Analysing Input Effort during Online Conference in a Client Server Application

[1]Saddam Hussain Malik, [2]Muhammad Asim Ali

[1, 2] Shaheed Zulfiqar Ali Bhutto Institute of Science and Technology, Karachi Pakistan

[1]saddamhussainmalik1987@gmail.com
[2]muhammad.asim@szabist.edu.pk

***Abstract*—Searching the recorded chat, newsgroup, mailing list etc is easy, which are the text archive generated by the persistent conversation, but it is not very expressive or accurate for the social patterns to tell the user about pointed information that user requires nor is it very expressive in showing different user's involvement and participation and also how much user is productive in the chat conference. In this paper discussion is on different designs of the graphical interfaces through which social chat structure is visualized and displayed but major focus would be on the participation of the different users using chat logs, world logs and utterances for the graph chart representation of users. With this we will discuss the problems and how the problems are minimized in this area.**

***Keywords*—** Online, conference, chat, chat graph, chat visualization, graph chart.

I. INTRODUCTION

It is known that with the help of technology it’s easier for the users to connect with other people around the world, making friends and chat with other people through chat rooms. Online chat offers real time direct transmission with which passage of messages can be done between sender and receiver. Delay of messages may happen but won’t hamper the communication flow in any direction.

As we realize that a large portion of the online discussion or correspondence is textual. That is on account of it is somewhat because of innovation history like for messages printed interfaces were the standard, chat rooms and news gatherings were made. Which brought about sharing of thoughts, text has an incredible qualities with which individuals can without much of a stretch express their words, thoughts and arrangements and so forth.

Persistency is one of the feature of online communication. Asynchronous discussions such as Mailing lists or Newsgroups are the asynchronous discussions which are inherently persistent, which leads to problems for the readers to figure out the structure of the discussion. For such problems interfaces or approaches are required for searching and viewing.

Through the graphical representation or Graphical interfaces we can see a lot of hidden textual information. We can see the number of audience in the online chat, the graphics can be added to the communication interface through which you can visualize the conversation in many ways but the problem would be to show salient features of the chat accurately.

In this report we will be outlining client/server based correspondence framework in which diverse clients can keep in contact with each other. Correspondence applications like Skype, Line, WhatsApp and etc. are the best cases. Client/-server gives the platform who upgrades the business methods by fittingly synchronizing the application handling amongst server and customer. Customer server is stage autonomous which gives chance to clients to get to the business usefulness which gets to be straightforward to different clients and basic innovation.

A) Objective of the Study

Our main objective is to design client server based communication and add a feature of graphical flow of the other users in the chat conference. After the users have joined the chat room they will be populated and then messages will be passed to one another the messages saved in the log file consisting the time, user name etc will be used to make chart. The chart would consist the user name and the amount of the chat done by the users, each user's activity will be displayed in the chart. Furthermore how much duplicate words are used during the chat by the users in the chat room. Number of unique utterances that are used by the users during the conversation. Number of utterance types. All these will be used to observe the objective which is that how much the user is active in the chat, how much resourceful the user is, these information will be useful for any organization to check their employee's work contribution after the adjustment of the utterances according to their work chat topic.

II. LITERATURE REVIEW

The objective of literature review is focused on the amount of work which deal with the study of chat and chat applications, the interaction of users with each other, their participation, gathering the graphical representation of the participation of the users, finding the text, words and sentence category and the relevancies resemblance and relation between them and importance of the chat applications in an organization.

In 1973, Doug Brown and David R. Woolley created the first online chatting application known as “Talkomatic” at the University of Illinois. This system was based on the PLATO system. The system features five channels each supporting maximum five people at a time. The messages from all the participants are displayed on the users are they were typed. It became very famous in the middle of 1980s. Later in 2014, a web version of Talkomatic was developed by Brown Wooley. In 1979, first online system to use the actual “cat” developed by Tom Walker and Fritz Thane Dialcom, Inc was introduced for The Source.

In the definition the uneasiness of being a democratic while being a local is also highlighted, like Florence project due to the process in which local delimits the no of groups that bear the opportunity of being head.

A study comparing the validity to the reliability of technique for SVT (Verification tests penalties) and the new version of test known as MIT was forwarded by Horace G. Merchant [1]. It was indicated by the outcomes about the test conducted by MIT methods was way much reliable than the other testing technique i.e. SVT, and when applied on a classroom of a college to predict the performance, the results shows the same greater points of the MIT procedures as compared to the SVT in case of validity [2].

For the field of education, electronic tools to handle discussion boards and meetings and to revolutionize the teaching was designed by Jacqueline K. Eastman in the year 2002 [3]. They played a great role in documenting individual student work and effort by providing accounting their fair share. By this mean, faculty was also facilitated to get access to that share and give their comments other than the class time. Moreover, Students will be facilitated by these electronic tools in shape of developing their own resources. With the help of these tools, the faculty addresses project issues to a group, provide help to students in achieving their objectives in learning and to cater teaching ability in order to keep in contact with the student body. Like this, these electronic chat rooms and discussion forums having group projects and teaches can remarkably enhance the learning and co-operation of the students.

Text Rank [4] was another addition that was developed by Rada Mihalcea in 2004. It was stated by them that their system works well on behalf of the special feature of the system that does not depend on any context that is local, vertex (text unit), but collects the recursive information from the whole description (Text). With the help of graphs using text, it finds the meaning between text of the different units, and machinery used in constructions for the thought of recommendation. Text unit Text prescribe other related units, and quality the suggestion depends on a recursive estimation of the significance of units to make a proposal. In the process distinguished critical expressions text message, prescribes a sentence, 2nd sentence that revises comparable thoughts valuable general comprehension of the text. Sentences that are very prescribed by different sentences liable to be instructive for a provided text, and will in this manner be given a higher score.

Later in 2006, Chat history visualization was presented in their work by Sheelagh Carpendale and Annie Tat using “Crystal Chat” [5]. It displays the personal social network and gives a visual structure that reveals the interaction in timely order and the quantitative information as well as the frequency of text messages that are being sent. It also gives all the previous chatting activity of a person by using the data stored in the logs of chat history. Individual Messages in Crystal Chat are apparent and are always related with their context. Inside a conversation, the messages are collective and the whole conversation is connected depending on individual involved. In the structure, there is also an indication of the content that shows the emotions of the chat which are displayed by the emoticons by using the punctuation.

It was in 1999 that Judith and his team made use of Chat circles and Loom to present status of the different users according to the discussion [6]. Chat Circles and Loom were in the early level of the development so it was too early and inappropriate to give feedback on the client experience. With Chat Circles is our worry with ventures Interface: not just should hypothetically be fascinating and outwardly striking, however it should likewise be straightforward and sufficiently intuitive for individuals to utilize it frequently. The inquiry we then face is how to assess it - what makes a discussion "better"? With Loom, our worry is with themes of finding in a chat chronicle. Our examination of its proficiency includes both surveying the decision of information and presentation mode, both of which can be changed autonomously of each other.

III. METHODOLOGY

Many researchers have worked on the chat visualization through which they can easily figure out the meaningful inputs entered by the users. Many of the researchers have created tools or applications for the input verification like Sentence verification technique (SVT) [7], graph based techniques to check sentences and the relevancies or relationship between the sentences for example if a question is asked or texted and some users have given certain answers this techniques will be used to determine whether the answer is right to the question asked or not. For the research following are the sequential steps followed in order to fulfil our objective and to show our progress which is the participation of the user and analyzing their input effort during the chat conference [8].

A) Making Chat Application

For the chat application we take two arguments on the command number of the port and name of server for the connection. Which establishes the connection with the socket resulting in small area of the entrance and the large area window outlet. Once text a user types in the input box and hit enter, text is then sent to server. The echoing of server occurs for all which is sent by user (client). User (client) then displays all the things which are received by the server. After many clients connect server, we get simple system of chat.

Then the implementation of IM user client is done by the class chat client. Which involves the setup of handing user interaction, interface of the user and message passage. Three parameters are taken by the constructor which includes stream of input, title of the window and stream of the output. On specified flow communication of the Chat Client occurs; Input and output of the buffered stream data is created for communication to be effective communication to upper level. A simple user interface is created for the entrance and exit of the field of text and area of the text. Then we display the window layout, listener of the thread is initiated to accept text from server. During the listening of thread between the execution methods, the observation is done from the stream of input on loop of infinite Strings. Then the chain appears, in the output region we include it and loop is repeated. The exception on input and output may occur if the loss of connection occurs. For such cases print cleaning and exception. This report will be submitted by the end of file exception, method of UTF read. For the clean, firstly we have to null the discussion which is our listener attribute; it means that the thread is completed of the code remainder. Then what we do next is to minimize the field of the input and validate the call function so that the interface gets switched back on once again, and to ensure that connection is off or closed we close or turn off the stream of output.

Point is to be noted that in a finally clause we performed cleanup operation, so that if input and output exception occurs it gets stopped forcibly. Since the user may need the session for reading or any other purpose that is why we don't immediately close window, even if the connection loses.

Now in the method of the handling of event, verification of the two important events in the user interface is needed: In the first one which is event of an action in field of text, means that the user hit enter or return. When the event is caught, message is written by us in the function of flush and stream of output for making sure that message or text is sent immediately. Stream of output data is current output, in order to use function of UTF write for the sending of the string. In case of occurrence of input and output exception know that connection failure has occurred, now we have to stop listening the thread; necessary cleaning will be performed automatically. Closing the window tried by the user is the event which comes secondly. It's programmer responsibility to take care of it; we then stop and hide headphone and frame.

The client is started again from the main method; we ensure that the required number of correct arguments been provided, then the socket is opened in order to specify port and host, and the creation of socket connection is established with chat client flow. Due to the socket connection creation we can show or display the exception thrown and exited. And now comes the chat server which can accept many connections from the clients also the distribution of all the things they reads from client. It is related or wired directly to it, to write and read formats of the strings in the format of UTF. The program would consist of two classes: first class in which there is server of the chat which accepts the connections of the clients after that assignment of the new connection is given to them by the objects manager. The role of class of handling chat is message listening job along with distribution of the messages to the clients that are connected. The thread which is the main one handles all the new connections and another thread which handles the chat is assigned for each and every customer of the chat. For every new client which connects the server; feature of a new instance is provided to the connection by the server of the chat, where messages sent by the new client are received. In the class of handling chat, the maintenance of the list of managers currently there is performed; a process is done by the function process diffusion in which the whole list is sent in textual form by the class of client chat [9].

The manager of the class which handles the chat is related to the management of the connection of individual. It is must to check whether messages or the text sent by the customer are received and afterwards sending other connections those same messages or texts. Statically maintaining the connections list. For handling the clients which are associated extension of the class of thread is done which separately allows thread for client handling. Socket that is attached is then accepted by the constructor; method run is then called by the thread new, which performs the client processing. Socket of the client is maintained by the manufacturer which allows to open the stream of input and stream of the output. Streamed data that is buffered is used again; results in effective input and output also high level data communication methods and strings. The location of wire is from where the method run is. Now the addition of the feed to the vector of the handler of chat manager is done. Here the maintenance of list of managers is done by the vector manager. This variable is static hence the vector instance for the instances and handler of chat class. Therefore handler of the chat class has access to all the connections. It is important that if in future the connection failure occurs it would be very disadvantage to us so we need to withdraw for the connection early. There are two major options for the implementation or creation of such protocol of this application. Which mostly can be supported by many or all applications, so in the case if message or text is sent by user, the information of the header appropriate to it is inserted by user himself. Another way is to ignore the classes set of stream protocol. The header details like insertion and construction is generated by the class of stream automatically, the user is left with the many of the similar interfaces: stream gets the message or text from the user, At As opposed to flushing the present , technique is called which connects those suitable headers What's more sends those messages which are encapsulated.

B) Using Chat Visualization Tools:

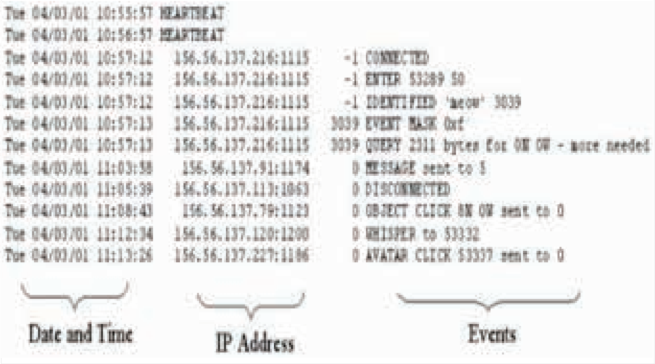
This tool kit is used to generate the text files and graphs based on the activeworlds log file. These tools have many applications some of them are;

i. GraphWorldLog Program
ii. GraphChat Program
iii. ChatTime Program

For our research we will be using activeworlds log file which contains three files;

B). a. World log file:

This file is only accessible by the administrator. It contains IP address, data, events and time as shown in figure 1.

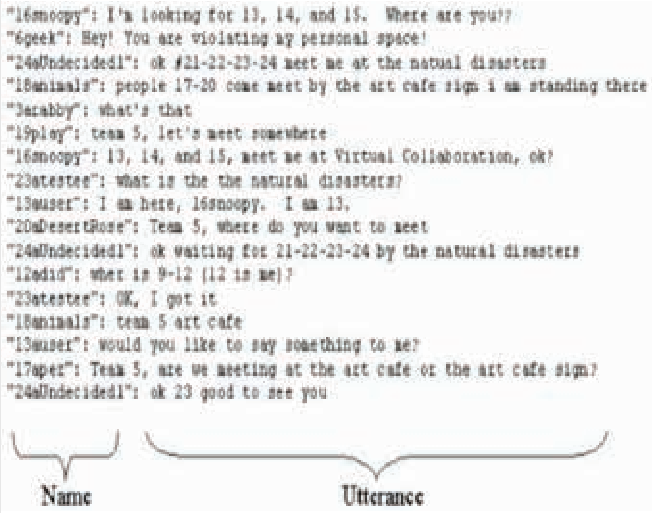


```
Tue 04/03/01 10:55:57 HEARTBEAT
Tue 04/03/01 10:56:57 HEARTBEAT
Tue 04/03/01 10:57:12   156.56.137.216:1115    -1 CONNECTED
Tue 04/03/01 10:57:12   156.56.137.216:1115    -1 ENTER 53289 50
Tue 04/03/01 10:57:12   156.56.137.216:1115    -1 IDENTIFIED 'meow' 3039
Tue 04/03/01 10:57:13   156.56.137.216:1115  3039 EVENT MASK 0xf
Tue 04/03/01 10:57:13   156.56.137.216:1115  3039 QUERY 2311 bytes for 0N 0W - more needed
Tue 04/03/01 11:03:58    156.56.137.91:1174     0 MESSAGE sent to 5
Tue 04/03/01 11:05:39   156.56.137.113:1063     0 DISCONNECTED
Tue 04/03/01 11:08:43    156.56.137.79:1123     0 OBJECT CLICK 8N 0W sent to 0
Tue 04/03/01 11:12:34   156.56.137.120:1200     0 WHISPER to 53332
Tue 04/03/01 11:13:26   156.56.137.227:1196     0 AVATAR CLICK 53337 sent to 0
```

Fig. (1). Log File.

B). b. Chat log file:

In most of the chat applications there is a feature for save the chat log file which contains chat history in the conference where many users are chatting with each other. This file includes names, whispers, participants and utterances as shown in figure 2.



```
"16snoopy": I'm looking for 13, 14, and 15.  Where are you??
"6geek": Hey! You are violating my personal space!
"24aUndecided1": ok #21-22-23-24 meet me at the natual disasters
"18animals": people 17-20 come meet by the art cafe sign i am standing there
"3scabby": what's that
"19play": team 5, let's meet somewhere
"16snoopy": 13, 14, and 15, meet me at Virtual Collaboration, ok?
"23atestee": what is the the natural disasters?
"13auser": I am here, 16snoopy.  I am 13.
"20aDesertRose": Team 5, where do you want to meet
"24aUndecided1": ok waiting for 21-22-23-24 by the natural disasters
"12adid": wher is 9-12 (12 is me)?
"23atestee": OK, I got it
"18animals": team 5 art cafe
"13auser": would you like to say something to me?
"17aper": Team 5, are we meeting at the art cafe or the art cafe sign?
"24aUndecided1": ok 23 good to see you
```

Fig. (2). Chat log file

B). *c. Utterance type file:*

In this file we will have to assign the words or sentences with types, which are represented by the numbers. Each number represents the category to the utterance as shown in figure 3. The few utterance categories that I chose for my research are as follows

i. Greeting
Contains word which means in welcoming or showing the presence of person itself. Sign of welcome.

ii. Explain
Giving statement for something. Referring something to.

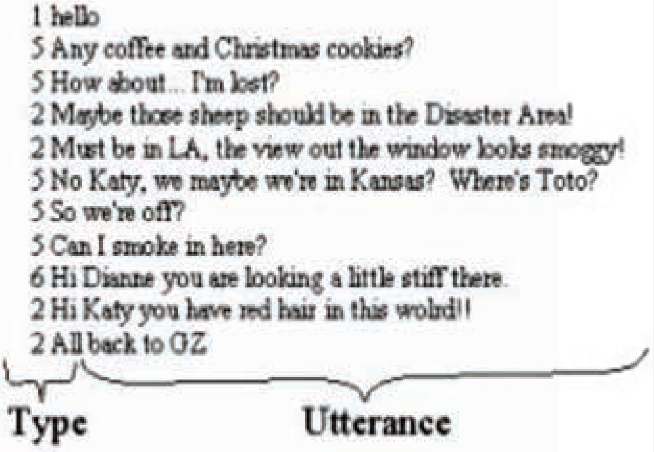
iii. Command
Giving order to someone.

iv. Whispering
Speaking or texting to some specific person like secrecy.

v. Questioning
Asking about something.

vi. Others
Miscellaneous talks or chats.

vii. Farewell / Good bye
Wishing good before leaving the chat conference.



1 hello
5 Any coffee and Christmas cookies?
5 How about... I'm lost?
2 Maybe those sheep should be in the Disaster Area!
2 Must be in LA, the view out the window looks smoggy!
5 No Katy, we maybe we're in Kansas? Where's Toto?
5 So we're off?
5 Can I smoke in here?
6 Hi Dianne you are looking a little stiff there.
2 Hi Katy you have red hair in this wolrd!!
2 All back to OZ

Type
Utterance

Fig. (3). Utterance type file.

C) Merging Application And Tools:

In this step, we will merge the chat application and the tool so that representation of the chat log can be carried out with the help of generated utterances. After the application of the chat starts the ip addresses of the clients will automatically be accepted by the server now the window of the server will be showing a send button, text area where he will input the text which would be send to the other user clients in the conference and the conversation area where all the chat will be displayed in there. Which would also be showing that who has joined the chat room and who has left the room.

For the merging it is simple that we merged the tool's class with the server class of the chat. Where it loads the current chat going on between the clients and server present in the conference. As the messages or texts passage starts it triggers the tool's class which generates the output in the form of the chat graph of the multiple users in the conference. This would all be happening dynamically that as soon as the text or messages are entered the graph would be generated at the same time in accordance to the messages.

For static chat visualizing the data sets were gathered from the internet sources like jabberwocky.com which is free source of datasets. Now the data sets of chat logs were loaded in the tools separately and saved in text file, after loading following results were gathered in which the users were populated in the graph showing their participation, also on the basis of the utterances created the graph shows how much utterances and unique utterances are used by the users in the chat. For now figure 4 shown below shows the number of the users and chat rate done between them also showing the utterances.

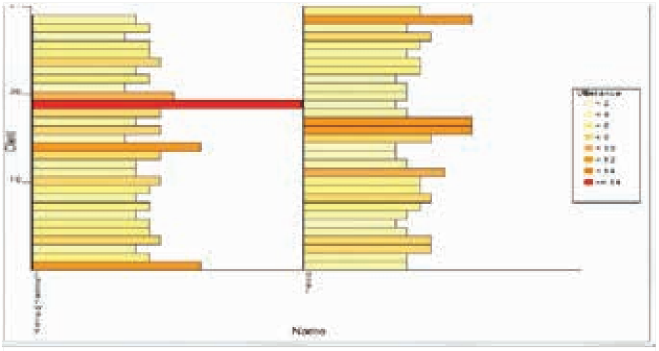


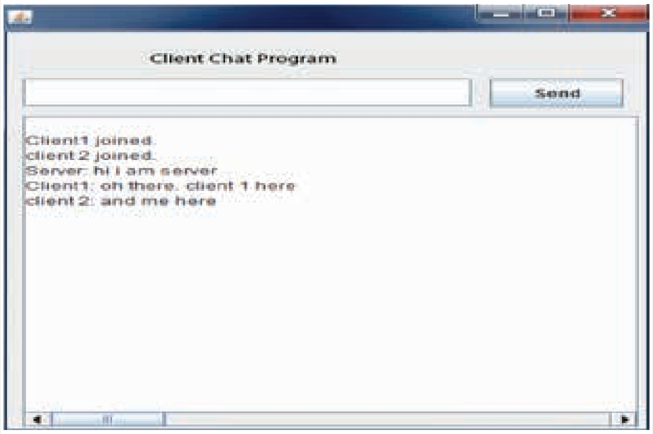
Fig. (4). the graph shows how much utterances and unique utterances are used by the users.

The chat file was loaded manually in the tool program by putting all the chat data in the text file after that loading it on the java it shows us the graph of the populated users, their names and the utterance rate used.

IV. RESULTS

As the merging of the tools with application ended it's time to check the efficiency of the work done so far two types of results were gathered one of them was dynamic result which was gathered through the application and other was static which was done manually inputting the log. The results were gathered out for chat graphical representation on the chat application that we made on Java programming with the integration of the tools used for chat graph and the second representation for the data sets that were used separately on the tool that we used. These results show the input values put by the users on the application which generates the graph helping us to understand the behaviour of the user in the chat rooms in such way that whether the users are using any repeated word etc.

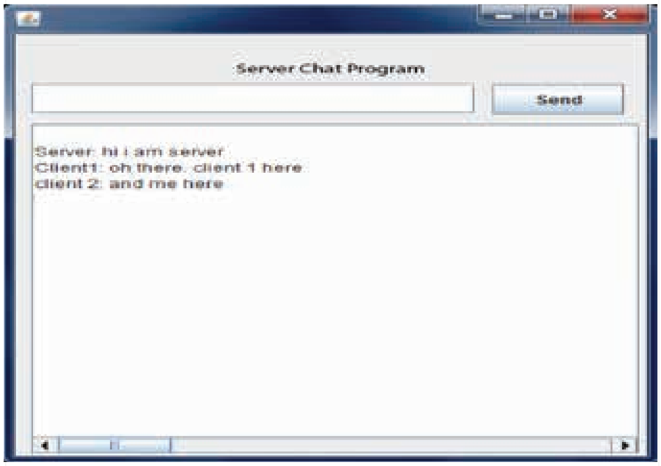
Firstly the result of the chat application communication is given below in which two or three users have joined the chat room and passed few words to each other;



```
Client1 joined.
client 2 joined.
Server: hi i am server
Client1: oh there. client 1 here
client 2: and me here
```

Fig. (5). Client chat room.

Here the interface in the figure 5 above shows that the client joined the room first of all after that another client joined the room. The room was created by the server. Then the chatting was started by the server with greeting words which are saved in our utterance file. After that both the clients relied to the greetings by the server. This is the window which will be shown on the client's screen after getting connected to the server.



```
Server: hi i am server
Client1: oh there. client 1 here
client 2: and me here
```

Fig. (6). Server chat room.

The above figure 6 belongs to the server, in this server will allow several clients to join his chat room and start the chat with other clients. Message or text sending and receiving for the server will be done here.

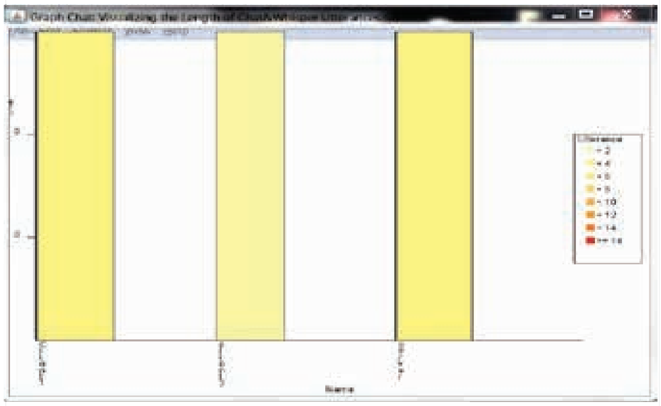


Fig. (7). Client Server Chart.

For the displaying of the users, their name and the amount of the utterances used by them in the chart through this application we don't need to do anything as stated in the methodology that tools are interacted with the application so the during the chat between clients and server this chart will be generated dynamically right on the side of the chat application. But for displaying only this chart we will have to keep the second parameter set as "1" to show the graph shown in figure 7.

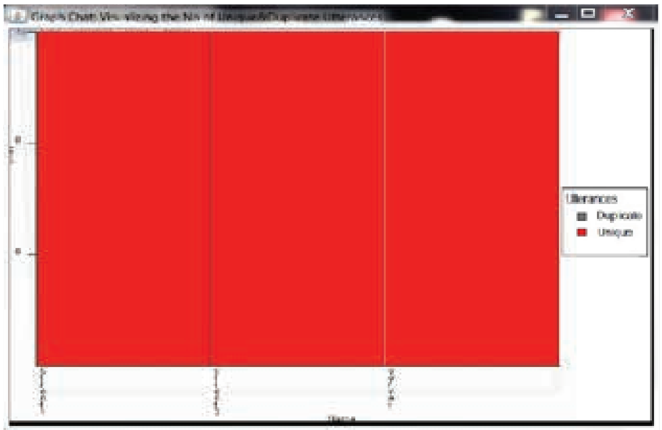


Fig. (8). duplicated words.

To show this chart in which number of duplication of the utterances means how much duplicate words are used either by the server or by the clients in the chat are going to be displayed as the portion of some of the bar as grey which would indicate the rate of duplicated words in the chat otherwise the unique words that are used in the chat are shown or displayed as red bar. Now for this chart the second parameter of the execution file of the chat you will have to set the value as "2" as shown in figure 8

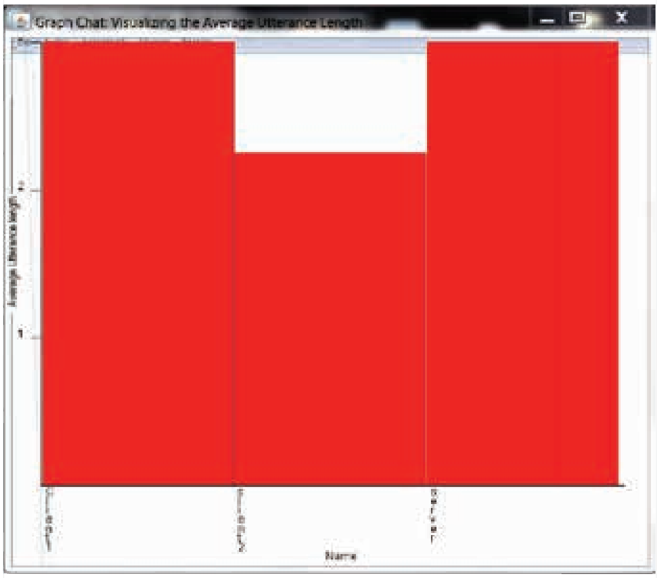


Fig. (9). average length of the utterances.

Now in this graph the result gathered through the chat between clients and server are the number of the average length of the utterances which are shown on the figure above on the y axis and names on the x axis. Figure 9 shows that how much each and single user in the chat has use the utterance over all the chat conference. For this graph change the parameter to "3".

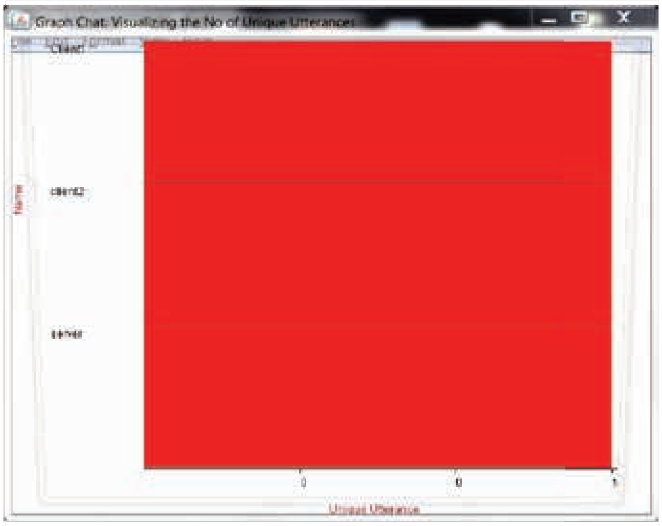


Fig. (10). unique utterances used by the users.

This figure 10 as shown above shows only the amount of the unique utterances used by the users in the chat for this one we have to change the parameter value to "4".

Now for the static results data was gathered from the internet free sources the data is generally the chat between different users on certain topic as shown in figure 11. The chat was saved in the text file and loaded in the bac.bat file where for every graph as shown previously for the dynamic will be shown in the same way for static data by changing the parameters for the desired graph.

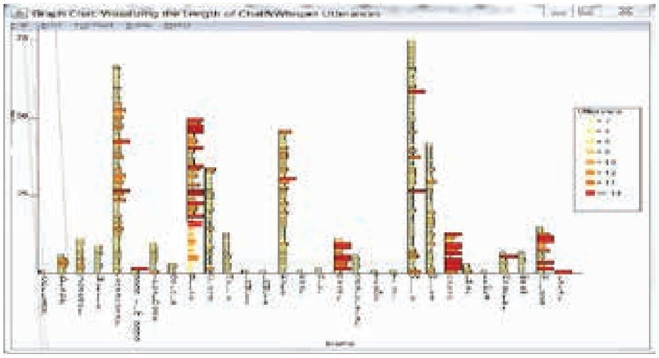

Graph Chat: Visualizing the Length of Chat&Unique Utterances
Utterance
Name

Fig. (11). Static data.

This is the result gathered from the chat data taken from the internet in which name of the users, chat done by users and utterances used by each of the users is shown, which can be figured out easily by the lightness or the darkness of the colour. In the bac.bat file setting second parameter as "1" would show us this graph.

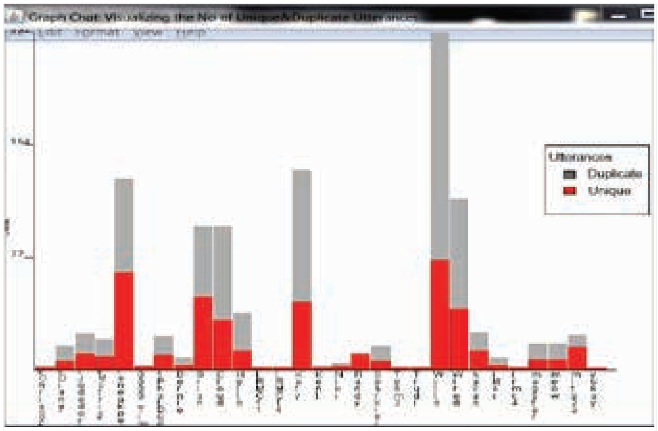

Graph Chat: Visualizing the No of Unique&Duplicate Utterances
154
77
Utterances
Duplicate
Unique

Fig. (12). graph duplication and uniqueness of the words.

Another result for the static data gathered from net sources in this graph duplication and uniqueness of the words used by the users are displayed in figure 12. Parameter value will be "2" in this case.

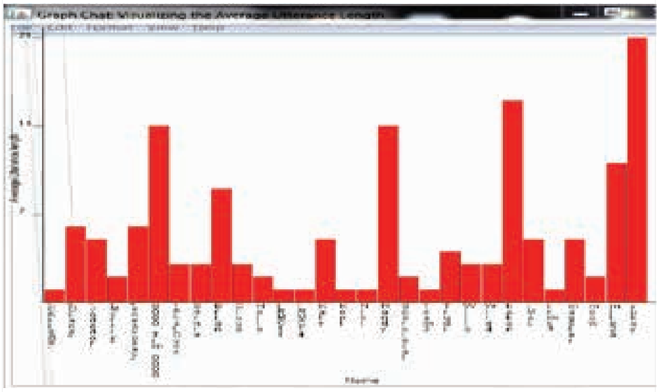


Fig. (13). Average length of utterances.

Above figure 13 shows result is for the average length of the utterance used, parameter "3".

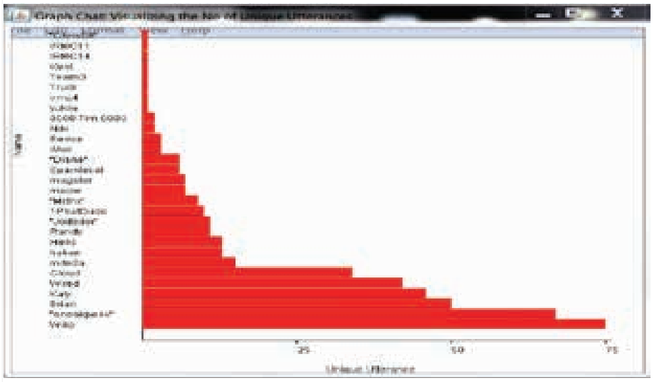


Fig. (14). Number of the unique utterances.

Number of the unique utterances used by the user over the conversation shown in figure 14.

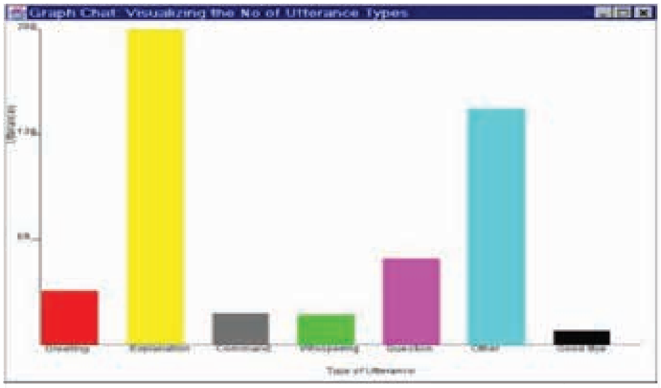
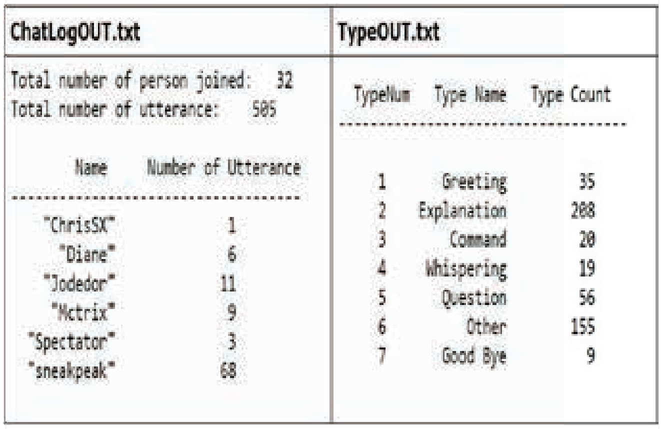


Fig. (15). utterance category.

This figure 15 shows that the utterance category which were made by us from the chat data. And shows how much the rate of these utterances are used in the chat conference by the multiple users in the chat room. This was made additionally in order to show our work on the categories made and used in the tools. This will help us understand how much worthiness of the use is in specific category or subjects created in the utterances.

Table 1. Conference chat data



ChatLogOUT.txt	TypeOUT.txt

```
Total number of person joined:   32
Total number of utterance:    505

      Name     Number of Utterance
----------------------------------
   "ChrisSX"          1
     "Diane"          6
   "Jodedor"         11
    "Mctrix"          9
 "Spectator"          3
 "sneakpeak"         68
```
	```
	 TypeNum   Type Name   Type Count
	-----------------------------------

	    1        Greeting        35
	    2     Explanation       208
	    3         Command        20
	    4      Whispering        19
	    5        Question        56
	    6           Other       155
	    7        Good Bye         9
	```

This table 1 shows the result of all the conference chat that how much users are active, and how much utterances are used by each user. While in the right portion it is displayed that how much utterance categories are used throughout the chat conference, how much in numbers those categories are used and displayed here in type count column.

V. CONCLUSION

In our research paper we have showed the importance of the social networks and their importance in organization. That how they can give more productivity in the organization through the discussion of the problems in the chat room where team members and their superiors are also there to reply their views and create open discussion for suggestion for the problems that are faced and discussion to minimize those problems. Chatting has minimized the problem of hesitation of discussing the subject, problems and other topics face to face. People feel more confident if the person is away and can discuss or advise other person about the issues.

The problem of this is minimized through our research in a way that no wonder every team member of the organization may be present in the conference but some of them would be idle, a few would be discussing irrelevant topic and some participant's conversation would not be meaning anything. Through our research we have discussed that if a person is showing any response to the chat his response would be graphically shown on the application dynamically. So for the supervisor it would be easy to see that how much active the users on the conference are, he would not have to go through the conference again and again, he just has to see the graph and see that who is more active and who needs to work hard on this open conference for the issues faced by organization.

In our research the making the utterance category has also the same purpose through which the server of the application will be able to see that who is focused on which category of the utterance once determined with the help of the graph tools the graphs would be generated and displayed. In such way our research helps the server or the superior of the company to analyze the input effort of the clients or the users in the chat conference. This research helped us a lot to understand the formal and un-formal behaviour of the users in sense of chatting.

REFERENCES

[1] H. G. Marchant, J. M. Royer and B. A. Greene, "Superior reliability and validity for a new form of the Sentence Verification Technique for measuring comprehension," *Educational and Psychological Measurement,* vol. 48, no. 3, pp: 827-834, 1988.

[2] D. Haghighi, A. Y. Ng and C. D. Manning, "Robust textual inference via graph matching," In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing (ACL),* 2005.

[3] J. K. Eastman, and C. O. Swift, "Enhancing collaborative learning: Discussion boards and chat rooms as project communication tools," *Business Communication Quarterly,* vol. 65, no. 3, pp: 29-41, 2002.

[4] R. Mihalcea, "Graph-based ranking algorithms for sentence extraction, applied to text summarization," In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions, (ACL),* 2004.

[5] Tat, and S. Carpendale, "Crystal Chat: Visualizing personal chat history." In *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06),* 2006, vol. 3.

[6] J. Donath, K. Karahalios and F. Viegas, "Visualizing conversation," *Journal of Computer-Mediated Communication,* vol. 4, no. 4, 1999.

[7] J. M. Royer, "Developing reading and listening comprehension tests based on the Sentence Verification Technique (SVT)," *Journal of Adolescent & Adult Literacy,* vol. 45, no. 1, pp: 30-41, 2001.

[8] J. M. Royer, B. A. Greene and G. M. Sinatra, "The sentence verification technique: A practical procedure for testing comprehension," *Journal of Reading,* vol. 30, no. 5, pp: 414-422, 1987.

[9] O. Böhlke, "A comparison of student participation levels by group size and language stages during chatroom and face-to-face discussions in German," *CALICO journal,* pp: 67-87, 2003.