf{Vector2_{music}} = \mathbf{<wt_1, wt_2, wt_3, \ldots wt_{m1}>}$
$\mathbf{Vector2_{movies}} = \mathbf{<wt_1, wt_2, wt_3, \ldots wt_{m2}>}$
$\mathbf{Vector2_{series}} = \mathbf{<wt_1, wt_2, wt_3, \ldots wt_{m3}>}$

The similarity between two social profiles is calculated using inverse of cosine function. As a result, similarity is in the range $[1,\infty]$, where,

- 1 indicates completely matching social profiles $\frac{1}{\cos(0)}$
- ∞ indicates completely different social profiles $\frac{1}{\cos(90)}$

Similarity is calculated as follows:

$$\mathbf{Similarity_{music}} = \frac{\mathbf{1}}{\frac{\mathbf{Vector1music}\,.\,\mathbf{Vector2music}}{|\mathbf{Vecto1music}|\,.|\mathbf{Vector2music}|}}$$

$$\mathbf{Similarity_{movies}} = \frac{\mathbf{1}}{\frac{\mathbf{Vector1movies}\,.\,\mathbf{Vector2movies}}{|\mathbf{Vector1movies}|\,.|\mathbf{Vector2movies}|}}$$

$$\mathbf{Similarity_{series}} = \frac{\mathbf{1}}{\frac{\mathbf{Vector1series}\,.\,\mathbf{Vector2series}}{|\mathbf{Vector1series}|\,.|\mathbf{Vector2series}|}}$$

$$\mathbf{Similarity} = \frac{\mathbf{Similarity(music) + Similarity(movies) + Similarity(series)}}{\mathbf{3}}$$

Algorithm 4. Search for Resources

Peer submits search query
Send query to all group members
Wait for results from peers
If result is received
Send result to requested peer
Show result in GUI
else if user selects advanced query
send query to all admins
match peer's group profile with other admin's group profiles
if result found
send result to peer
show result in GUI
end the process
Show no result found

IV. Simulation Environment

- We used JXTA to form P2P network. JXTA is set of protocols that can be implemented using their own technology to establish P2P connections with other peers using identical technologies or different implementations of JXTA. We used JXSE 2.6 which is JAVA implementation of JXTA
- In order to extract tags from music files we used JAudioTagger which is the Audio Tagging library used for tagging data in Audio files
- In order to extract tags from movie/series files we used OMDB API which is a free web service to obtain movie information

- In order to speed up process of building profile we used ThreadPool
- In a P2P network, any peer can tap and listen to communicate on between any two peers which can cause security issues. In our simulation , we used JXTA UnicastSecurePipe which automatically encrypts and decrypts messages and also shares public key certificates with peers
- While clustering threshold for cosine function used was 1/cos (30)
- Each peer generated around 3000-10000 tags depending on the number of files it possesses

V. Performance Evaluation

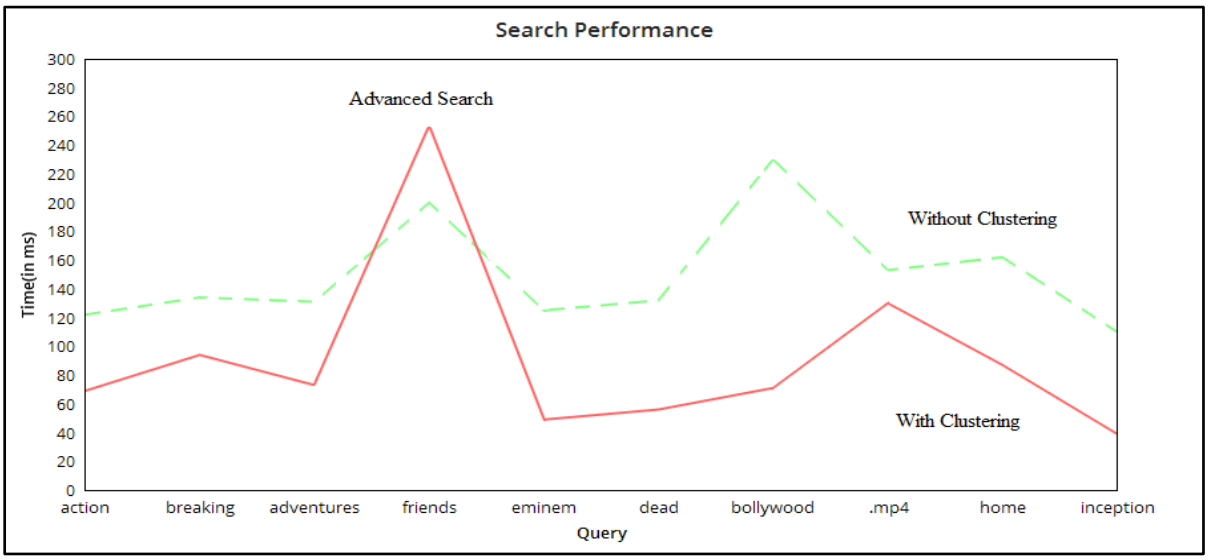

Figure 4: Performance evaluation

Following Observation were made:

1. We Simulated SocioMediaConnect using 100 peers. The average time required for retrieving the search results without clustering is 149.9 milliseconds whereas the average time required for retrieving the search results with clustering is 92.1 milliseconds. Hence, the time required is considerably less for clustered network.
2. The search time increases linearly as new peers are added to the P2P network in both the cases (with and without clustering), but the overall performance of SocioMediaConnect is always better than a normal P2P network.
3. The time required for advanced search in SocioMediaConnect is greater, but the probability of performing advanced search is very low. Advanced search is only performed when the peer does not find the requested resource in the peer group. As a result, advanced search does not degrade the overall performance of SocioMediaConnect and hence the overall output is much better than a normal P2P network.

VI. Conclusion

In this paper, we proposed a method for clustering peers over a P2P network based on the social interests of the peers. With this method, we found that peers will be able find resources of their interests easily from their neighbouring peers present in the same group. Also, we improved the search efficiency in the P2P network by searching only in the peer group because the probability of finding the requested resource was very high in the same group. This resulted in a narrower such window and considerably lesser network congestion. Moreover, there was no need to flood the entire network to find the required resources. Network overhead was also considerably reduced and also lesser number of messages were exchanged between peers. SocioMediaConnect can be easily integrated with any existing P2P system

References

[1]. Kate Ching-Ju Lin, Chun-Po Wang, Cheng-Fu Chou, and LeanaGolubchik, "SocioNet: A Social-Based Multimedia Access System for Unstructured P2P Networks"
[2]. Ching-Ju Lin, Shuo-Chan Tsai, Yi-Ting Chang and Cheng-Fu Chou, "Enabling Keyword Search and Similarity Search in Small-World-based P2P Systems"
[3]. Practical JXTA,Cracking the P2P puzzle DawningStreamsInc
[4]. JXTA in a Nutshell, O'Reilly
[5]. OMDB API, http://omdbapi.com/
[6]. JAudioTagger,http://www.jthink.net/jaudiotagger/

[7]. GuessIT, https://github.com/wackou/guessit
[8]. Clustering Techniques, http://www.ise.bgu.ac.il/faculty/liorr/hbchap15.pdf
[9]. JXSE-2.6, https://jxse.kenai.com/
[10]. Cosine Similarity Function, http://en.wikipedia.org/wiki/Cosine_similarity