2011

Modeling Human Group Behavior In Virtual Worlds

Fahad Shah
University of Central Florida

MODELING HUMAN GROUP BEHAVIOR IN VIRTUAL WORLDS

by

FAHAD SHAH
M.S. University of Central Florida, 2011
B.S. NED University, 2004

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Fall Term
2011

Major Professor:
Gita Sukthankar

ABSTRACT

Virtual worlds and massively-multiplayer online games are rich sources of information about large-scale teams and groups, offering the tantalizing possibility of harvesting data about group formation, social networks, and network evolution. They provide new outlets for human social interaction that differ from both face-to-face interactions and non-physically-embodied social networking tools such as Facebook and Twitter. We aim to study group dynamics in these virtual worlds by collecting and analyzing public conversational patterns of users grouped in close physical proximity. To do this, we created a set of tools for monitoring, partitioning, and analyzing unstructured conversations between changing groups of participants in Second Life, a massively multi-player online user-constructed environment that allows users to construct and inhabit their own 3D world. Although there are some cues in the dialog, determining social interactions from unstructured chat data alone is a difficult problem, since these environments lack many of the cues that facilitate natural language processing in other conversational settings and different types of social media. Public chat data often features players who speak simultaneously, use jargon and emoticons, and only erratically adhere to conversational norms.

Humans are adept social animals capable of identifying friendship groups from a combination of linguistic cues and social network patterns. But what is more important, the content of what people say or their history of social interactions? Moreover, is it possible to identify whether

people are part of a group with changing membership merely from general network properties, such as measures of centrality and latent communities? These are the questions that we aim to answer in this thesis. The contributions of this thesis include: 1) a link prediction algorithm for identifying friendship relationships from unstructured chat data 2) a method for identifying social groups based on the results of community detection and topic analysis. The output of these two algorithms (links and group membership) are useful for studying a variety of research questions about human behavior in virtual worlds. To demonstrate this we have performed a longitudinal analysis of human groups in different regions of the Second Life virtual world. We believe that studies performed with our tools in virtual worlds will be a useful stepping stone toward creating a rich computational model of human group dynamics.

To God, my parents and my advisor

ACKNOWLEDGMENTS

First, I would like to take the opportunity to thank my advisor Dr. Gita Sukthankar, who has been my advisor throughout the Ph.D. program. I am indebted to her for her great insight and guidance at every step for it was her creative genius and attention to detail that made much of this work possible.

Next, I would like to thank my parents for their continued support and unrelenting efforts because of which I am where I see myself today. I would also like to thank various teachers in my undergrad (Shahab Tehzeeb, Syed Zaffar Qasim and Muhammad Khurram) and Zia Khan who inspired me to think bigger and aspire for higher education.

Thanks to my dissertation committee members Dr. Hassan Foroosh, Dr. Michael Georgiopoulos, and Dr. Georgios Anagnostopoulos and also Dr. Fernando Gomez and Dr. Ladislau Boloni for offering their advice and donating their time and suggestions. I also want to thank Philip Bell as my collaborator for work on social recommendation system (covered in appendix).

Fahad Shah

University of Central Florida

Nov 2011

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Massively multi-player online games and virtual environments provide new outlets for human social interaction that are significantly different from both face-to-face interactions and non-physically-embodied social networking tools such as Facebook and Twitter. There are millions of users inhabiting some of the most popular massive multiplayer online games (MMO's) [Dro10], with titles such as World of Warcraft boasting about 12 million subscribed users [Rel10], and millions more playing the increasingly pervasive (2D) MMO's like Farmville [Nie10]. Second Life is arguably [Rep09] one of the most popular MMOG's (with over 23 million registered users [Yok10]). It is a departure from many games in that it doesn't have a task-oriented (role playing) nature; one is free to do what one wants in Second Life, which not only stimulates the social aspect of the game but also allows for the more motivated users to establish role-playing sims within the game itself. Indeed the large user base and relative freedom in terms of content creation and programmability of the platform was the foremost factor in pushing us to consider Second Life as our platform of research. This broadening user base for MMO's also presents a challenge for researchers to explore the interactions occurring in these virtual worlds.

[IAR09] characterizes virtual worlds as having the following defining characteristics:

1. **Graphical landscape**: The central characteristic of a virtual world is that the primary interactions occur in a rich graphical landscape. This can be a 3-Dimensional (3D) client (like that of Second Life) or a 2-Dimensional (2D) browser medium.

2. **Avatars**: User presence in the virtual world is by means of a visual entity that functions as the "user" in the virtual world.

3. **Persistent**: A virtual world has a notion of "reality", hence, the state resulting from the actions of the user avatar persists between log offs. This can include modifications of the user avatar, as well as modifications to the virtual world.

4. **Shared**: Changes resulting from the user avatar actions in the virtual world are visible to the other users of the virtual world (as perceived by the avatar).

5. **Massive**: To facilitate understanding of the human behavior, the virtual world must have enough users for realistic social patterns to manifest. Preferably the virtual world should have some densely populated regions to increase the frequency of social interaction.

6. **Goals**: There may or may not be a goal built into the virtual world by the designers. For instance, World of Warcraft has roles that the user needs to assume as part of completing quests or missions. On the other hand, the virtual world of Second Life does not impose a concept of roles upon the users who are free to select their own goals.

1.1 Existing Research on Virtual Worlds

It is to be noted that although virtual worlds are increasingly becoming a popular choice for research by social scientists [Bai07] much of the research is anthropological or focuses on writers' accounts [MS07, Boe08]. However, this does suggest the belief that many real world behaviors are reproduced in virtual worlds.

Many virtual world studies on social behavior in virtual worlds focus on a qualitative rather than a quantitative analysis of hypotheses, and overlook the person behind the avatar. Demographic studies have largely focused on well-understood areas such as marketing [Int08] or game design [DM04]. Some of the more relevant research include a study of personal space use in Second Life [YBU07], that revealed that male/female dyads stood closer together than male/male or female/female dyads. This finding illustrated consistency between real world and virtual world use of personal space. Another study of human behavior was performed in World of Warcraft, analyzing the guilds [WDX06]. Their findings revealed that the smaller guilds (those with fewer than 10 members) are composed of people who were already acquaintances in the real world. Finally a study was organized in Second Life to explore the hypothesis that virtual world use of the virtual currency mimics that of ther real world users [CCH09]. Their findings show that it is indeed the case, though the individual difference in use of the currency and the influence of culture on the practices was not considered.

1.2 Research Focus

The following characteristics have been identified as highly influential by researchers studying real-world and virtual world human behavior: gender, approximate age, economic status, educational level, occupation, ideology, degree of influence, digital nativity, physical geographic location, native language, and culture. Here, we present some of the most important research topics relevant to modeling individual and group behavior in virtual worlds.

1.2.1 Individual Behavior

1. **Avatar manifestation**: One important research question is how the virtual world characteristics of the avatar relate to the real world characteristics of the user. Is the avatar that users select similar to their real world persona or the opposite? Evidence from the existing literature suggests both possibilities coexist: some users choose similar avatars [Yee08, MGS08] while others go to the extent of gender swapping [Yee05]. Although we include characteristics related to avatar manifestation as part of our models, these research questions are beyond the scope of this thesis [YBU07, MGS08].

2. **Communication**: Communication refers to the means used by the users in the virtual world to contact other users via verbal or non-verbal means.

Verbal: Invariably the most commonly used method for communication in virtual world is textual chat. Textual chat requires no additional hardware, very little bandwidth, increases anonymity, and is more amenable to automated language translation. It often contains jargon that is not found in other mediums (such as blogs) but is very similar to the chat that occurs in chat-rooms on Internet Relay Chat (IRC). It can be made more expressive using the physical embodiment in the virtual world. Virtual world text analysis poses additional challenges beyond text analysis on blogs and web pages, such as: 1) simultaneous conversation between multiple users 2) frequent topic switches 3) user entrances/exits in the virtual world 4) lack of available corpora for data of this type.

Partitioning individual conversational exchanges is a stepping stone toward research on groups, their formation, and their time evolution. Of further interest is what can be made of the additional information from the content—can we still make decisions based on what is said as well as who made the utterance? Does it corroborate with the findings from the who-talks-to-whom? Can we predict the environment where the conversation is taking place based on the frequency of exchanges or the content? What is the difference between the results obtained over time versus a single snapshot? Moreover, are there some useful conclusions that can be made about the author from the content [AC05]? It is in answering these questions to which this thesis is devoted.

Non-verbal: Non-verbal communication (while not considered in this thesis) invites thoughts on reasons for using it as a preferred means, its inter-relationship to the use of the non-verbal in real-world and what it implies about the person engaging in using such a medium.

3. **Avatar activities**: Depending on the type of MMO (RPG versus non-RPG), the avatar can engage in various activities. MMORPGs (like World of Warcraft) promote a role based approach where the avatar engages in activities specific to the guild (such as stealing, cooking or fighting). With non-RPG MMOs like Second Life, the avatar has more freedom and can perform activities such as dancing, swimming, horse-riding, visiting libraries, attending meetings, exploring or socializing with other users. One important question is what is the relationship between the user's real world characteristics and their activities in the virtual world? We pay some attention to this aspect of the virtual world, with regards to creating a regional activity map of Second Life for our studies.

1.2.2 Group Behavior

1. **Group formation and dynamics**: There is usually a significant amount of social interaction is massively multiplayer virtual worlds. This might be spontaneous or occur after a period of acquaintance. There might be different types of user groups (task-dependent versus societal), with varying characteristics (short-term versus persistent); an important research question is what factors affect the groups' persistence. We explore this question in the longitudinal analysis component of this thesis.

2. **Economics**: Virtual worlds possess virtual artifacts that may be constructed by the users themselves, earned through gameplay, or purchased using either real-world money or virtual

currency. Second Life boasts a 28.4 million dollar money supply in Linden dollars [Lab10] and 119 million dollars of total LindeX volume for the year 2010 alone [Lab11]. This gives credence as to how Linden Labs is able to maintain the game without a subscription model and indicates the immense importance the in-world economy holds for the general user populace. There has been an increasing body of research on virtual world economic analysis [BSH10, CWS09, Cas05] that aims to draw a parallel between the real world and virtual world. One area of research, has been to identify people using illegitimate techniques such as gold-farming [KAW11, KAW10, AKW09]. Unfortunately conducting detailed research on economic practices requires confidential user data and transaction histories from the gaming companies and is beyond the scope of this thesis.

1.3 Research Contribution

In this dissertation, we introduce new algorithms for predicting links and group membership from unstructured chat data. Then we demonstrate how these techniques can be used to study group dynamics in virtual world of Second Life by performing a longitudinal analysis of groups in different Second Life regions and analyzing the effects of avatar characteristics and group membership on the evolution of the social network.

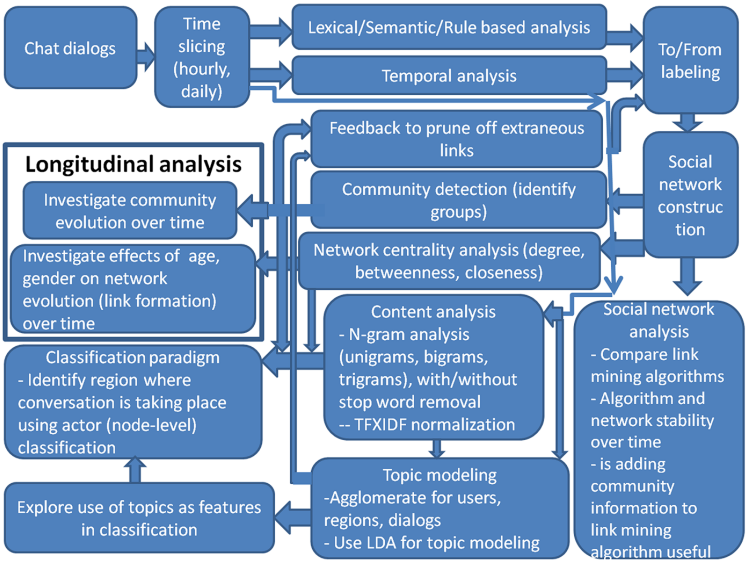


Figure 1.1: Overview of dissertation

1.3.1 Why Second Life?

Second Life (SL) is a massively multi-player online environment that allows users to construct and inhabit their own 3D world. Users are able to chat with other users directly through private instant messages (IMs) or to broadcast chat messages to all avatars within a given radius of their avatar using a public chat channel. The physical environment in Second Life is laid out in a 2D arrangement, known as the SLGrid. The SLGrid is comprised of many regions, with each region hosted on its own server and offering a fully featured 3D environment shaped by the user population. Second Life contains users of widely divergent expertise levels, ranging from complete novices who congregate in the orientation areas practicing basic controls to highly skilled scripters who craft objects and storefronts to sell within Second Life. There is a broad spectrum of group persistence. One can observe rapidly-formed crowds gathered around a temporary attraction, and

also semi-permanent groups of people who share interests either within or outside of the virtual environment. Similar to real-life, these differences are somewhat correlated with SL regions, since each SL region contains a different mix of entertainment opportunities.

Second Life is a unique testbed for research studies, allowing scientists to study a broad range of human behaviors. The ease of creation and interaction with the objects in virtual world enables the rapid exploration of new product designs and customer appreciation studies [Rhe07]. Social scientists are using Second Life to study norms and etiquette in dressing and meeting people [FSS07]. Several studies on user interaction in virtual environments have been conducted in SL including studies on augmented reality [LMZ08], conversation [WTR08], gestures [KRA08], collaborative construction [KA08] and virtual agents [BSE08, PV08]. Many real-world institutions have Second Life portals that allow users to shop for real and virtual items, attend classes, access library materials, visit virtual art displays, and view entertainment broadcasts (see [Sec09a, Dai08] for examples). Additionally, Second Life serves as a platform for communication and teaching in universities [Sec09b] and companies [Sec07, Too07, Sec06]. [ESK11] is a recent example, where IBM's T.J. Watson lab demonstrated that it is possible to hold meetings in Second Life with hundreds of users.

1.3.2 Identifying Social Linkages from Text Chat

fMRI and fossil record studies have revealed that humans possess highly-specialized neuronal machinery to identify social interactions such as organizational hierarchies, cheating, and altruism from subtle social signals [Pow04]. Language is clearly an important part of this process, and Clippinger observes that much of the linguistic apparatus is actually dedicated to expressing social roles and relationships [Cli10]. In fact, it has been hypothesized that language evolved directly to support social coordination [Pin94, Cho00]. Although Second Life provides us with rich opportunities to observe the public behavior of large groups of users, it is difficult to interpret who the users are communicating to and what they are trying to say from public chat data. Network text analysis systems such as Automap [CCD09] that incorporate linguistic analysis techniques such as stemming, named-entity recognition, and n-gram identification are not effective on this data since many of the linguistic pre-processing steps are defeated by the slang and rapid topic shifts of the Second Life users. This is a hard problem even for human observers and it was impossible for us to unambiguously identify the target for many of the utterances in our dataset. In this thesis, one of our main contributions is an algorithm for addressing this problem, Shallow Semantic Temporal Overlap (SSTO), that combines temporal and language information to infer the existence of directional links between participants.

1.3.3 Mining Groups from Social Linkages

Unlike many groups formed in communities and workplaces, groups formed in virtual environments can be rapidly-forming (arising from few interactions), persistent (remaining stable over a long period of time), and agnostic to socio-cultural influences [Car91, For05]. There has been increasing interest in mining community structure in the networks. In general, network sections exhibiting denser linkages among themselves are classified as part of the same community. This phenomena has been studied in social networks, biochemical networks and the WWW [PDF05, GA05, GN02, HHJ03, FLG02]. Understanding the community structure of a network can reveal interesting trends and increase our knowledge of the function and evolution of the system. In our work we restrict our attention to a more recent and state-of-the-art eigenvector based approach to modularity optimization which has been shown to perform very well as reported in [DGD05] and [New06b].

To examine the influence of the extracted groups found with community detection on network evolution, we analyze the system using the dynamic actor-oriented model for network evolution [SSS07]. We use this model to explore the evolution of the network (mined from the dialog exchanges) considering the community membership from previous time period as an actor attribute. This gives us statistical evidence whether the community membership persists over time and provides additional support on the accuracy of our community detection. Using longitudinal network data analysis [Sni05], we consider sequences of network observations extracted from dialog exchanges, along with attributes of the SL avatars, and model them in an actor-oriented model using

RSiena (Simulation Investigation for Empirical Network Analysis) [SR10]. The methodology has been successfully employed in a number of sociological studies on the influences of different factors on group behavior [MSS10, PSS06, GS99, HS03, LML10].

1.3.4 Inferring Location from Context

In virtual worlds, groups are often drawn together by common interests such as shopping, gaming, or scripting. The SLGrid contains many regions, with each region boasting an array of specialized attractions such as shopping markets, gaming grounds, libraries, information kiosks, and scenic views. Users usually frequent a small set of SL regions that offer activities of interest.

Also, Second Life has two specialized types of regions to enable users to better explore game functionality:

orientation areas: specially-designed areas to teach new users how to use the SL navigational controls.

sandboxes: construction areas where more experienced users can construct SL buildings without needing to own SL land.

Text-only chat exchanges lack many informative verbal cues such as prosody and information about socio-economic factors that are highly predictive of real-world social groupings. Another contribution of this research is extracting social structure from unstructured conversational

exchanges—is it possible to determine an actor's social group from a combination of network structure and conversation content in public chat data from a virtual world? Here we evaluate the relative contribution of conversation features vs. network-based ones when learning supervised classifiers to predict a user's region. Our data does not include any SL regions that directly correspond to real-world regions; for instance, certain institutions (e.g., IBM, Princeton University) have created mirror sites in SL. The regions included in the study were selected to span a number of region types. Autonomous SL bots were strategically placed in areas with a high amount of English-speaking user traffic to collect the public chat data from six regions over multiple days. Social networks were constructed from user chat using the Shallow Semantic Temporal Overlap algorithm, which combines semantic and temporal cues to predict network links from conversation data. SSTO uses a combination of semantic and temporal cues to partition unstructured chat data into distinct conversational exchanges between users. We examine the contribution of three types of features for our user classification task:

1. centrality measures (degree, betweenness, closeness),

2. soft community membership,

3. content of user utterances.

Based on our experiments, supervised classifiers trained with network-based features (a combination of community and centrality features) outperform the unigram word features that measure the content of the user utterances. This finding holds true even though the actors in the regions differ substantially across different days. Experiments reveal that there is enough similarity between net-

works emerging in the same regions on different days to classify user groups with an accuracy of 44%. We hypothesize that the form of the network follows the functionality required by the users to pursue their activities. For instance, orientation areas have large groups of transient users speaking briefly to the small set of expert helper users. This characteristic "fingerprint" appears to be sufficiently predictive to enable classification. Combining all measures (network, community, and content) yields the best overall accuracy at correctly predicting the regional origin of user dialog. This shows that while knowing what the people said is important, knowing who talked to whom also provides important information that can be used as a basis for user classification.

1.3.5 Improving Network Text Analysis with Topic Modeling

We also wanted to take a deeper look at the semantic content in the dialogs, rather than relying solely on the network properties, the shallow semantic cues, or the time ordering of the dialogs. This allowed us to measure the predictive power present in the language itself. The intuition was to see if the environment influences the chat topics and if this can be used to decipher the chat location. Another interest was to explore its use in improving the link mining algorithm, based on the proposition that users who share similar topics are more likely to be in the same community.

To extract topics from the text, we use Latent Dirichlet Allocation (LDA) [BNJ03]. There has been some previous work on topic modeling for document classification such as [BNJ03, GS07], but, in our virtual world dialog dataset, the notion of what constitutes a document is unclear. Pos-

sibilities include: 1) creating a new document from each utterance 2) grouping all utterances from a single user into one document 3) grouping all utterances from a single region during a fixed time period into one document. This choice impacts the amount of text in the document, the number of documents, the number of documents per author and many other aspects of the topic. Our results show that using topic modeling for network text analysis yields a higher predictive power than using n-grams. The combination of topics with centrality measures yields better results on the region identification task than a combination of n-grams with any features; however a combination of all the network properties and the topics wasn't able to surpass the classification performance obtained from combining using n-grams. In the next chapter, we provide an overview of the related work on network text analysis for virtual worlds.

CHAPTER 2

RELATED WORK

One of the key elements in this thesis is addressing the problem of constructing social network linkages from public chat exchanges. This is simultaneously useful for analyzing the group dynamics in different Second Life regions and has the potential practical benefit of allowing Second Life land owners to analyze the relative utility of various attractions. Our main contribution is the creation of link-mining algorithms for chat dialogs that use rules and heuristics supplemented with timing and topic information to mine network connectivity. The output of the algorithms is the to-from labeling indicating the sender and recipient of the message. This can be used to construct a social network, using the frequency of exchanges between the users, for a given time-period. We use an additional feature set consisting of latent community information extracted using an eigenvector-based approach to modularity optimization. The key related work in these areas is covered in detail in the following sections.

2.1 Chat Dialog Analysis

Dialog analysis has been previously explored within the Restaurant Game [OR07], where a corpus of human dialog was collected and leveraged to improve the realism of the bot's dialog in a social situation. Although Second Life provides us with rich opportunities to observe the public behavior of large groups of users, it is difficult for even humans to identify who a user is communicating with at a given moment; for instance, it was impossible for us to unambiguously identify the target for many of the utterances in our dataset even with human labelers. Much of the previous work on analyzing chat has been restricted to a small number of users and is topic-specific. Network text analysis systems such as Automap [CCD09] that incorporate linguistic analysis techniques such as stemming, named-entity recognition, and n-gram identification are not effective on this data since many of the linguistic pre-processing steps are defeated by the slang and rapid topic shifts of the Second Life users.

There has been one notable previous work in analyzing chat dialogs - the work by Naval Postgraduate Labs for the NPS chat corpus [FM07]. They have analyzed chat dialogs in online chat rooms and have focused their efforts on identifying sentences as belonging to one of a set of semantic classes using manual annotations and filtering. There work is different from ours in that their main contribution was sentence classification and that they used manual filtering and labelings. Also, the chat room setting differs from the 3D virtual world experience, resulting in different patterns of social interaction. [AM08] is the latest research on this dataset; the authors propose an alternate method to the problem of topic identification.

Another similar effort done within a controlled environment on a smaller corpus is that of [SSB10]. They employed manual annotation at four levels: communication links, dialog acts, local topics and meso-topics, whereas in our case we are concerned with the automation of the first level (communication links). Another important difference is that they imposed structure to their communications by directing the conversation towards a topic and using an arbiter. Our dataset is unique in its size, lack of communication structure, and dynamic groups.

[Thu03] focuses on the problem of analyzing SMS communication, which is by nature bi-directional rather than unstructured. The problem of inferring directional links between participants is somewhat analogous to the task of recovering thread structure in online discussion groups with missing metadata [WJC08]. In [AM08] the authors use a similar approach to ours though their objective was to identify threads; their approach requires certain features, such as sufficient word similarity overlap in discussions belonging to common topics. [WJC08] is another example of related work where the objective is limited to identifying the thread structure of the chat (within the context of voting/opinionating on specific topics).

There are a number of approaches that have been devised by the previous researchers [FM07, AM08, EC10, WO09] for analyzing chat using both classification and clustering. Charniak [EC10] takes a clustering approach to the problem, using a two pass method. Their method utilizes a set of lexical, timing, and discourse features, similar to ours, and weights are learned for different feature types. In earlier work, Sheen [SYS06] and Adams [AM08], investigated a similar clustering approach to the problem that relies on a cosine measure of distance while using a threshold parameter to determine how much closer to the cluster center the utterance can be to be considered for addi-

tion to the cluster. A supervised approach makes it possible to weight the individual feature types based on their predicability (as opposed to uniform weights or using heuristics as in unsupervised approach) and thereby obviating the need for a separate tuning phase (while requiring the availability of labeled training data). Most of the other related work (such as [ARS03, aGM05]) comprises of efforts to cluster the users rather than conversations - with the underlying assumption that each user is participating in one conversation at time. In terms of the lexical features, Sheen [SYS06] and Adams [AM08] demonstrated the use of the n-grams approach (with TF-IDF normalization) to good effect. Time as well as Wordnet augmentation [Wor09] were used as features in Adams [AM08] but their results fail to demonstrate that these are in fact useful features.

The main differences between our approach and Charniak is that they employ a machine learning algorithm for the task that relies on specific timing information. The main disadvantage of this approach is that it requires labeling; also, machine learning might not be able to generalize to an effective level the variations in our much larger dataset, where conversations are more open ended than their dataset which was composed of Linux-specific user queries. Wang ([WO09]) presents an improvement to Charniak ([EC10]) using the information retrieval technique of message expansion, while using the same Linux-specific dataset.

2.2 Social Network Extraction in MMOs

There has been research on the problem of constructing social networks of MMORPG players, for example, [SH04] demonstrate that concepts from social network analysis and data mining can be used to identify MMORPG tasks. In this dissertation our social network analysis is focused toward revealing network characteristics rather than actor characteristics, which is significantly different from prior work at mining social networks from multi-player game data. We wish to identify differences between *groups* of participants rather than between different *actors* within the same social network.

2.3 Community Mining

Networks are increasingly gaining importance as the choice of representational (mathematical) structure for complex systems in many fields — for example physics, biology, chemistry as well as computer science [New03, BLM06, DM03, NBW06]. There has been increasing interest in mining community structure in these networks, with networks exhibiting denser linkages among themselves as opposed to others being classified as part of the same community. This phenomena has been studied in social networks, biochemical networks and the WWW [PDF05, GA05, GN02, HHJ03, FLG02] and has revealed interesting trends about the functioning and evolution of these systems. For example these communities have been shown to correspond to web pages

about a topic [FLG02] or to functional units in metabolic networks [HH07, PDF05, GA05] and are increasingly being applied to study finer nuances that distinguishes these substructures from the whole. For this reason community detection has been proposed via difference algorithms utilizing centrality measures, flow models, random walks, resistor networks and others (a more comprehensive review is provided in [DGD05] and [New04]). In this work we restrict our attention to a more recent eigenvector based approach to modularity optimization that has been shown to perform very well as reported in [DGD05] and [New06b]. In prior work, community membership has been successfully used to identify latent dimensions in social networks - for example [TL09] uses the eigenvector based approach to modularity optimization to extract community memberships that can be used instead of relational features in classifying nodes. Similarly, we use memberships as one type of feature for our region classifier. Our Second Life dataset does not contain social networks that span multiple regions, hence relational features are not applicable for our classification task since all the members of the same network effectively belong to the same class. Communities within USENET have been analyzed by comparing structures of induced social networks for each group using metrics such as size, degree, and reciprocity [MH09]. Our analysis of Second Life communities is similar in concept but uses different techniques for constructing linkages. Recent work [KN09] has compared the relative utility of different types of features at predicting friendship links in social networks; in this study we only examine conversational data and do not include information about other types of Second Life events (e.g., item exchanges) in predicting unobserved links in our social networks. Our aim here was to capture the public dialog exchanges between multiple users to create linkages in a social network, as well as to explore the relationship between

the network properties and the underlying environment where the exchanges took place. Thus, our work represents the first attempt to predict user groups in virtual worlds from a combination of network and community-based measures.

2.4 Longitudinal Analysis Using Stochastic Actor-Oriented Models

Some groups are created for a particular purpose, either by group members or by an exogenous authority. For instance, teams, task forces and production lines all fall under this category of planned groups; these are generally well-described by current AI frameworks such as shared plans [GK99] and joint commitment [CL90]. Planned human groups typically progress through a lifecycle characterized by the Tuckman model of "forming, storming, norming, performing, and adjourning" [Tuc65]. The forces governing emergent groups, which form spontaneously through social interactions between individuals [For05], are less well studied. These groups can be divided into two classes: self-organizing and circumstantial [For05]. The former describes individuals who have aligned their activities in a cooperative system of interdependence after a series of repetitive interactions with a small set of people (e.g., regular customers at a neighborhood bar). The latter are short-lived groups, such as the tourists among a set of pedestrians waiting at a crosswalk. In our proposed research, we are particularly interested in understanding how this ubiquitous set of circumstantial groups influences agent behavior. Psychologists characterize the sense of bonding experienced by group members and the external perception of a group's coherence as the group's

entiativity [CYB03]. One expects typical circumstantial groups to exhibit low entiativity, making them particularly challenging to model.

A model for network dynamics that aims to represent the effects of current network structure on the ongoing changes in the network must take into account the entire network structure when estimating the probabilities of relational changes. This approach may be described as a macro-to-micro modeling, where macro refers to the entire network and the individual tie the micro level. An actor-oriented approach to such a model was proposed by Snijders [Sni95, Sni96, SD97, Sni01b], In this dissertation, we use this model to explore the evolution of the network (mined from the dialog exchanges) considering the community membership from the previous time period as a cue for new time period as an actor attribute to evaluate the influence of community membership on network evolution. This gives us statistical evidence whether the community membership persists over time. The actor-oriented model for longitudinal analysis has previously been used for exploring the effect of smoking and drinking behavior on adolescent friendship network [MSS10], homophily and assimilation among sport-active substance users [PSS06], friendship among university freshmen [GS99], development of social competence in children [HS03], analyzing immigrant personal networks [LML10] and preferential trade agreements among countries [MPS08] and is considered the state of the art technique for exploring network evolution based on psychological and sociological theories about friendship.

2.5 Topic Modeling

Three major approaches to the problem of topic modeling in natural languge processing are: latent semantic analysis (LSA), probablistic latent semantic analysis (pLSA), and latent Dirichlet allocation (LDA). LSA was developed in 1988 by [DDF90] and is also known as latent semantic indexing (LSI) in the context of information retrieval. In LSA a term-document matrix is constructed with columns representing each document and rows the frequency of terms in each document. A low rank approximation is then found to this term-document matrix using singular value decomposition (SVD). Probabilistic latent semantic analysis (pLSA), also known as probabilistic latent semantic indexing (pLSI) [Hof99], adds a probabilistic model to LSA by introducing latent classes (topics) using a (multinomial) mixture decomposition, which gives better statistical properties in terms of objective function, model interpretation and error minimization. Proposed by Blei et al. [BNJ03], Latent Dirichlet allocation (LDA) belongs to a generative family of probabilistic models, which assumes that items (documents) in a corpus are formed as a finite mixture of topic probabilities, such that each word's creation is attributable to one of the document's topics. The use of a Dirichlet distribution as as a prior for the topic distribution of the document is what distinguishes LDA from pLSA.

We hypothesize that discussion topics are a valuable cue for identifying linkages and can also aid in classification of SL regional groups. The topics can be used as a cue when deciding the linkages between the users. Also, instead of using terms as features for classification, we can identify topics from the dialogs and then use the topics as features. It is important to note that

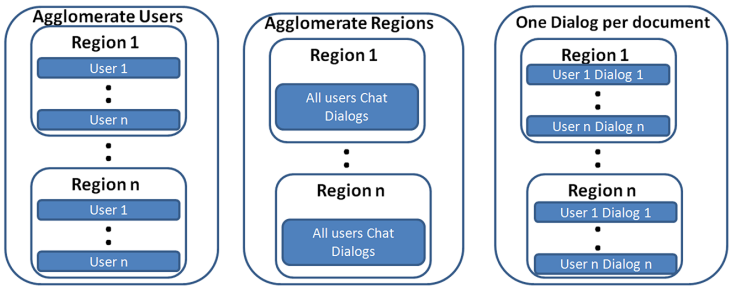


Figure 2.1: Three possible approaches to topic modeling for dialogs.

topic modeling methods are primarily developed around the notion of documents (as belonging to one or more topics), from which the probability for the assignment of a document to a topic is made, which is then used to estimate an assignment of a word to a topic. However, this notion of document is missing in the series of dialog exchanges in the chat data. Instead, it contains a series of alternating dialogs from users, often with overlapping and interleaving conversations, that can shift or abort anytime, whilst the users engage in more than one conversation.

There has been some recent work [RDL10, PSG11] on topic modeling for Twitter posts. Twitter is one of the largest social media outlets, boasting nearly 200 million users and about 110 million tweets per day, as of 2011 [Chi11]. The posts consists of just 140 characters, a restriction that was chosen to suit the SMS, primarily targeting mobile users. In both [RDL10] and [PSG11], they considered each tweet (twitter post) to be a document in its own right. There are two significant differences between tweets and our dialog dataset. While a conversation sometimes ensues between the users on Twitter (in terms of replying to a user post), many of the posts are about one topic only (implicitly bounded by the character limit); moreover, the user replying to the post uses a signal (using the format called Retweet or RT for short), that separates out the conversations into

related tweets. Additionally, users also use hash tags within the posts to mark them as belonging to a specific topic. However, the challenge there is that all the posts are not hash-tagged or marked with a retweet (RT), but the character limitation and the very nature of the medium, restricts the posts to specific topics for the most part. This is clearly not the case in our dataset with unstructured, overlapping dialogues, where the users implicitly designate the recipient of the message by spatio-temporal proximity (which in some cases involve open-ended questions answered by multiple people at different times). This makes our problem both different and more challenging than the topic-modeling for the Twitter posts.

CHAPTER 3

BOT CONSTRUCTION AND DATA COLLECTION

To conduct our study of group social interactions in Second Life, we had to address the following issues:

- Creating chatbots of sufficient realism to record public conversations without perturbing existing social interactions;
- Identifying the linkages between the users from the unstructured dialogs.
- Identifying the communities (groups) from the social network thus obtained.
- Improving link mining through incorporating community as a feedback mechanism.
- Longitudinal analysis of the network data for analyzing how age, gender and community (groups) affect the network evolution (influentials).
- Classification of where the communication is taking place (environment) using a combination of individual, group (community) and content features.

In this section, we discuss the tools that we created or used to solve each of these subproblems. Although we were usually able to "divide and conquer" these problems, the effectiveness of each

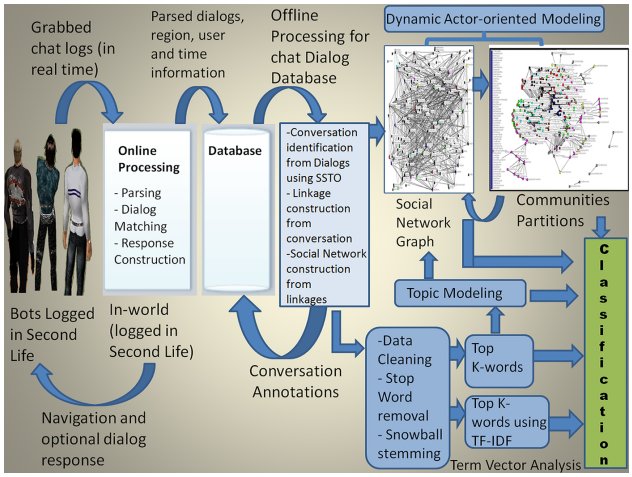


Figure 3.1: Multi-agent architecture for Second Life data collection

module affected the other modules. For instance, early versions of chatbots were less successful at interacting with people and thus had a reduced ability to mine conversational data. The second problem, identifying conversations from unstructured data, is particularly challenging, since users in Second Life often jump rapidly from one topic to another, converse simultaneously with multiple people, and refer to each other by nicknames. However, by leveraging the available temporal co-occurrence and word cues, we were able to reconstruct the most salient social connections within a region.

Figure 3.1 shows the overall data collection architecture. Multiple bots, stationed in different SL regions, listen to all the messages within their hearing range on the public channel. The bots forward chat messages to the server, which parses and conditions messages for storage in the dialog database. Occasionally the server sends the bots navigational commands and optional dialog response if the communication was directed to the bot. Linkages between SL actors are inferred offline by partitioning the unstructured data into separate conversations; these linkages are used to

Figure 3.2: Bot collecting public chat messages.

construct the graphs used in the social network analysis, the properties from which are then used in the classification task.

3.1 Bot Construction

Instead of being controlled by a human user, Second Life avatars can be controlled by an automated agent known as a bot. A bot connects to the SLGrid like a normal user, but is controlled by a program that does not require user interaction. Our bots were implemented using LibOpenMetaverse (LibOMV) [Ope09], an open source .NET based library that allows applications to be able to simulate much of the functionality of the official Second Life client and server software. Using this library, we were able to build multiple bots that log in at a given location and collect all desired data for chat messages within the bot's hearing range on the public channel.

Figure 3.3: Bot responding to dialog from human user

The bot application begins execution by passing login info for a Second Life account to LibOMV. Once LibOMV successfully logs into Second Life, the application enters its main execution loop. Here, the application waits for notification from LibOMV that an event has occurred involving the bot. When a chat message is received, LibOMV passes the following information to the application: the name of the user who sent the message, the time and date the message was received, the region and local coordinates (relative to the bot's current region) that the message was sent from, and the text of the message itself. After this information is recorded in the database, the application returns to the main loop, waiting for the next event to occur. Figure 3.2 shows a bot harvesting data in Second Life.

After observing some initial interactions between the bot and users, we realized that many users were persistent in their attempts to converse with the bot. To handle cases where the user attempts to converse with the bot, we added basic conversational functionality to the bots. Instead of remaining mute, the bot generates a message based on randomly selecting a series of response

Table 3.1: Sample conversation between a user and the bot. Even just by asserting that it isn't a bot, the bot is able to convince that user of its verisimilitude.

User Name	**Dialog**
user1	Welcome, bot1.
bot1	hello user1
user1	what's up bot1?
bot1	yo user1 :)
user1	...yes bot1?
bot1	Hi there user1
user1	Hmm... bot1, your response makes me think you're a bot.
bot1	hey I am not a bot.
user1	Oh, ok.

components that form a message when pieced together. Four components are selected, with five possible choices for each component, giving a total of $5^4 = 625$ possible responses, with the additional variability stemming from the fact that some response components include the username of the avatar being responded to. These responses, while still coming from a fairly limited pool of choices, added some variety to the responses, and made the bot seem more human-like under superficial observation. Additionally, we made the bot's avatar play a typing animation whenever making a response to another user.[1] Table 3.1 shows a sample of the conversation dialog between the bot and a human user; by calling the user by name and asserting that it is not a bot, it deflects the user's suspicions.

Since most users rarely stand still in Second Life, we enhanced the bot by adding rudimentary navigational abilities to enable to move around the data collection area. The bot attempts to move to a specified location from its current location using the LibOMV auto-pilot feature. This feature

[1] A typing animation occurs by default in Second Life when a human-controlled avatar is saying a message.

Table 3.2: Description of regions selected for analysis.

Region Name	**Region Description**
Help Island Public	An orientation area for new users to learn SL usage, scripting, and building.
Help People Island	Similar to Help Island Public, but with fewer people.
Mauve	A sandbox for users to try scripts, build objects, and seek crafting help.
RezzMe	Sandbox and shopping area containing resources like building classes and script guides.
Kuula	Heavily-social sandbox area.
Morris	A entertainment-oriented region with one sandbox, a maze, fun houses, and boating.
Pondi Beach	Beach environment resembling Australia, includes places for sitting, dancing, gaming.
Moose Beach	A densely populated beach environment.

doesn't provide a way for the bot to maneuver around obstacles (including both static obstacles like walls and dynamic obstacles like other avatars) but the generated movement makes the bot appear more natural. In previous work [FSS07] in Second Life, it has been demonstrated that the Second Life environment can be explored effectively through random movement. The addition of these two features greatly lowered the frequency of issues occurring from other users messing with the bot. This allowed the bot to remain logged into areas for extended periods of time, collecting the desired data.

Table 3.3: Anonymized transcript of a public conversation collected in Second Life's RezzMe region.

User Name	Dialog
user1	anyone know if there's a way to turn off notifications in local chat for shields or any other objects when you're in a no-rez zone?
user2	brb need to get drink :)
user3	lol I put the pengiun in the trash can
user4	not too many who knows what it actually stands for
user5	user1, pls can you explain in more detail what you ask? mute it?
user3	GRR YOU DARN PENGUIN
user4	/status
user1	i can paste it in for you:
user5	user3 pls dont pushy ppl
user3	ok sorry
user1	Can't rez object 'animcept4' at {55.9452, 35.1487, 23.4774 } on parcel 'Help People Island'! in region Help People Island because the owner of this land does not allow it Use the land tool to see land restrictions

3.2 Data Collection

To collect data on social interactions in Second Life, we launched bots (each with a different Second Life account and avatar) in eight regions, over fourteen consecutive days, which resulted in a total collection of more than 160,000 utterances. For this task we restrict ourselves to daily and hourly analysis of 5 randomly selected days from this corpus. This comes out to 523 hours of information and about 80,000 utterances (across all the regions). Table 3.3 gives an example of dialog exchanged between users in the RezzMe region. A general description of the activities that the users tend to perform in each region is shown in Table 3.2. The regions fell into three different general categories: 1) orientation areas for new users to learn how to interact with Second Life, 2) sandbox areas that permit users to experiment with building construction, and 3) general entertainment areas (e.g., beaches).

Second Life's multi-user, open-ended setting poses unique challenges to dialog analysis. In such situations it is imperative to identify conversational connections before proceeding to higher level analysis like topic modeling, which is itself a challenging problem. We considered several approaches to analyzing our dialog dataset, ranging from statistical NLP approaches using classifiers to corpus-based approaches using tagger/parsers; however we discovered that there is no corpus available for group-based online chat in an open-ended dialog setting. It is challenging to label the conversations themselves for the large size of the dataset, and the ambiguity in a multi-user open-ended setting makes it difficult even for a human to figure who is talking to whom. Furthermore, the variability of the utterances and the nuances such as emoticons, abbreviations and the presence of emphasizers in spellings (e.g., "Yayyy") makes it difficult to train appropriate classifiers. Since there is no corpus available and the vocabulary is not restricted to English words, parser/taggers perform poorly.

Although there are some cues in the dialog, determining social interactions from unstructured chat data alone is a difficult problem. The earlier work in dialog management has been done primarily for a particular context and in question-answer format rather than in open-ended dialog. Dialog analysis has been previously explored within the Restaurant Game [OR07], where a corpus of human dialog is collected and leveraged to improve the realism of the bot's dialog in a social situation. Unlike the bot in the Restaurant Game, our bot must operate in a broader range of multi-person social situations, rather than the well-defined single-user social scenario.

Consequently, we decided to investigate approaches that utilize non linguistic cues such as temporal co-occurrence. Although temporal co-occurrence can create a large number of false

links, many aspects of the network group structure are preserved. Hence we opted to implement two-pass approach: 1) create a noisy network based solely on temporal co-occurrence 2) perform modularity detection on the network to detect communities of users 3) attempt to filter extraneous links using the results of the community detection.

For our study, we evaluated the performance of two different conversation partitioning algorithms, proposed by us:

1. one based solely on temporal co-occurrence of user conversation (Temporal Overlap Algorithm) and
2. the other based on linguistic cues (Shallow Semantic and Temporal Overlap Algorithm (SSTO)).

3.2.1 Temporal Overlap Algorithm (TO)

The temporal co-occurrence version of our conversation partitioning algorithm separates the data into time intervals (called sessions), which are then partitioned and organized into a hierarchical structure. Each session, marked by a start and an end time, denotes an interval during which a user was chatting. The messages that make up a session are determined by finding consecutive chat messages from a user with less than 20 minutes (the default Second Life inactivity timeout) between any two of the messages. This will not always guarantee that each session represents an actual chat session, but it provides a logical cut-off point that can be used to obtain reasonably

accurate results. While the start time is not null, the following process is repeated: 1) the end of the session with that start time is found; 2) the session marked by that start time and that end time is added to the user's list of sessions; 3) the start time is set to the time of the next message from the same user and region after the current end time.

After the data is partitioned in this manner, each user in a region can easily be compared with other users in the same region to check for overlapping chat sessions. If the chat sessions of users overlap, the users were chatting at same time, which could indicate that a conversation occurred between those users. Note that due the limited range of the bot, even if two users are not directly communicating, there is a fairly high likelihood that they will be reading each other's chat messages since this communication is occurring on the public chat channel.

3.2.2 Shallow Semantic Temporal Overlap Algorithm (SSTO)

Because of an inability to use statistical machine learning approaches due to the lack of sufficiently labeled data and absence of a tagger/parser that can interpret chat dialog data, we developed a rule-based algorithm that relies on shallow semantic analysis of linguistic cues that commonly occur in chat data including mentions of named entities as well as the temporal co-occurrence of utterances to generate a to/from labeling for the chat dialogs with directed links between users. Our algorithm employs the following types of rules:

salutations: Salutations are frequent and can be identified using keywords such as “hi”, “hello”, “hey”. The initial speaker is marked as the *from* user and users that respond within a designated temporal window are labeled as *to* users.

questions: Question words (e.g., “who”, “what”, “how”) are treated in the same way as salutations. We apply the same logic to requests for help (which are often marked by words such as “can”, “would”).

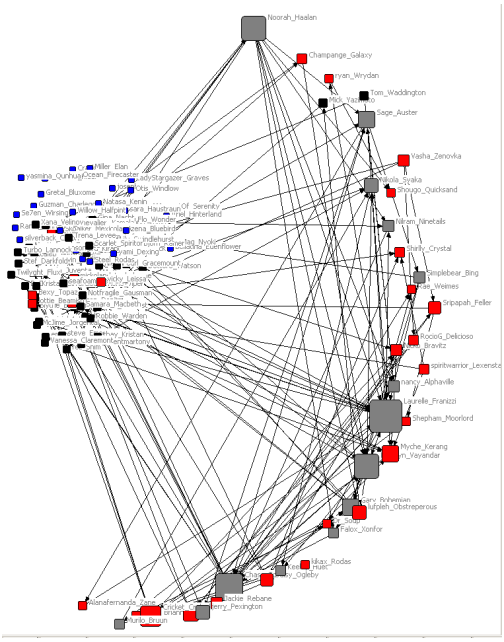
usernames: When a dialog begins or ends with all or part of a username (observed during the analysis period), the username is marked as *to*, and the speaker marked as *from*.

second person pronouns: If the dialog begins with a second person pronoun (i.e., “you”, “your”), then the previous speaker is considered as the *from* user and the current speaker the *to* user; explicit mentions of a username override this.

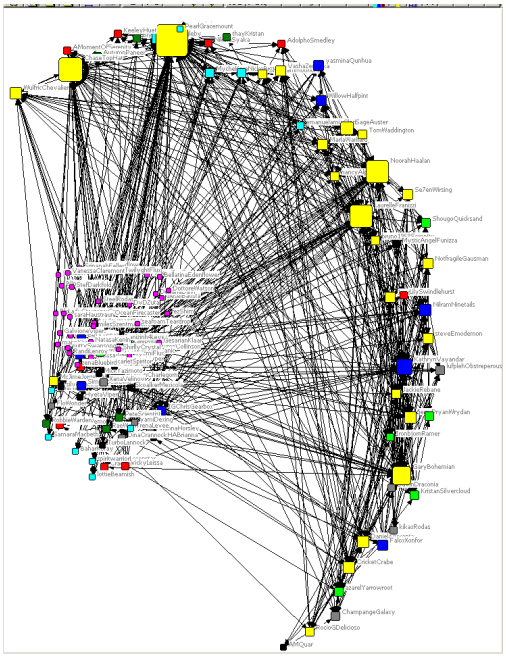
temporal co-occurrences: Our system includes rules for linking users based on temporal co-occurrence of utterances. These rules are triggered by a running conversation of 8–12 utterances.

This straightforward algorithm is able to capture sufficient information from the dialogs and is comparable in performance to SSTO with community information, as discussed in Section 5.3.

We also evaluated the use of the Automap software package [CCD09], a network text analysis system, that incorporates more sophisticated linguistic analysis techniques such as stemming, named-entity recognition, and n-gram identification. However, we found Automap to be ill-suited



(a) Betweenness based layout with size based on degree from linguistic cue conversation partitioning network



(b) Betweenness based layout with size based on degree from temporal co-occurrence conversation partitioning network

Figure 3.4: Network visualizations for linguistic cue vs. temporal co-occurrence conversational partitioning

for working with the raw data since most of the linguistic pre-processing steps were defeated by the slang and rapid topic shifts of the Second Life users and were not able to generate any meaningful networks using that software. In contrast, the simpler temporal cues and domain specific heuristics employed by our algorithms were more robust to these issues; Automap seems to be more appropriate for post-processing the partitioned dataset.

CHAPTER 4

ANALYSIS OF SOCIAL INTERACTIONS

Our first goal was to do social network analysis on the networks obtained from the algorithms. In our analysis, we focused on three main objectives:

group identification: identifying groups of actors having repeated social interactions;

regional differences: characterizing differences between social interactions in different Second Life regions.

social process model: creating a parameterized model of the underlying social process that gives rise to the observed social networks.

Using the information extracted from the raw chat logs, we construct social networks of the users in each Second Life region monitored by the bots. We employed three forms of analysis: 1) k-core analysis to identify groups of actors 2) a comparison of univariate statistics and centrality measures across regions and 3) creating exponential random graph models for the observed networks. In most of these analysis we focus on the network as a whole, analyzing actors and relations (vs. egocentric actors and attributes). Each form of analysis provides slightly different insights into the social interactions.

4.1 K-Core Analysis

A k-core is a graph for which there exists at least *k* paths between any two pairs of vertices; this concept is called structural cohesion in sociology [MW03]. In terms of a social network, each k-core is a group of actors in which an actor is connected to k other actors, making that actor a member of that group. This bottom-up method of analysis quantifies number and size of distinctive groups in the network with sufficient exchanges. This is a simpler form of community analysis than the modularity-based community detection, employed later in our work.

4.2 Univariate Statistics

A statistical view of the social network data describes the data as consisting of a sample of some larger population of possible observations. Relationship measures are viewed as probabilistic realizations of an underlying tendency of relationship strengths. The *mean* of the tie strength can be calculated considering the network as a whole and taking the mean of all tie strengths over the entire adjacency matrix containing all of the actors in the network. To evaluate the hypothesis that this mean value is zero (indicating a lack of true social connection), we perform a zero mean test and use it as a screening method to eliminate from consideration those networks that are composed of random communications between actors.

4.3 Measures of Centrality

An alternate method to analyze networks is to calculate the centrality measures of nodes in the network, which correspond to how connected nodes are to other nodes. To facilitate statistical analysis, we selected a continuous measure of centrality, *betweenness*. Betweenness, in the context of binary data, examines actors based on their presence on the geodesic paths between pairs of other actors in the network. The more people the actor is connected to the greater its betweenness, discounting for all the indirect links to other actors.

The algorithm for calculating betweenness is as follows: if b_{jk} is the proportion of all geodesics linking vertex j and vertex k which pass through vertex i, the betweenness of vertex i is the sum of all b_{jk} where i, j and k are distinct. The normalized betweenness centrality is the betweenness divided by the maximum possible betweenness, expressed as a percentage. For a given network with vertices $v_1 \dots v_n$ and maximum betweenness centrality c_{max}, the network betweenness centralization measure is $\sum(c_{max} - c(v_i))$ divided by the maximum value possible, where $c(v_i)$ is the betweenness centrality of vertex v_i [Lin79]. We use the un-normalized individual betweenness centrality measure for our analysis. When we compare social networks, we look at betweenness measure over the entire population of nodes in the network.

4.4 Exponential Random Graph Analysis

Exponential random graph models (ERG) [FS86, SPR06, WP96] are a statistical model of the social processes underlying the observable social network; the observed social network is viewed as one of the many possible networks that could have been generated by the social process. The parameter estimates from an ERG both give general information about the network, such as whether there is reciprocity of communication links, and can be used as qualitative basis for comparison between networks.

We used SIENA [SR10], from StOCNET [BHS06] for ERG analysis. The ERG model is defined by the probability function $P_\theta\{X = x\} = \exp(\theta' u(x) - \psi(\theta))$, where x is the adjacency matrix of a graph or digraph, $u(x)$ is a vector of sufficient statistics (the ones we chose are given below), θ' are the weights for the linear combination of the statistics (to be estimated) and $\psi(\theta)$ the normalizing constant that enures the probabilities sum to one. The number of steps for generating one exponential random graph is given by $rn^2/2d(1-d)$ [SR10], where r is a constant termed the multiplication factor; n is the number of actors; and d is the density of the graph (average degree), truncated to lie between 0.05 and 0.95. Stability is achieved by tuning the multiplication factor and initial gain parameter.

We use the following parameters as recommended in [Sni06] and [RSW06], for the model (convergence is determined using Metropolis-Hastings pseudo maximum likelihood with continuous chain assumption for the Markov chain to make successive draws from the ERGM, using model 14 as explained in [SR10]:

1. The reciprocity effect $\sum_{i<j} x_{ij}x_{ji}$.

2. The alternating k-out-stars effect to represent the distribution of the out-degrees, given by $c^2 \sum_{i=1}^{n} \{(1-\frac{1}{c})^{x_{i+}} + \frac{x_{i+}}{c} - 1\}$, for some value c.

3. The alternating k-in-stars effect to represent the distribution of the in-degrees, given by $c^2 \sum_{i=1}^{n} \{(1-\frac{1}{c})^{x_{+i}} + \frac{x_{+i}}{c} - 1\}$.

4. The alternating transitive k-triangles effect to represent the tendency to transitivity, given by $c \sum_{i,j} x_{ij}\{1-(1-\frac{1}{c})^{L_{2ij}}\}$ where L_{2ij} is the number of two-paths from i to j, $L_{2ij} = \sum_h x_{ih}x_{hj}$.

where x_{ij} is the observed graph (entry for the adjacency matrix with index i, j), c is a constant set by the user and x_{i+} denotes the degree for the i^{th} node in the graph.

4.5 Results

In this section, we present a systematic examination of regional differences in Second Life. Since each region contains different types of activities, we hypothesize that there are observable differences in the groups that frequent different regions. Our method of social network analysis differs from most of the existing work in that we perform the comparison of social networks with different users and no common attributes rather than performing a comparison between the same actors and

different attributes, or different networks and the same attributes. Our goal is to compare different social groups and networks, rather than different actors.

We use four methods of comparison for the social networks. First, we calculate univariate statistics—the mean value and standard deviation of the linkage matrix of the network. The mean value indicates the density of the network (amount of linkages) and standard deviation, shows the variation in the linkages across actors. Next, we use the hypothesis tests for the mean value as a basis of comparison for the networks as a whole. Then we do an analysis of betweenness over all the nodes in the population and compare the networks using a non-parametric test. Lastly, we look at the k-core partitions for the different social networks. The results from the exponential random graph analysis were not conclusive because of the algorithm convergence problems given our dataset. We used two social networking analysis tools, UCINET [BEF02] and Netdraw [net02], to visualize the social networks and perform the statistical analysis.

To study the effects of regional differences/similarities on social interactions, we performed the above mentioned four analysis on the following two datasets:

- an aggregated dataset of multiple days of data from one region visualized in a single network;
- a series of partitioned datasets with each day's communication data divided into a separate network.

We analyzed these datasets to compare the social networks that form in different regions in Second Life. Additionally, we positioned multiple bots in the same region to determine whether positioning the bots in different locations affected our regional analysis. To determine the stability

Table 4.1: Regional univariate statistics for aggregate social network data.

	Help Island Public	Help People Island	Kuula	Mauve	Moose Beach	Morris	RezzMe	Pondi Beach
Mean	0.33	0.18	0.61	0.07	0.07	0.24	0.84	0.55
SD	11.20	2.67	16.54	2.48	1.51	2.65	5.99	22.00
Min	0	0	0	0	0	0	0	0
Max	3756	172	3180	211	118	80	164	2681
Obs.	590592	89102	117992	18090	102080	5550	13572	69960

of our regional comparisons, we evaluated networks formed in the same region over multiple days to examine how different network metrics change over time. We also looked at the effects of the following network post-processing steps: 1) removing isolates from the network data and 2) eliminating common actors from network comparisons. In this section, we highlight the most interesting results obtained from our comprehensive analysis.

4.5.1 Univariate Network Statistics

Table 4.1 displays the value of the univariate statistics for the aggregate social network data aggregated across all days for each region. These network statistics convey important information about the characteristic of the network (e.g., the mean value is the density of the social network). These statistics are calculated from the adjacency matrix, ignoring the diagonal values. A high mean value indicates high network density. Similarly, a high standard deviation (SD) indicates that there is a large amount of difference in linkages between actors.

Table 4.2: Regional univariate statistics (Day 1)

	Help Island Public	Help People Island	Kuula	Mauve	Moose Beach	Morris	RezzMe	Pondi Pondi
Mean	0.6	0.5	0.7	0.3	0.3	2.4	0.7	0.5
SD	4.6	2.8	6.6	3.2	3.2	5.9	5.0	3.8
Min	0	0	0	0	0	0	0	0
Max	192	39	351	53	74	20	118	61
Obs.	72630	4032	28056	1482	8190	42	13572	10302

Table 4.2 gives the value of the univariate statistics for each region (Day 1). These network statistics convey important information about the characteristics of the network and are calculated from the adjacency matrix, ignoring the diagonal values. A high mean value indicates high network density. Similarly, a high standard deviation (SD) indicates that there is a large amount of difference in linkages between actors.

Table 4.3 gives the mean hypothesis test (from [SB99] as implemented in UCINET using bootstrap to compare density to a specified value) results for the same data set for all the regions using a 5% significance level. We use a zero mean hypothesis test to filter out the regions for which the null hypothesis is true, indicating that any ties among actors are due to chance. The results indicate that Mauve, Morris and RezzMe are those regions for which the null hypothesis (that the mean is zero) cannot be rejected at 5% significance level (and therefore they are not included in the table 4.3). The columns give the results of using a hypothesis test to determine whether the mean of the region shown in the column is equivalent to the mean value of the region that appears as the row value. A ✓ indicates cases for which the null hypothesis could not be rejected at significance level α.

By examining the univariate statistics over all regions and days, we observe the following trends:

- Sandbox regions usually exhibit a low mean value and high standard deviation suggesting a low and irregular pattern of communication, which is understandable as most of the users are busy in construction activities. The test for the mean hypothesis further confirms this belief, as all the sandboxes (Mauve, Morris and RezzMe) are observed to have a statistically significant similarity to a mean value zero.

- The standard deviations on orientation islands range from within normal standard deviation values (suggesting regular pattern of communication between users) to high standard deviation values (suggesting greater interaction among few users.) Orientation islands are mostly populated by new users are more comfortable talking to the few people they engage with initially.

- The recreation areas have standard deviation values suggesting regular pattern of communication among the users with about half of the users engaging in communication.

- Orientation and recreation areas have statistically similar mean values, indicating that overall the strength of the communication ties among the actors is similar in pattern for both regions of this type. There is a large amount of socializing in these regions and more chance of dialog exchanges between users.

Table 4.3: Mean Hypothesis Test for Day 1.

	Help Island Public	Help People Island	Kuula	Moose Beach	Pondi Beach	Mean
Help Island Public	N/A	✓	✓	✓	✓	0.561
Help People Island	✓	N/A	✓	✓	✓	0.510
Kuula	✓	✓	N/A	✓	✓	0.655
Moose Beach	**X**	**X**	**X**	N/A	**X**	0.553
Pondi Beach	**X**	**X**	**X**	**X**	N/A	0.521

Table 4.4: K-S test results for aggregate valued data.

	Help Island Public	Help People Island	Kuula	Mauve	Moose Beach	Morris	RezzMe	Pondi Beach	
	B	B	B	B	B	B	B	B	NN
Help Island Public	N/A	**X**	✓	**X**	✓	**X**	**X**	✓	299
Help People Island	**X**	N/A	✓	**X**	✓	**X**	**X**	✓	769
Kuula	✓	✓	N/A	**X**	✓	**X**	**X**	✓	344
Mauve	**X**	**X**	**X**	N/A	**X**	**X**	✓	**X**	135
Moose	✓	✓	✓	**X**	N/A	✓	✓	✓	117
Morris	**X**	**X**	**X**	**X**	✓	N/A	✓	✓	320
RezzMe	**X**	**X**	**X**	✓	✓	✓	N/A	✓	75
Pondi	✓	✓	✓	**X**	✓	✓	✓	N/A	265

4.5.2 Centrality Measure

An alternative to using mean values to measure the connectedness of nodes in the network is the *betweenness* centrality measure, a measure of the nodes presence on geodesic paths between pairs of actors., The betweenness measure for our data did not satisfy the normality condition (calculated with the Anderson-Darlington test [AD51] for normality at 5% significance level). Therefore, we used the Kolmogorov-Smirnov (KS) two sample test for goodness of fit [Mas51] to perform network comparisons. The KS test is a non-parametric test that evaluates the hypothesis that the two distributions come from the same underlying population. For each potential value x, the KS test compares the proportion of X1 values less than x with proportion of X2 values less than x and then uses the maximum difference over all x values as its test statistic ($\mid \max(F1(x) - F2(x) \mid$), where $F1(x)$ is the proportion of $X1$ values less than or equal to x and $F2(x)$ is the proportion of $X2$ values less than or equal to x). It has no underlying assumptions, other than requiring the distributions to come from a continuous distribution.

Table 4.4 shows the K-S test results at five percent significance level, comparing pairs of networks obtained from different regions in Second Life on the basis of the betweenness measure of centrality; these tests were performed on aggregate networks built from conversational utterances across all days. The abbreviation NN denotes the number of nodes. A ✓ indicates the cases for which the null hypothesis for the K-S statistics (that the two distributions come from the same continuous population) could not be rejected at significance level α and a **X** indicates otherwise.

Table 4.5 gives the K-S test results at five percent significance level for comparing pairs of networks obtained from different regions in Second Life on the basis of betweenness for all the days. The social networks based on the day partitioned data show more similarity across regions than the networks created from the aggregate data. NN denotes the node count and ✓ indicates similarity and a **X** indicates otherwise. Values for the centrality measures are un-normalized. We note the following:

- Less similarity in betweenness was observed in aggregated data than in the day-wise data across the different regions.
- Social networks for the same region are similar across multiple days and across multiple sampling points.
- Disappointingly, no definite conclusion can be made about activity-specific regional differences based on betweenness alone.

For instance, over five days of collected data, three days of social networks constructed from the two Help islands are similar, whereas the other two days show significant differences. Kuula, another orientation area, is similar to the two Help islands on most days. For sandboxes, the data agrees on all the days for the three regions (Mauve, RezzMe and Morris) for RezzMe, but for the other two it does not agree on two of the days. For beaches (Pondi Beach and Moose Beach) the data agrees on all the days, which shows that there exists complete agreement for the social networks from these regions.

Table 4.5: K-S Test Results on Betweenness (Day 1)

	Help Island	Help People	Kuula	Mauve	Morris	RezzMe	Moose Beach	Pondi Beach	
	B	B	B	B	B	B	B	B	NN
Help Island Public		**X**	✓	**X**	**X**	✓	**X**	**X**	270
Help People Island	**X**		✓	✓	✓	✓	✓	✓	64
Kuula	✓	✓		**X**	**X**	✓	✓	✓	168
Mauve	**X**	✓	✓		✓	✓	✓	**X**	39
Morris	**X**	✓	**X**	✓		✓	✓	✓	91
RezzMe	✓	✓	✓	✓	✓		✓	✓	7
Moose Beach	**X**	✓	✓	✓	✓	✓		✓	117
Pondi Beach	**X**	✓	✓	**X**	✓	✓	✓		102

If we look at the results across different activity-based regional categories, we observed that beaches (entertainment areas) are similar to orientation areas in most instances for all the five days of dataset. Similarly there is considerable similarity in one of the help islands (Help People Island) and one of the sandbox regions (RezzMe) for the five day data set. So betweenness alone is not a good predictor of the activities commonly performed in a region.

4.5.3 K-Core Group Measures

The network analysis presented in previous measures were predicated on centrality measures, one based on individual nodes (betweenness) and the other (mean) based on global network statistics. An alternate approach is to analyze the social network at local scales based on the frequency with which strongly-connected groups of various sizes are observed. The strictest such criterion is that of a clique (a group of actors that is fully connected among itself but not beyond); N-cliques relax

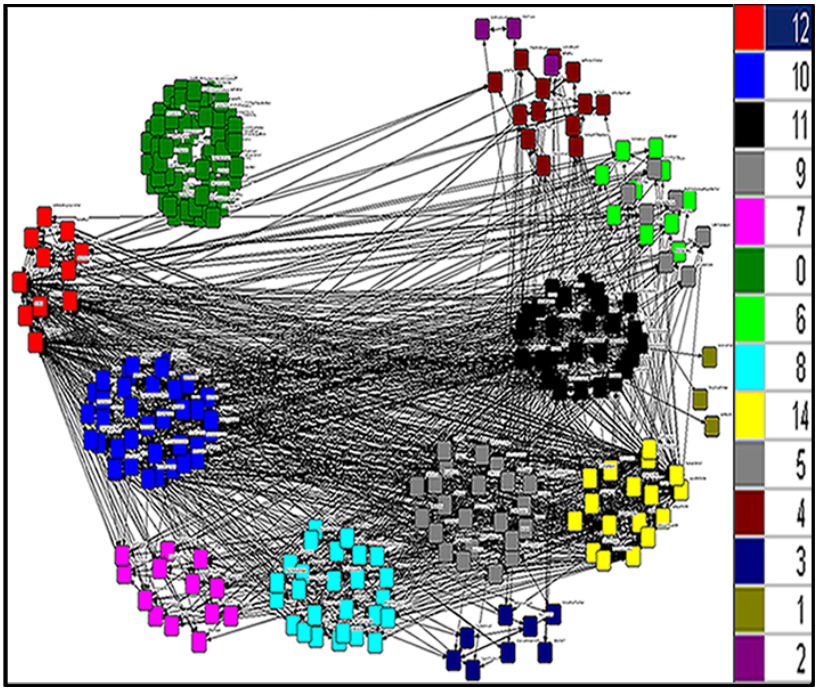


Figure 4.1: k-core partitioning for Help Island Public (Day 1). The legends on right show the color for each degree k-core present.

a clique to include actors if they are connected to every other member of the group at a distance of N. We chose to collect statistics on a more general notion of local groups, the k-core. A k-core is a group where each member is connected to at least k other members in the group (irrespective of its other ties). Modifying k reveals different groups, and in most real-world graphs, group sizes increase as k becomes smaller. We believe that this is the most promising method to partition the graphs into social groups.

Figure 4.1 is a visualization of the k-core partitioning of conversations collected on Help Island Public (Day 1). We see the social network can be assembled from an aggregate of several smaller clusters of strongly-connected components. Table 4.6 summarizes the k-core statistics for the

Table 4.6: K-core summary (#nodes of degree k)

K-core	Help Island Public	Help People Island	Kuula	Mauve	Morris	RezzMe	Moose Beach	Pondi Beach
0	75	26	42	22	51	4	49	36
1	3	0	1	8	9	0	3	8
2	3	11	8	3	10	3	6	18
3	8	5	10	6	2		6	11
4	13	5	20		3		3	5
5	7	9	36		7		13	5
6	10	0	6		9		7	0
7	13	8	11				20	2
8	26		34				0	2
9	22						10	3
10	36							0
11	26							12
12	11							
13	0							
14	17							

different SL regions on Day 1, showing the node count corresponding to each of the k-cores found in a given region. We can immediately make several observations:

- As expected, sandboxes have a high fraction of isolates and few large groups of communicating people.
- Orientation areas and entertainment districts generally contain many larger-sized conversational groups.
- Interestingly, the number of people in a group is much higher in orientation areas than in a corresponding group found at beaches.

CHAPTER 5

COMMUNITY MINING

In this section we examine the accuracy of our conversation partitioning and to/from labeling algorithm using the Frobenius norm by comparing against one hour of hand labeled data. We also present the idea of detecting communities from the social network thus obtained and add this information to the two algorithms to see if the addition makes any improvement over our existing method.

5.1 Modularity Optimization

Modularity (denoted by Q below) as described by Newman in [New06b] measures the chances of observing a node in the network versus its occurrence being completely random; it can be defined as the sum of the random chance $A_{ij} - \frac{k_i k_j}{2m}$ summed over all pairs of vertices i, j that fall in the same group, where s_i equals 1 if the two vertices fall in the same group and -1 otherwise, A_{ij} is the actual number of edges falling between a particular pair of vertices i and j (entries of the adjacency matrix) and $\frac{k_i k_j}{2m}$ is the expected number given by dividing the $k_i k_j$ (the multiple of the

degrees of the vertices i and j) by $2m$, where $2m = \sum_i k_i$ (half the sum of column/row vectors of the adjacency matrix, equaling the total number of edges in the network):

$$Q = \frac{1}{4m} \sum (A_{ij} - \frac{k_i k_j}{2m}) s_i s_j. \tag{5.1}$$

If B is defined as the modularity matrix given by $A_{ij} - \frac{k_i k_j}{2m}$, which is a real symmetric matrix and s column vectors whose elements are s_i then Equation 5.1 can be written as $Q = \frac{1}{4m} \sum_{i=1}^{n} (u_I^T s)^2 \beta_i$, where β_i is the eigenvalue of B corresponding to the eigenvector u (u_i are the normalized eigenvectors of B so that $s = \sum_i^n a_i u_i$ and $a_i = u_i^T s$). We use the leading eigenvector approach to modularity optimization as described in [New06a] for the strict community partitioning (s being 1 or -1 and not continuous). For obtaining the partitioning we choose the eigenvector corresponding to the maximum positive eigenvalue and set $s = 1$ for the corresponding element of the eigenvector if its coefficient is positive and $s = -1$ otherwise. Finally we repeatedly partition a group of size n_g into two and calculate the change in modularity measure given by $\Delta q = \frac{1}{4m} \sum_{i,j \epsilon g} [B_{ij} - \delta_{ij} \sum_{k \epsilon g} B_{ik}] s_i s_j$, where δ_{ij} is the Kronecker δ symbol, terminating if the change is not positive (or adds insignificantly small contribution) and otherwise choosing the sign of s for further partitioning in the same way as described earlier.

5.2 Incorporating Community Membership

Our dataset consists of 5 randomly-chosen days of data logs. We separate the daily logs into hourly partitions, based on the belief that an hour is a reasonable duration for social interactions in a virtual world. The hourly partitioned data for each day is used to generate user graph adjacency matrices using the two algorithms described earlier. The adjacency matrix is then used to generate the spectral partitions for the communities in the graph, which are then used to back annotate the tables containing the to/from labeling (in the case of the SSTO algorithm). These annotations serve as an additional cue capturing community membership. Not all the matrices are decomposable into smaller communities so we treat such graphs of users as a single community.

There are several options for using the community information — we can use the community information on an hourly- or daily basis, using the initial run from either SSTO or the temporal overlap algorithms. The daily data is a long-term view that focuses on the stable network of users while the hourly labeling is a fine-grained view that can enable the study of how the social communities evolve over time. The SSTO algorithm gives us a conservative set of directed links between users while the temporal overlap algorithm provides a more inclusive hypothesis of users connected by undirected links.

For the SSTO algorithm, we consider several variants of using the community information:

SSTO: Raw SSTO without community information;

SSTO+LC: SSTO (with loose community information) relies on community information from the previous run only when we fail to make a link using language cues.

SSTO+SC: SSTO (with strict community information) always uses language cues in conjunction with the community information.

For the temporal overlap algorithms, we use the community information from the previous run.

TO: Raw temporal overlap algorithm without community information;

TO+DT Temporal overlap plus daily community information;

TO+HT Temporal overlap plus hourly community information.

5.3 Results

In this section we summarize the results from a comparison of the social networks constructed from the different algorithms. While comparing networks for similarity is a difficult problem [Pr07], we restrict our attention to comparing networks as a whole in terms of the link difference (using Frobenius norm) and a one-to-one comparison for the *to* and *from* labelings for each dialog on the ground-truthed subset (using precision and recall).

5.3.1 Network Comparison Using the Frobenius Norm

We constructed a gold-standard subset of the data by hand-annotating the to/from fields for a randomly-selected hour from each of the Second Life regions. It is to be noted that there were instances where even a human was unable to determine the person addressed due to the complex overlapping nature of the dialogs in group conversation in an open ended setting (Table 5.2).

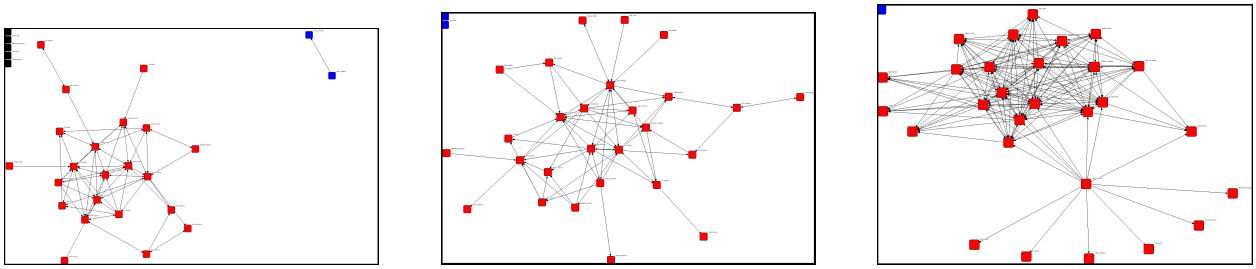
To compare the generated networks against this baseline, we use two approaches. First we compute a Frobenius norm [GL96] of the difference of the adjacency matrices from the corresponding networks (against the hand-labeled network). The Frobenius norm is the matrix norm of an $M \times N$ matrix A and is defined as:

$$\|A\| = \sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}|a_{ij}|^2}. \tag{5.2}$$

The Frobenius norm directly measures how many connectivity differences exist between the two compared graphs and can be used since the networks consists of the same nodes (users). Thus, the norm serves as a measure of error (a perfect match would result in a norm of 0). Table 5.1 shows the results from this analysis.

Table 5.1: Frobenius norm: comparison against hand-annotated subset.

	SSTO	SSTO+LC	SSTO+SC	TO	TO+DT	TO+HT
Help Island Public	35.60	41.19	46.22	224.87	162.00	130.08
Help People Island	62.23	60.50	66.34	20.29	20.29	54.88
Mauve	48.45	45.11	51.91	58.44	58.44	49.89
Morris	24.67	18.92	20.76	43.12	37.54	38.98
Kuula	32.12	30.75	32.66	83.22	73.15	77.82
Pondi Beach	20.63	21.77	21.56	75.07	62.62	71.02
Moose Beach	17.08	18.30	21.07	67.05	53.64	50.97
Rezz Me	36.70	39.74	45.78	38.72	39.01	41.10
Total error	277.48	**276.28**	306.30	610.78	507.21	514.74



(a) Hand labeled network. (b) SSTO labeled network. (c) TO labeled network.

Figure 5.1: Networks from different algorithms for one hour in the Help Island Public region.

5.3.2 Direct Label Comparisons

The second quantitative measure we present is the head-to-head comparison of the to/from labelings for the dialogs using any of the approaches described above (for SSTO) against the hand annotated dialogs. This gives us the true positives and false positives for the approaches and allows us to see which one is performing better on the dataset, and if there is an effect in different Second Life regions. Table 5.2 shows the results from this analysis.

Table 5.2: Precision/Recall values for one-to-one labeling comparison.

		Help Island Public	Help People Island	Mauve	Morris	Kuula	Pondi Beach	Moose Beach	Rezz Me
Total Dialogs		360	184	128	179	227	144	128	97
Hand Labeled	recall	0.6278	0.9076	0.9453	0.6983	0.8370	0.6944	0.6797	0.8866
	total	226	167	121	125	190	100	87	86
SSTO + SC	match	61	59	49	43	63	27	12	23
	precision	0.2607	0.6629	0.6364	0.4216	0.4632	0.3971	0.2105	0.4600
	recall	0.2699	0.3533	0.4050	0.3440	0.3316	0.2700	0.1379	0.2674
	F-Score	0.2652	0.4609	0.4204	0.3789	0.3865	0.3214	0.1667	0.3382
	total	234	89	77	102	136	68	57	50
SSTO + LC	match	61	51	37	39	52	26	12	15
	precision	0.3005	0.6456	0.6607	0.4643	0.4561	0.4194	0.2667	0.4688
	recall	0.2699	0.3054	0.3058	0.3120	0.2737	0.2600	0.1379	0.1744
	F-Score	0.2844	0.4146	0.4181	0.3732	0.3421	0.3210	0.1818	0.2542
	total	203	79	56	84	114	62	45	32
SSTO	match	76	68	51	45	66	30	20	27
	precision	0.3065	0.7083	0.6145	0.4500	0.4748	0.4225	0.3077	0.4576
	recall	0.3363	0.4072	0.4215	0.3600	0.3474	0.3000	0.2299	0.3140
	F-Score	0.3207	0.5171	0.5000	0.3617	0.4012	0.3509	0.2299	0.3724
	total	248	96	83	100	139	71	65	59

5.3.3 Summary

For the temporal overlap algorithm (TO), the addition of the community information reduces the link noise, irrespective of the scale — be it hourly or daily. This is shown by the decreasing value of the Frobenius norm in all the cases as compared to the value obtained using temporal overlap algorithm alone. In general the shallow semantic approach (SSTO) performs the best and is only improved slightly by the loose incorporation of community information. For the SSTO algorithm, the daily or hourly community partition also does not affect the improvement. Table 5.2 shows how the dialog labeling generated from various algorithms agrees with the gold standard notations produced by a human labeler. Since TO only produces undirected links, we do not include it in the comparison. Plain SSTO generally results in a better precision and recall than SSTO plus either strict or loose community labeling. These results are also confirmed from the visualizations for one of the hours of data for all the three methods in figure 5.1, where the SSTO network most closely resembles the hand-labeled network while the TO network contains many spurious links.

The table also shows the subtle difference between the strict and loose community labeling. Looser incorporation of the community results in reducing the noise, by eliminating the spurious connections (while also eliminating some valid ones), resulting in the precision and recall around same mark, showing insensitivity of the SSTO to the community approach and stability in the network generated. The stricter information tends to produce more dialogs but increases the false positives as well, resulting in lower precision and recall values for the algorithm.

The challenging nature of this dataset is evident in the overall low precision and recall scores, not only for the proposed algorithms but also for human labelers. We attribute this largely to the inherent ambiguity in the observed utterances. Among the techniques, SSTO performs best, confirming that leveraging semantics is more useful than merely observing temporal co occurrence. We observe that community information is not reliably informative for SSTO but does help TO, showing that link pruning through network structure is useful in the absence of semantic information.

CHAPTER 6

EXAMINING COMMUNITY EVOLUTION

Homophily is one of the most pronounced factors influencing the formation of the social network [EK10]. This is evident from the fact that our friends are more similar to ourselves than a random sample drawn from the general population. The propensity of people to form links with others who are similar to them is called *selection*. One the other hand, there are qualities of humans that are adaptable such as behavior, interests and activities and can be shaped by our friends; this process is called influence or socialization [EK10]. The interplay between the selection and influence on the homophily of the network can be understood by analyzing multiple snapshots of the network at various time stamps; this process is known as longitudinal analysis.

The statistical modeling of the social networks involves modeling the complex dependencies and their underlying processes, which is a difficult task. Moreover the only data we have for the longitudinal analysis is the discrete time snap shots for the network; however, unobserved network evolution occurs between the observed time snap shots. The approach, then, is to take the first observation being as given and thus ignoring the history resulting in network formation as part of the modeling process. Another key assumption is assuming that the network process is not in a steady state (no equilibrium assumption), since it can bias the conclusion. The explanation for the Actor oriented model in the following section is adapted from Snijders [Sni95, Sni96].

6.1 Longitudinal Analysis

Using longitudinal analysis (with the stochastic actor oriented model) we can further delve into the following:

- Trace the evolution of communities over time.
- Observe the effects of actor attributes like gender and age on the network evolution.

To evaluate the usefulness of the community detection and determine if the patterns determined by the algorithm prevail over time, we devised the following experiment utilizing the longitudinal (cross-sectional) analysis of the network [Sni05]. in relation to the attribute information:

1. We use social network formed from the data collected over three days and determine the community membership for each of the actors in this set.
2. Next, we randomly selected four hours worth of data from a following day to be used for longitudinal analysis for each of the periods in the same day.
3. We used the community membership information from step 1 in step 2 as a constant actor-covariate. The actors that were not present in either of the three days, who were only present in the four hour period under study, were bunched in the same community. The objective here was to explore if the actors with same community membership communicate more among themselves (and thus validating the efficacy of the community membership).
4. We also explored the effect of avatar gender and age on the linkages.

Table 6.1: Number of actors in each hour

	Help Island Public	Help People Island	Morris	Kuula	Pondi Beach	Moose Beach
No. of actors	92	59	44	50	38	50
No. of actors common with previous 3 days	26	15	19	25	20	16

The distribution of the actors for all the regions considered is as shown in Table 6.1. Note that results for Mauve and RezzMe are not included because the sparsity of the network and small size caused the estimation algorithm to fail to converge to any meaningful results. We use the stochastic actor-oriented model from Snijders [Sni01a, Sni05] to explore the co-evolution of the network behavior including the parameters for Similarity (to justify the hypothesis whether actors from the same community prefer talking to each other), as well as for Ego (covariate-related activity) and Alter (covariate-related popularity) to explain the remaining effects in the model (due to dependent variables).

6.1.1 Actor-Oriented Model

Formally, there are M repeated observations for a network involving the same g set of actors; observed networks are represented as digraphs with adjacency matrices $X(t_m) = X_{ij}(t_m)$ for $m = 1, ..., M$, where i and j range from 1 to g. The variable $X_{ij}(t)$ represents the existence of the

tie at t from i to j as being 1 (present) or 0 (not present), with the diagonal being zero ($X_{ii}(t) = 0$ for all i). Longitudinal analysis requires the knowledge of at-least 2 time periods.

The actor oriented model implies that for each change in the network the perspective is used for the actor in question that is involved in the tie change. The assumption is that the actor i controls the set of outgoing tie variables ($X_{i1}, ..., X_{ig}$). The changing of at-most one tie at a time is called a mini-step. The instant when an actor changes his tie, and to which actor the tie change takes place depends on the network structure as well as the attributes of the observed covariates. This instant is stochastically determined in the model via *rate function*, the particular change by *objective function* and the *gratification function*.

The network evolution model, thus, models the actors' decisions to establish new ties or break existing ties (as defined by evaluation and endowment functions), and the model of the timing of these decisions (controlled by rate function). The objective function of the actor x is then defined by the sum of the network evaluation function and the network endowment function as shown in Equation 6.1.

$$u^{net}(x) = f^{net}(x) + g^{net}(x) \tag{6.1}$$

plus a random term. The evaluation function $f^{net}(x)$ and the endowment function $g^{net}(x)$ are defined subsequently.

6.1.1.1 Network Evaluation Function

The network evaluation function refers to the preference distribution of the actor over the set of all possible networks χ. Further, the objective functions for the actors can be actor specific, that is, it can contain actor covariates. The network evaluation function for actor i can be written as given in Equation 6.2

$$f_{net}(x) = \sum_{k} \beta_k^{net} s_{ik}^{net}(x) \tag{6.2}$$

where β_k^{net} denotes the parameters and $s_{ik}^{net}(x)$ the effects (discussed below). The effects that we considered in the network evaluation function are those defined in the following.

Structural Effects The structural part of the network dynamics is modeled by the structural effects (these depend only on the network). We considered the following two structural effects in our model:

- out-degree effect or density effect, called the out-degree as given in Equation 6.3

$$s_{i1}^{net}(x) = x_{i+} = \sum_{j} x_{ij} \tag{6.3}$$

where presence of a tie from i to j is indicated by $x_{ij} = 1$ and $x_{ij} = 0$ denotes the absence.

- reciprocity effect is defined as the number of reciprocated ties as shown in Equation 6.4

$$S_{i2}^{net}(x) = \sum_{j} x_{ij} x_{ji} \tag{6.4}$$

Monadic Covariate Effects Covariates are the variables that depend on the actors (also called actor covariates). For actor-dependent covariates v_j the following effects were used for the analysis:

- Covariate-Alter or Covariate-related Popularity is the sum of the covariate over all actors with which actor i has a tie and is given by Equation 6.5

$$s_i^{net}(x) = \sum_{j} x_{ij} v_j \tag{6.5}$$

- Covariate-Ego or Covariate-related Activity is the actor i's out-degree weighted by his covariate value as given by the Equation 6.6

$$s_i^{net}(x) = v_i x_{i+} \tag{6.6}$$

- Covariate-related Similarity is given by the sum of centered similarity scores sim_{ij}^{v} between the actor i and the other actors j that are tied to i as given by Equation 6.7

$$s_i^{net}(x) = \sum_{j} x_{ij} (sim_{ij}^{v} - \hat{sim}^{v}) \tag{6.7}$$

where $\hat{sim}^v$ is the mean of all similarity scores given by $sim_{ij}^v = \frac{\Delta - |v_i - v_j|}{\Delta}$

where $\Delta = max_{ij} \mid v_i - v_j \mid$ is the observed range of the covariate *v*

6.1.1.2 Network Rate Function

Rate function is an indication of how often the actors take mini-steps. The network rate function λ^{net} (lambda) is given by the following equation:

$$\lambda_i^{net}(\rho, \alpha, x, m) = \lambda_{i1}^{net} \lambda_{i2}^{net} \lambda_{i3}^{net} \tag{6.8}$$

where the factors depend respectively on period m, actor covariates, and actor position. The corresponding rate function can then be computed as the following:

- The dependence on the period can be denoted by a simple factor given in Equation 6.9:

$$\lambda_{i1}^{net} = \rho_m^{net} \tag{6.9}$$

for $m = 1, ..., M - 1$. If we have $M = 2$ observations, the basic rate parameter can be written as ρ^{net}

- The effect of actor covariates with values v_{hi} can be denoted by a factor as shown in the Equation 6.10

$$\lambda_{i2}^{net} = exp(\sum_{h} \alpha_h v_{hi}) \tag{6.10}$$

- The actor's dependence on the position is given by as a function of the actor's out-degree, in-degree, number of reciprocated relations, and *reciprocated degrees* as shown in Equation 6.11:

$$x_{i+} = \sum_{j} x_{ij} \tag{6.11}$$

$$x_{+i} = \sum_{j} x_{ji} \tag{6.12}$$

$$x_{i(r)} = \sum_{j} x_{ij} x_{ji} \tag{6.13}$$

where $x_{ii} = 0$ for all i. The out-degrees effect on λ_{i3}^{Net} is given by $exp(\alpha_h x_{i+})$ if the related parameter is given by α_h for some h, and similarly for the in-degrees and the reciprocated degrees contributions.

6.1.1.3 Gratification Function

The gratification function can be regarded as the weighted sum as depicted in the Equation 6.14, where the possible choices for $r_{ijh}(x)$ are listed in the following, keeping in mind that a factor x_{ij} refers to the gratification for breaking a tie and $(1 - x_{ij})$ the opposite.

$$g_i(\gamma, x, j) = \sum_{h=1}^{H} \gamma_h r_{ijh}(x) \tag{6.14}$$

1. Breaking of a reciprocated tie $r_{ij1} = x_{ij}x_{ji}$

2. Number of indirect connections for creating a new tie (to account for the fact that actors with geodesic 2 are more likely to become friends in near future), denoted by $r_{ij2}(x) = (1 - x_{ij}) \sum_h x_{ih}x_{hj}$

3. Dyadic covariate effect W (given by $r_{ij3}(x) = x_{ij}w_{ij}$) when disconnecting a tie

6.1.1.4 Intensity Matrix

The various functions described combine to define a continuous time Markov chain over the state of all digraphs χ for the set of actors g. The intensity matrix can then be written as in Equation 6.15. This gives the rate for ministeps for actor i multiplied by the probability (if the ministep is taken) that the tie variable X_{ij} is changed.

$$q_{ij}(x) = \lim_{dt \to 0} \frac{1}{dt} P\{X(t+dt) = x(i \rightsquigarrow j) | X(t) = x\} \quad (6.15)$$

$$= \lambda_i(x) p_{ij}(x) \quad (6.16)$$

The Markov Chain thus obtained can be simulated as given in the following

1. Let $\lambda_+(x) = \sum_{i=1}^{g} \lambda_i(x)$, and Δt a random variable, exponentially distributed with parameter $\lambda_+(x)$
2. Actor i is selected to make the ministep with probabilities $\frac{\lambda_i(x)}{\lambda_+(x)}$
3. Once actor i is decided, actor j is chosen according to probabilities in Equation 6.14
4. t advances to $t + \Delta t$, while the tie x_{ij} to $(1 - x_{ij})$

6.1.2 Specification of the Actor-Oriented Model

The main component of the actor-oriented model is the evaluation function [Sni05, Sni01a], as given in Equation 6.2. The objective function can give an idea of the "attractiveness" of the network (or behavior, respectively) for a given actor. Interpretation of the values for the estimates can be explained by the objective function computations that indicate how attractive various different tie changes are.

A variable V's effects can best be understood by considering all effects in the model on which it appears simultaneously. In our network dynamics model the Ego, Alter, and Similarity effects of a variable V were considered and the formula for their contribution can be obtained from the components listed in Equation 6.2 as

$$\beta_{ego} v_i x_{i+} + \beta_{alter} \sum_j x_{ij} v_j + \beta_{sim} \sum_j (sim_{ij}^v - \hat{sim}^v) \tag{6.17}$$

where in Equation 6.17 the similarity score is given by $sim_{ij}^v = 1 - \frac{|v_i - v_j|}{\Delta_V}$ with $\Delta_V = max_{ij}|v_i - v_j|$ denoting the observed range of the covariate v and sim^v being the mean of all similarity scores. The $superscript^{net}$ is removed from the notation for the parameters.

The single tie variable x_{ij} gives the contribution of the tie from i to j, hence, the difference between the values of Equation 6.17 for $x_{ij} = 1$ and $x_{ij} = 0$ can be computed from this equation. Since we are using Siena [SR10] for estimating these parameters and it centers the values around the mean the Equation 6.17 can be rewritten as:

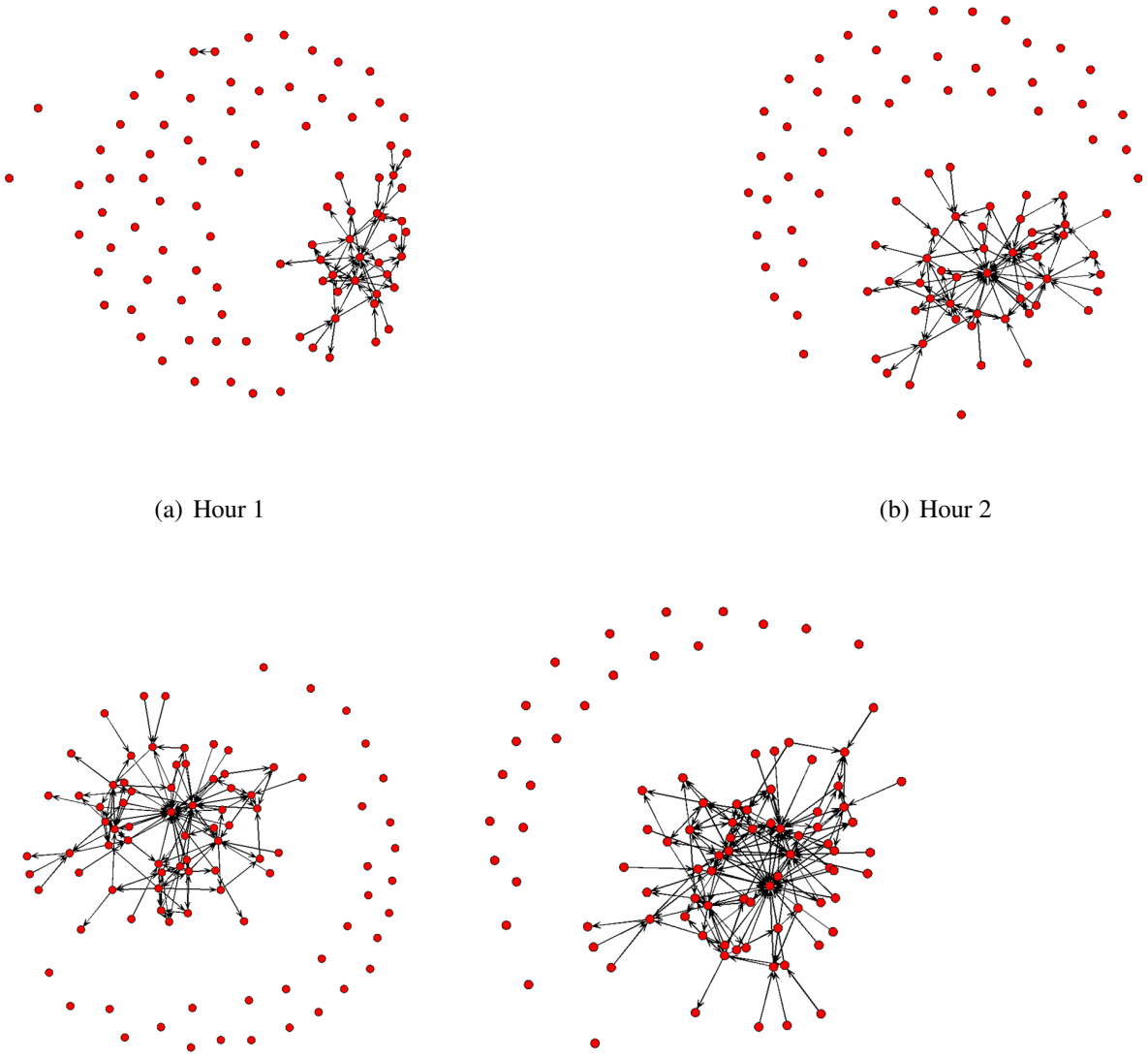
$$\beta_{ego}(v_i - \hat{v}) + \beta_{alter} \sum_j (v_j - \hat{v}) + \beta_{sim} \sum_j (1 - \frac{|v_i - v_j|}{\Delta_V} - \hat{sim}^v) \tag{6.18}$$

6.2 Estimation Results

This section summarizes the statistics obtained from running the estimation on the Ego, Alter and Similarity parameters considered for all the regions, except for Mauve and RezzMe, for which the network for successive hours was small and sparse enough to cause the model to fail to converge, for community membership. For one of the regions (Help Island Public) we also tried looking at the covariates for Age and Gender besides Community. We start with an overview of the network statistics, then proceed to the results and finally discuss the findings.

6.2.1 Network Statistics

First we present summary statistics for the network as shown in the Table 6.2. The average density for all the periods, across all the regions is quite low, indicating the sparse nature of the data. The average degree shows that at observation time 1 all regions, except for Moose Beach, have a low average indicating the asymmetric nature of the ties. This improves to 1 or above by the last time period, where all the regions have reciprocated ties. Help Island Public and Moose Beach have an average degree greater than 1 indicating more communication happening in these areas. Help Island Public is a newbie place, where users are helping each other and new users are repeatedly asking questions. Moose Beach is an entertainment beach, where users usually linger just to enjoy the beach with friends and hence the high cross-talk. Lastly the number of ties are listed for each,



(a) Hour 1

(b) Hour 2

(c) Hour 3

(d) Hour 4

Figure 6.1: Networks for the four hours considered for longitudinal analysis from Help Island Public region.

Table 6.2: Network Density Indicators

Observation Time		Help Island Public	Help People Island	Morris	Kuula	Pondi Beach	Moose Beach
One							
	density	0.007	0.003	0.012	0.007	0.008	0.030
	average degree	0.670	0.169	0.523	0.360	0.289	1.480
	number of ties	61	10	23	18	11	74
Two							
	density	0.012	0.012	0.018	0.009	0.016	0.055
	average degree	1.099	0.695	0.773	0.460	0.605	2.700
	number of ties	100	41	34	23	23	135
Three							
	density	0.017	0.016	0.025	0.013	0.023	0.067
	average degree	1.549	0.949	1.068	0.660	0.842	3.260
	number of ties	141	56	47	33	32	163
Four							
	density	0.022	0.021	0.032	0.025	0.026	0.074
	average degree	1.989	1.203	1.386	1.240	0.947	3.640
	number of ties	181	71	61	62	36	182

where a higher number of ties in each successive observation time explains its higher density, across all regions. A visualization of the networks from four hours considered for analysis from Help Island Public is shown in the Figure 6.1.

Table 6.3 shows the changes between the observations for each period, across all regions. First, across all regions the sparsity of the network means that a substantially large number of ties stay 0 (null). Furthermore, for each successive time period the number of ties remaining 1 keeps increasingly monotonically. This is due to the fact that the ties for the actors remain at their last time step value for all successive observations. Thus, the changes from 1 to 0 or 0 to 1 are more indicative of ties changes happening across each time step (which are few). It is because of this reason that

Table 6.3: Changes between Observations

Periods		Help Island Public	Help People Island	Morris	Kuula	Pondi Beach	Moose Beach
1 to 2							
	0 to 0	8090	3381	1858	2427	1383	2322
	0 to 1	39	31	11	5	12	56
	1 to 0	0	0	0	0	0	0
	1 to 1	61	10	23	18	11	72
	Distance	39	31	11	5	12	56
	Jaccard	0.610	0.244	0.676	0.783	0.478	0.562
2 to 3							
	0 to 0	8049	3366	1845	2417	1374	2300
	0 to 1	41	15	13	10	9	22
	1 to 0	0	0	0	0	0	0
	1 to 1	100	41	34	23	23	128
	Distance	41	15	13	10	9	22
	Jaccard	0.709	0.732	0.723	0.697	0.719	0.853
3 to 4							
	0 to 0	8009	3351	1831	2388	1370	2283
	0 to 1	40	15	14	29	4	17
	1 to 0	0	0	0	0	0	0
	1 to 1	141	56	47	33	32	150
	Distance	40	15	14	29	4	17
	Jaccard	0.779	0.789	0.770	0.532	0.889	0.898

the distance between the networks for the same region for successive time steps remains around the initial distance value, with only some observations for some regions showing drastic change. This is also the reason for the Jaccard coefficient being around 0.7 for each region across all time periods. Thus there appears to be a underlying trend for network changes in each of the regions.

Table 6.2 shows the rate parameter estimates for the regions across all the time periods. These correspond to the estimates of the tie changes between the periods given the observed data. There

seems to be a trend to the rates, specific to each region hovering around a particular rate, with only a few major changes.

6.2.2 Estimation Procedure

We used the Only estimation procedure implemented in Siena: the Method of Moments (MoM) [SBS10, Sni01a]), where the parameters are estimated in such a way that expected values of a vector of selected statistics are equal to their observed values (for the network in this case).

The SIENA software implements two methods for MoM estimation: conditional and unconditional. The difference between the two is in the stopping criteria for the simulations of the network evolution.

For unconditional estimation, the network evolution simulations for each time period (along with co-evolution of the behavioral dimensions, if included) continue until a predetermined time (taken to be 1.0 for each consecutive time period moments) has passed.

In conditional estimation, the simulations for each period continue to run until a stopping criterion (calculated from the observed data) is reached. It is possible to do conditioning (on the observed number of changes on this dependent variable) for each of the dependent variables (network, or behavior). The conditioning on the network variable refers to running the simulations until the difference in entries for the initially observed network of this period and the simulated

network equals the number of entries in the adjacency matrix for the difference between the initial and the final networks of this period.

6.2.2.1 Algorithm Steps

During the estimation process (using the SIENA program), the estimation algorithm has three phases [SR10]:

1. Phase 1, a rough estimate for the matrix of derivatives is made while the parameter vector is held constant at its initial value.

2. Phase 2 has a number of subphases (and hence greater precision). The default is 4 subphases. The changing parameter values from different runs reflect the deviations between generated and observed values of the statistics which become smaller in the later subphases.

   The objective is to come up with parameter values where the deviations average out to 0 - called the 'quasi-autocorrelations', which are averages of products of successively generated deviations between generated and observed statistics.

3. Phase 3 consists of estimating the covariance matrix and derivative matrix used for computing standard errors with the parameter vector again being kept constant, now at its final value. The default number of iterations for phase 3 is 1000.

The user has the option to change the number of subphases executed in step 2 or step 3 of the algorithm. Also, there is an option to fix the values to standard values (especially when experiencing problems with convergence) before the estimation process. For our purposes we used the default settings for the number of iterations in each phase and the conditional MoM, except for the gender case, where we resorted to using unconditional MoM with two fixed parameter values (for the change ratio for periods 1 and 2).

6.2.2.2 Convergence Check

A convergence check is provided based on the execution of phase 3 of the algorithm (computed from the deviations between the simulated values of the statistics and the observed values). Ideally these deviations should be as close to zero as possible for good convergence. Siena provides t-statistics computed from these averages and standard deviations. The recommendation for the t-statistics for the longitudinal analysis [SR10] is that the convergence is excellent when these values are less than 0.1 (absolute value) and good when less than 0.2 and moderate when less than 0.3. In our case the t-ratios for all estimated parameters in the model were less than 0.1 in the absolute indicating good convergence.

Table 6.4: Rate Parameter Estimates

Rate Parameter		Help Island Public	Help People Island	Morris	Kuula	Pondi Beach	Moose Beach
Period 1							
	Estimate	0.4663	0.5359	0.3076	0.1208	0.4835	0.9500
	Standard Error	0.0752	0.0993	0.0941	0.0541	0.1388	0.1265
Period 2							
	Estimate	0.5145	0.2628	0.3946	0.2518	0.4344	0.3338
	Standard Error	0.0812	0.0666	0.1117	0.0803	0.1417	0.0714
Period 3							
	Estimate	0.5178	0.2564	0.4622	0.8251	0.2139	0.2496
	Standard Error	0.0794	0.0646	0.1205	0.1468	0.1085	0.0599

6.2.2.3 Interpretation of Parameter Values

The rate parameter (called ρ in Section 6.1.1.2) for the three periods for each region is shown in the Table 6.4. It indicates the estimated number of changes per actor (i.e., changes in the choices made by this actor, as indicated in the row for this actor in the adjacency matrix) between the two observations. It is to be noted that this refers to unobserved changes, and that some of these changes may cancel (a choice can be made and then withdrawn), so the average observed number of differences per actor can be actually smaller than the estimated number of unobserved changes. The Table 6.4 shows that for each region there is a value around which the rate change hovers for at-least 2 successive time periods indicating that according to the model, there is an estimated number of changes per actor for this network that it prefers (going from 1 to 0 or 0 to 1).

We also included a outdegree (density), however as the manual for Siena [SR10] points out, no definite conclusion can be made on the basis of this value alone as all the parameters depend

Table 6.5: Out-degree Parameter Estimates

	Help People Island	Help Island Public	Morris	Kuula	Pondi Beach	Moose Beach
Out-degree	-0.0919	0.1078	-0.4148	-0.3633	-6.4260	0.0813

Table 6.6: Similarity Estimates for the Community Covariate

Effect		Help Island Public	Help People Island	Morris	Kuula	Pondi Beach	Moose Beach
Alter							
	Estimate	0.0800	0.0598	-0.1489	0.0822	-0.0891	-0.1101
	Standard Error	0.0379	0.0347	0.1230	0.0458	0.0673	0.0688
Ego							
	Estimate	0.5802	-0.2605	0.3338	-0.3965	-0.3563	-0.2788
	Standard Error	4.7544	0.1483	3.1013	0.1490	0.1571	0.1474
Similarity							
	Estimate	1.9342	0.7418	-3.0664	1.4151	-1.9858	-2.3192
	Standard Error	0.6274	0.8156	1.8444	1.1918	1.5672	1.6072

on this parameter. The value estimates for the parameter are as shown in Table 6.5. A -ve value means that the actors prefer others with opposite degrees (higher or lower) and magnitude tells how pronounced the effect is. Table 6.5 shows that Help Island Public has a small negative value which implies newbies talking to more experienced users, whereas the Help Island Public has a small positive value meaning the opposite (newbies talking more among themselves). Both the entertainment type regions (Morris and Kuula) have a negative value meaning that users there prefer more users with opposing degrees according to the model.

To investigate the premise that the communities from previous days persist over time and are indicative of the future interactions, we explored the effect of Community covariate for all the re-

gions. The values for the Ego, Alter and Similarity for the community actor covariate as described in Section 6.1.1.1 are presented in Table 6.6. These are the weights in the evaluation function described in detail in Section 6.1.2. A positive value of Similarity indicates that for the covariate the actors are more likely to make connection to other actors of the same value of the covariate as them, whereas a negative value indicates otherwise (the same goes for the Ego and Alter). The following general comments can be made from the values for Ego, Alter and Similarity for the community covariate in Table 6.6:

- A high positive value of Similarity for community means that more actors are likely to connect to other actors that have the same value of community membership (and vice versa for the negative Similarity value).
- A slight positive value of Alter means that actors are more likely to converse to other actors with different (higher) value of the community covariate (and vice versa for the negative Alter value).
- A high positive value of Ego also means that actors are more likely to accept other actors with different (higher) value of the community covariate (and vice versa for the negative Ego value).

Table 6.6 shows that both the Help islands and one sandbox region (Kuula) have a positive value for the Similarity indicating that people from the same communities prefer talking to others from the same community, whereas the two beaches and one sandbox region (Morris) have a negative value of Similarity indicating otherwise. Further, the Alter effect mirrors the Similarity (for the

Table 6.7: Similarity Estimates for the Different (Constant) Covariates

Parameter	Community	Gender	Age
Ego	0.5802 (4.7544)	-0.0350 (0.5568)	0.0379 (0.6681)
Alter	0.0800 (0.0379)	-0.0350 (0.5568)	-0.7767 (0.5726)
Similarity	1.9342 (0.6274)	0.4057 (0.5131)	-1.1280 (3.3675)

sign of the estimate value) meaning the actors with a low (for negative) or high (for positive) value of the covariate are more likely to increase their in-degrees. Lastly, the Ego is positive for one of the Help islands and one sandbox region (Morris) but negative for all other regions indicating the preference for accepting incoming connections from other actors similar to it.

To further unravel the effect of community covariate on the community evolution vis-a-vis other possible effects, we explored two other constant actor covariates in our model for one of the Help Islands (Help Island Public):

1. Gender

2. Age (number of days since the SL avatar has been created))

The values for the Ego, Alter and Similarity for the three actor covariates as presented in Table 6.7. These are the weights in the evaluation function described in detail in Section 6.1.2. The following can be concluded from the values for Similarity in Table 6.7

- A high positive value of Similarity for community means that more actors are likely to connect to other actors that have the same value of community membership.

- A slight positive value of Similarity for the gender means that actors are more likely to talk to other people that are of the same gender.

- A negative value of Similarity for age means that actors are more likely to communicate to other actors that are different from their own age group.

Since the value of Ego and Alter parameters for the Gender is near zero, the Similarity affect is pronounced, however for Age and Community membership this is not the case and it affects the linkages as discussed in Section 6.2.3. Briefly speaking, a positive value for the Ego means the that actors with higher values on this covariate are more likely to increase their out-degrees rapidly. Similarly for the Alter a positive parameter value implies that the actors with higher values on this covariate are more likely to increase their in-degrees rapidly. The negative Alter values for Age forces the connections to lower values of the covariate variable whereas the positive Alter values for Community forces the connections to higher values of the covariate variables. Similar affect is enforced by positive value of Ego for Community and Age (both being positive), favoring the actors with higher age/community values.

6.2.3 Model Estimates for the Community Covariate

In Section 6.2.2.3 we discussed the values for the Similarity, Ego and Alter covariates given in Table 6.7 for the three covariates (age, gender and community) and their effect on the tendency of the actors to form links for Help Island Public. Here we dissect the effect further, to make the

Table 6.8: Network Dynamics Model Contribution from Ego, Alter and Similarity for Community Covariate

v_i/v_j	0	1	2	3	4	5	6	8	9	13
0	-0.85	-0.77	-0.69	-0.61	-0.53	-0.45	-0.37	-0.21	-0.13	0.19
1	-0.27	-0.19	-0.11	-0.03	0.05	0.13	0.21	0.37	0.45	0.77
2	0.31	0.39	0.47	0.55	0.63	0.71	0.79	0.95	1.03	1.35
3	0.89	0.97	1.05	1.13	1.21	1.29	1.37	1.53	1.61	1.93
4	1.47	1.55	1.63	1.71	1.79	1.87	1.95	2.11	2.19	2.51
5	2.05	2.13	2.21	2.29	2.37	2.45	2.53	2.69	2.77	3.09
6	2.63	2.71	2.79	2.87	2.95	3.03	3.11	3.27	3.35	3.67
8	3.79	3.87	3.95	4.03	4.11	4.19	4.27	4.43	4.51	4.83
9	4.37	4.45	4.53	4.61	4.69	4.77	4.85	5.01	5.09	5.41
13	6.69	6.77	6.85	6.93	7.01	7.09	7.17	7.33	7.41	7.73

discussion more concrete, while solely focusing on the Community covariate (the discussion for the other two covariates is similar and is omitted for brevity).

The Community covariate has a range of 0-13 (values 10 and 11 were not used as they represent structural zeros and ones respectively), with an average value $\bar{v} = 1.857$ and average dyadic Similarity $\hat{sim}^v = 0.8037$. Substituting these values into the actor-oriented model (described in Section 6.1.2, Equation 6.18), yields Equation 6.19. Table 6.8 gives the values from the equation for each value of v_i, v_j for the covariate.

$$- 0.38(v_i - \bar{v}) - 0.12(v_j - \bar{v}) + 3.81(1 - \frac{|v_i - v_j|}{\Delta_V} - 0.8037) \tag{6.19}$$

Table 6.8 indicates that the highest values for each row are along the last column. The last column encodes the actors that are from the community that comprises the actors that were not present in the three days data that was considered for the community labeling. The high value

of the Similarity indicates a preference for the actors that have the same community membership, similarly a positive Alter value of the actor favors the actors that have a higher value for community and similarly the higher membership actors are favored by the positive value of the Ego (from Table 6.7). The end result as shown is that for all the row values the actors end up favoring the tie with the actor with highest value of the community membership. This agrees with the intuition that most changes in the network are likely to happen from a actor initiating communication with this new user group (the encoded community comprising of users present within that hour only).

6.3 Summary

In this chapter, we presented the results from the longitudinal analysis of the network data for the different types of regions. The main impetus for doing the longitudinal analysis was to be able to make statements about the network properties, from each region, as observed over time, on a statistical footing using a probabilistic framework. The longitudinal analysis is the recommended approach [SR10] for the small networks, as in our case, where the network size is typically less than 30. An alternate approach, with larger network data, would be to use the Exponential Random Graph model [FS86, SPR06, WP96], which we attempted for one days worth of data, as mentioned in Section 4.4. Both the longitudinal analysis and the Exponential Random Graph models allow us to make statements about the network properties on a statistical basis, based on the Markov-chain simulation of the network. Thus, our foremost objective in this chapter was to put-forth the

network properties, that can be generalized from the simultaneous analysis of this aggregated data. There has been previous work done in this area; for example, [HSW09] uses ERGM to analyze the network of economic exchanges in EverQuest II. Our study is the first of its kind for the virtual world of Second Life, which differs from the other MMO's due to the lack of enforcement of user goals and quests, where the user must assume a role and accomplish a 'mission'.

Secondly, we were looking for properties that can be generalized across the region types for the regions sampled (orientation islands, entertainment areas, and sandboxes). However, the results, show that no definite conclusion can be made on the basis of the estimated parameters for the rate and the covariates. If we look at the rate parameters in Table 6.4 for the three time periods there are no distinctive values that can be generalized across the different types of regions, although it does show a likeness individually for a particular value. The values for the community covariate were analyzed for all the regions in Table 6.6. Again, we see no definitive trends from these for the Ego, Alter and Similarity parameter estimates, but there are some similarities across regions. For example, both the beaches have negative estimates for all the parameters, similarly for the Help islands, they have all positive estimates save for the Ego value for one of the islands. However, for the sandboxes, the values alternate between a positive and a negative for each estimate. If we look at the magnitude of the parameter values themselves, each of the regions has an Alter value of about 0.1, Ego value of 0.3 and a Similarity estimate of 2 for most of the regions in the absolute.

Third, we decided to further explore the underlying preferences of the actors by examining other covariates, that is, age and gender. Table 6.7 shows that the Ego and Alter have a small negative value for the gender, while the Similarity is high positive, meaning that the effect of Simi-

larity is more pronounced and actors are more likely to connect to other actors that are of the same gender. Alter has small positive Ego value and high negative values for the Alter and Similarity, meaning that actors prefer others with different age group (negative Similarity) and a positive Alter implies that the actors with lower values on this covariate are more likely to increase their in-degrees rapidly. This indicates inter-relationship of the gender and age covariates in addition to the community covariate on the link formation for one of the Help islands.

Table 6.8 summarizes the effects of Ego, Alter, and Similarity for the community covariate. All the rows have values increasing monotonically from left to right, as values for the Ego, Alter and Similarity for community covariate were all positive, favoring higher covariates. The last column in each row appropriately contains the highest value.

Thus, we can conclude that while there appears to be some underlying pattern and some similarities to the various network properties for different types of regions, there isn't a single distinguishing feature that can separate the different types of regions. The findings presented here provide insight into the various factors affecting the network evolution on the basis of mined networks from social-interaction data and the membership of the extracted communities.

CHAPTER 7

TOPIC MODELING

Thus far we have relied on the information mined from the network itself to improve the link prediction algorithm. The logical next step was to explore the use of the content in the dialogs to see if we can improve upon the link mining algorithm, by using the higher level semantic information present in the dialogs. To accomplish this, we opted to perform topic modeling on the dialog data. Topic modeling provides a higher level semantic interpretation of the text than simple n-gram frequency analysis, which relies on the frequency of word tokens in the documents. There have been different techniques used in topic modeling for conventional document-based corpus [MS99] and recently there has been growing interest in using them for social media such as Twitter [RDL10]. We present an overview of these techniques here.

7.1 Probabilistic Latent Semantic Analysis (pLSA)

Probabilistic latent semantic analysis (pLSA), also known as probabilistic latent semantic indexing (pLSI) [Hof99], adds a probabilistic model to LSA by introducing latent classes (topics) using a (multinomial) mixture decomposition, which gives better statistical properties in terms of objec-

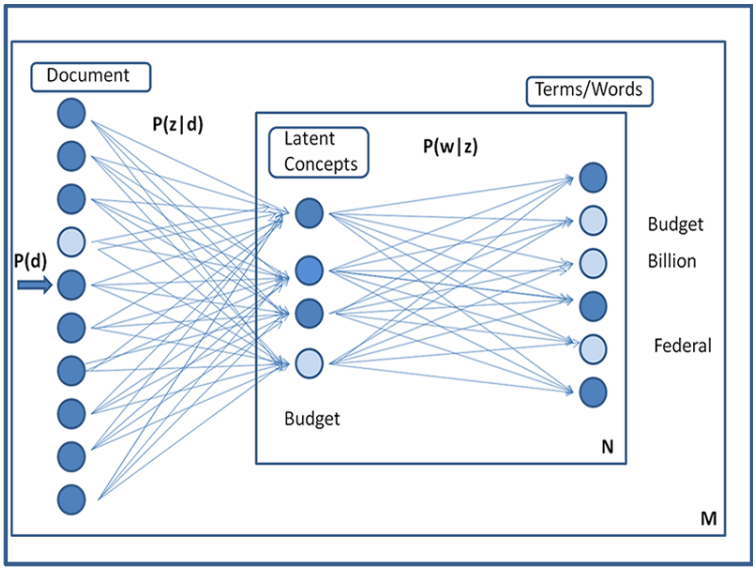


Figure 7.1: Aspect model for pLSA

tive function, model interpretation and error minimization. The likelihood function for pLSA is derived from multinomial sampling and includes explicit maximization of the predictive power of the model.

If we write the word-document co-occurrences as (w, d), the probability of co-occurrence is given by pLSA as a mixture of conditionally independent multinomial distributions as shown in the Figure 7.1. The aspect model associates an unobserved (latent) class variable $(c = c_1, c_2, ..., c_k)$ with each of the observations, such that the joint probability model over the $D \times W$ is given by Equation 7.1 called the 'aspect model' (where c acts as a bottleneck variable in predicting the words with cardinality less than the documents d or the words w).

$$P(w,d) \quad = \quad \sum_c P(c)P(d|c)P(w|c) = P(d)\sum_c P(c|d)P(w|c) \tag{7.1}$$

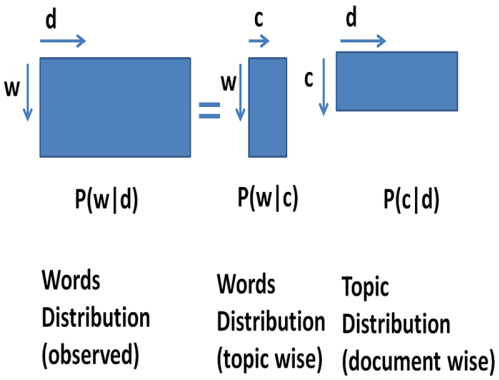


Figure 7.2: Matrix form for pLSA

The first part in Equation 7.1 gives the symmetric formulation in which words w and documents d both are generated from latent class c using conditional probabilities, ($P(d|c)$ and $P(w|c)$) . The second (asymmetric) formulation chooses a latent class conditioned on the document ($P(c|d)$), and then generates a word from that class according to $P(w|c)$. The second formulation can also be written in matrix form as shown in Figure 7.2.

The class-conditional multinomial distributions over the vocabulary (factors) can be assigned to a $M - 1$ dimensional simplex R of all possible multinomials of M components. Each topic c defines a point on the simplex R, given by the multinomial distribution $P(w|c)$. These (k) topics define k-1 points that form a k-1 dimension simplex. The assumption is that $P(w|d)$ is defined by a convex combination(s) of $P(w|c)$ with $P(z|d)$ as factors.

pLSA has still been found to suffer from deficiencies. The aspect model used is reported to suffer from severe overfitting problems [BNJ03]. The number of parameters increase linearly with the documents. More importantly, pLSA might be a generative model of the documents in the collection but not for new (unseen) documents.

7.2 Latent Dirichlet Allocation (LDA)

Proposed by Blei et al. [BNJ03], Latent Dirichlet allocation (LDA) belongs to a generative family of probabilistic models, which assumes that items (documents) in a corpus are formed a finite mixture of topic probabilities, such that each word's creation is attributable to one of the document's topics. The use of a Dirichlet distribution as as a prior for the topic distribution of the document is what distinguishes LDA from pLSA.

Hofmann's pLSI is also incomplete in the sense that it presents no probabilistic model at the level of documents [BNJ03]. The representation for each document in terms of numbers (representing the mixing proportions of the topics) carries no explanation in terms of a generative probabilistic model. This means that there is a linear growth in number of parameters for the model with the corpus, leading to overfitting problems. Furthermore, this also means that assigning of probabilities to documents not included in the training set is also a source of problem.

We first cover the specifics of the original model as proposed in Blei et al. [BNJ03]. Formally,

- A *word* is considered to be the basic unit of the discrete data and is defined as an item from the vocabulary, indexed as $1, ..., V$. Thus, the words are represented as unit-basis vectors (having one component being equal to one and the others zero), meaning the vth word in the vocabulary is represented as a V-vector w such that $w^v = 1$ and $w^u = 0$ for $u \neq v$.

- A *document* is considered as a sequence of N words denoted by $w = (w_1, w_2, ..., w_N))$, such that w_n represents the nth word in the sequence.

- A *corpus* is defined as the collection of M documents such that $D = w_1, w_2, ..., w_M$

The generative process LDA assumes for each document w in a corpus D is then given by [BNJ03]

1. Choose $N \sim Poisson(\xi)$

2. Choose $\theta \sim Dir(\alpha)$

3. For each of the N words w_n

   (a) Choose a topic $z_n \sim Multinomial(\theta)$

   (b) Choose a word w_n from $p(w_n|z_n, \beta)$, a multinomial probability conditioned on the topic z_n

Here, the dimensionality k of the Dirichlet distribution (and hence the dimensionality of the topic variable z) is considered to be known and fixed. Further, the word probabilities, parameterized by $k \times V$ matrix β (such that $\beta_{ij} = p(w^j = 1|z^i = 1)$), is also taken to be a fixed quantity (to be estimated). It is to be noted that these are simplifying assumptions for the basic model to make it more amenable to understanding, and the complete model removes these assumptions [BNJ03].

The k-dimensional Dirichlet random variable θ can assume the values in the $(k-1)$-simplex (from the fact that a k-vector θ lies in the $(k-1)$-simplex if $\theta_i \geq 0, \sum_{i=1}^{k} \theta_i = 1$), such that the probability simplex is defined as:

$$p(\theta|\alpha) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\Pi_{i=1}^{k} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} ... \theta_k^{\alpha_k - 1} \tag{7.2}$$

In Equation 7.2 parameter α is a k-vector with elements $\alpha_i > 0$, and $\Gamma(x)$ is the Gamma function. The Dirichlet distribution provides a more convenient distribution on the simplex since it belongs to the exponential family of distributions, has finite dimensional sufficient statistics and is conjugate to the multinomial distribution.

For the parameters α and β, the joint distribution for the topic mixture θ, set of N topics z, and a set of N words w can be written as in Equation 7.3,

$$p(\theta, z, w|\alpha, \beta) = p(\theta|\alpha)\Pi_{n=1}^{N} p(z_n|\theta)p(w_n|z_n, \beta) \tag{7.3}$$

where $p(z_n|\theta)$ is θ_i for each i so that $z_n^i = 1$. Integrating over θ and summing over z gives the marginal distribution of the document as shown in Equation 7.4

$$p(w|\alpha, \beta) = \int p(\theta|\alpha)(\prod_{n=1}^{N} \sum_{z_n} p(z_n|\theta)p(w_n|z_n, \beta))d\theta \tag{7.4}$$

Taking the product of the marginal probabilities of single documents, the probability of the corpus can be written as

$$p(D|\alpha, \beta) = \prod_{d=1}^{M} \int p(\theta_d|\alpha)(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn}|\theta_d)p(w_{dn}|z_{dn}, \beta))d\theta_d \tag{7.5}$$

The probabilistic graphical model of LDA comprises three levels in terms of representation: parameters α and β are the corpus level parameters (sampled once in the process of generating the corpus), whereas variables θ are the document-level variables (sampled once per document) and

z_{dn} and w_{dn} the word level variables (sampled once for each word in the document). This means that the topic node is sampled repeatedly within the document and a document can be assigned to multiple topics.

$$p(w, z) = \int p(\theta)(\prod_{n=1}^{N} p(z_n|\theta)p(w_n|z_n))d\theta \tag{7.6}$$

LDA assumes words to be generated from topics (as defined by fixed conditional distributions) and assumes that those topics are infinitely exchangeable within a document. The probability of a sequence of words and topics can be given by de Finetti's theorem as given in Equation 7.6, where θ is defined as the random parameter of a multinomial over topics. Marginalizing the topic variables and endowing θ with a Dirichlet distribution in Equation 7.4 then gives the LDA distribution on documents.

A two-level interpretation of LDA [BNJ03] can be obtained by marginalizing over the hidden topic variable z, such that the word distribution is given by the Equation 7.7:

$$p(w|\theta, \beta) = \sum_{z} p(w|z, \beta)p(z|\theta) \tag{7.7}$$

The generative process for the document w is then given by [BNJ03]

- Choose $\theta \sim Dir(\alpha)$
- For each word w_n in N:

  (a) A word w_n is chosen from $p(w_n|\theta, \beta)$

This gives the marginal distribution of the document in terms of a continuous mixture distribution given by Equation 7.8, where $p(w_n|\alpha, \beta)$ are mixture components and $p(\theta, \alpha)$ the mixture weights.

$$p(w|\alpha, \beta) = \int p(\theta, \alpha)(\prod_{n=1}^{N} p(w_n|\theta, \beta))d\theta \tag{7.8}$$

7.2.1 Author-Topic LDA Model

Griffiths, Steyvers, Smyth and Rosen-Zvi [SSR04a, SSR04b, RCG10] extended the LDA topic model to include authorship information to address the problem of authorship attribution, stylometry, and forensic linguistics. The model can be used both to discover the document topic as well as the associated authors. A multinomial distribution over topics is assumed for each author, with multi-author documents having a mixture of such distributions. During document generation, an author is randomly chosen for every word in the document. The chosen author then decides the topic (from the multinomial over topics), and subsequently a word is sampled from the distribution of words over that topic, with the process iterating over all the words in the document.

The authors generate the words from T topics. In cases where T is small relative to the authors and vocabulary size, the author-topic model provides dimensionality reduction to the documents (number of topics being much smaller than the number of authors and vocabulary size).

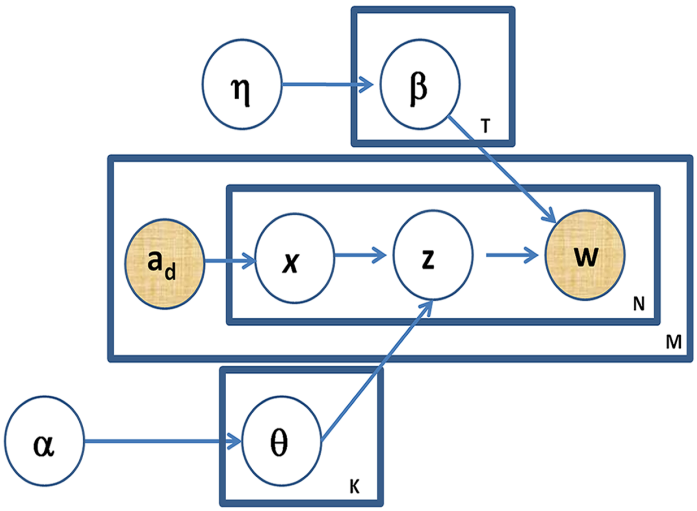


Figure 7.3: The Author-Topic LDA model by Steyvers, Griffiths and Rosen-Zvi.

A graphical representation for this topic model is as shown in Figure 7.3 β is used as the multinomial distribution over words for a topic by a $k \times V$ matrix. The plate surrounding the β shows the repeated sampling of the word distributions for each topic z until T topics are generated. α is the symmetric Dirichlet prior over the multinomial over the topic (θ). An author x is sampled uniformly over A_d for each word in the document, leading to sampling of a topic z from the multinomial θ associated with author x and then to sampling of a word from the multinomial β related to topic z, with the process repeated N times to form the document d.

7.2.2 Extracting Author-Topics using Author-Topic LDA

The main parameters to be estimated are T topic-word distributions ϕ and the K author-topic distributions θ for each of the document. The corresponding latent variables for the assignment of each word to topics z and author x are also to be estimated. [GS04] have suggested using Gibbs Sampling (a variant of Markov Chain Monte Carlo) to sample over the posterior distribution over parameters, evaluating on only x and z and then using the results to compute θ and β.

The topics and authors are sampled according to Equation 7.9 for every word, where $z_i = j$ and $x_i = k$ respectively denote the assigning of *i*th word for a document to topic j and author k. $w_i = m$ denotes that *m*th word in lexicon corresponds to *i*th word whereas z_{-i} denotes the topic assignment and x_{-i} the author assignments not including *i*th word. C^{WT}_{mj} represents the count of times word m is assigned to topic j save for the current instance, and C^{AT}_{kj} is count of the times author k gets assigned to topic j, again, save for the current instance and V denotes lexicon size.

$$P(z_i = j, x_i = k | w_i = m, z_{-i}, x_{-i}) \propto \frac{(C^{WT}_{mj} + \beta)(C^{AT}_{kj} + \alpha)}{(\sum_{\acute{m}} C^{WT}_{\acute{m}j} + V\beta)(\sum_{\acute{j}} C^{AT}_{k\acute{j}} + T\alpha)} \tag{7.9}$$

The algorithm maintains track of the $V \times T$ (word by topic) and $K \times T$ (author by topic) count matrices, which can then be used to compute the topic-word distributions ϕ and author-topic distributions θ as shown in Equation 7.10 and Equation 7.11, where ϕ_{mj} denotes the probability of assigning word *m* in topic *j* and θ_{kj} the probability of use of topic *j* by author *k*, both corresponding to the predictive distributions over new words *w* and new topics *z* conditioned on *w* and *z*.

$$\acute{\phi}_{mj} = \frac{C_{ij}^{wt} + \beta}{\sum_{k=1}^{W} C_{kj}^{WT} + W\beta} \tag{7.10}$$

$$\acute{\theta}_{kj} = \frac{C_{dj}^{AT} + \alpha}{\sum_{k=1}^{W} C_{dk}^{AT} + T\alpha} \tag{7.11}$$

The algorithm performs similarly to LDA for document-topic modeling, with assignment of words to randomly chosen authors (from the set of authors for the document) and topics, proceeding to Gibbs sampling, each word in the corpus, iterated over *I* times, which can then be averaged over multiple linked-chains.

7.3 Improving Link Mining

In this section, we describe how we were able to use topic modeling to refine the link predictions algorithms introduced in Chapter 3. We evaluated three variations for topic modeling in Second Life, differing on how the dialogs from the users are grouped as documents (as shown in Figure 7.4):

1. The first approach was to group the dialogs from each user for each region into a document of its own.

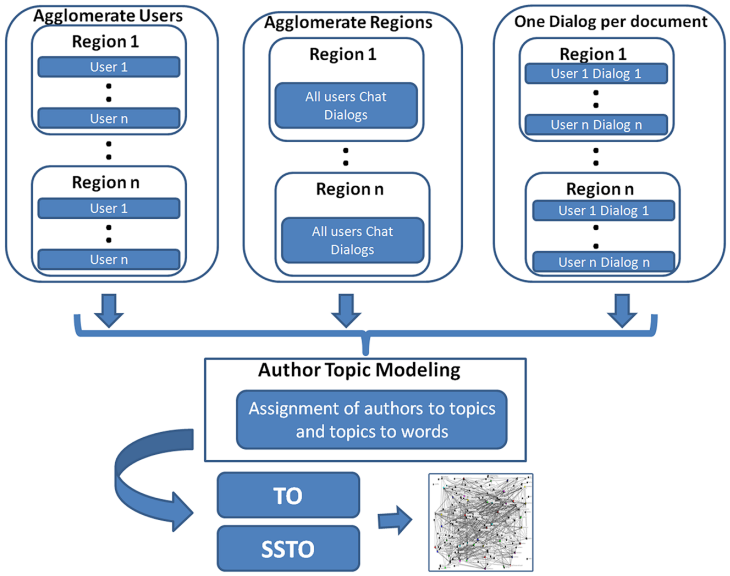


Figure 7.4: Using topic modeling to improve link mining.

2. The second approach was to agglomerate the dialogs from each region (for all users) into a document of its own.

3. The final approach was to breakdown each of the utterances into a separate document (one document per dialog).

For link refinement, we opted to use the author topic model; for this model, the third document option (one document per dialog) yielded the best results. This yields the topics, and the assignment for each author for the topics. This probability assignment was then used to compute the similarity of the authors using relative entropy or Kullback-Leibler (KL) divergence [KL51]. This measure has been previously used for computing similarity between documents by Steyvers and Griffiths [GS07]. This list was then sorted on the basis of similarity or lowest KL divergence

score. We proceed according to following for the Temporal Overlap (TO) and Shallow Semantic Temporal Overlap (SSTO) algorithms:

- For the Temporal Overlap (TO), a value was selected from the KL divergence scores, and all the links between the users with a similarity score below the value were eliminated. Furthermore, we chose the variation of TO that incorporates community information from the same hour. There are two ways link elimination was accomplished:
  - In one scheme we incorporated the KL divergence as a fall back to the TO+HT method, meaning that if there is no community information available as a prune off we use the KL divergence similarity list as a prune off list. We call this variant TO+HT+KL (fall back)..
  - For the second scheme, we used the KL divergence similarity list in conjunction with the community information, meaning that the connection is made not only if the user is from the same community but also is in the KL divergence prune off list as well. This variant is called TO+HT+KL (community ANDed with KL divergence).
- For Shallow Semantic Temporal Overlap (SSTO), we evaluated two variations
  - one in which we use the most similar user from the sorted list of KL divergence distances, as a post-processing step to the SSTO and SSTO plus Loose Community (SSTO+LC) algorithms. Once the algorithms are finished labeling the data, we make another pass through the dialogs and label the unlabeled dialogs with the user that is

most similar to the user who uttered the dialog that went un-labeled. We call these SSTO+KL and SSTO+LC+KL respectively.

- In the second variation, we incorporated the most similar user as a fall back heuristic to the link mining algorithm for SSTO+LC itself, making it part of the algorithm. We use SSTO+LC+KL (fall back) to denote this algorithm.

It is to be noted that KL divergence is an asymmetric measure of similarity and was symmetrized by averaging the measure in both directions.

To enable comparison we used the same dataset that was used in Section 7, consisting of 5 randomly-chosen days of data logs, separated into hourly partitions. The same hourly partitioned data was selected for each day, used to generate user graph adjacency matrices using the modifications of the TO, SSTO and SSTO+LC algorithms described earlier. The parameter values for the hyper-parameters were chosen to be α as 0.02, β as 0.0001, number of iterations 2000, number of Markov chains 10 and number of topics, T as 200. The procedure for choosing the parameters is explained in detail in Section 8.6.2.

7.4 Results

In this section we summarize the results from a comparison of the social networks constructed from the modification of the algorithms presented in Section 5.2. Again we adopt the scheme similar to Section 5.2, where we restricted our attention to comparing networks as a whole in terms of the link

difference (using Frobenius norm) for both the symmetric (TO and its variants) and asymmetric networks (SSTO and its variants) and a one-to-one comparison for the *to* and *from* labelings for each dialog on the ground-truthed subset (using precision and recall) for the asymmetric networks.

7.4.1 Network Comparison using the Frobenius Norm

Again, similar to Section 5.2 we used a gold-standard subset of the data by hand-annotating the to/from fields for a randomly-selected hour from each of the Second Life regions. To compare the generated networks against this baseline, we use two approaches. First we compute the Frobenius norm [GL96] of the difference of the adjacency matrices from the corresponding networks against the hand-labeled network.

7.4.2 Direct Label Comparisons

Following the approach in Section 5.3.2 we present a head-to-head comparison of the to/from labelings for the dialogs using the approaches described above (for SSTO variants) against the hand annotated dialogs. This yields the true positives and false positives for the approaches and allows us to see which one is performing better on the dataset, and if there is an effect in different Second Life regions. Table 7.2 shows the results from this analysis.

Table 7.1: Frobenius norm: comparison against hand-annotated subset.

	SSTO+KL (post processing)	SSTO+LC+KL (post processing)	SSTO+LC+KL (fall back)	TO+HT+KL (fall back)	TO+HT+KL (community ANDed with KL divg.)
Help Island Public	43.20	40.60	41.26	130.08	77.90
Help People Island	66.80	67.06	60.93	54.88	76.01
Kuula	38.68	34.12	30.27	77.82	41.20
Mauve	50.23	50.01	41.54	49.89	54.53
Moose Beach	19.16	18.47	18.71	50.97	21.58
Morris	30.36	25.90	19.92	38.98	30.56
Pondi Beach	25.87	22.58	17.89	71.02	28.95
Rezz Me	40.04	38.78	39.15	41.10	43.90
Total error	314.33	298.5	**259.62**	514.74	**374.65**

7.4.3 Summary

For the temporal overlap plus daily community algorithm (TO+HT) variants, the addition of the similarity information from KL divergence reduces the link noise only when used in conjunction with the community information (logical AND), whereas the fall-back approach gives the same value as TO+HT alone, since there is no missing community information for our data. This is shown by the decreased value of the Frobenius norm for TO+HT+KL (community ANDed with KL divergence) that is substantially lower than any of the variants for the TO algorithm as presented in Table 5.1. For the SSTO variants, SSTO+KL and SSTO+LC+KL, where the KL divergence was used as a post-processing step to the original algorithms themselves, topic modeling fails to give any improvement over the original algorithms, in-fact resulting in a higher Frobenius norm than

Table 7.2: Precision/Recall values for one-to-one labeling comparison.

		Help Island Public	Help People Island	Mauve	Morris	Kuula	Pondi Beach	Moose Beach	Rezz Me
Total Dialogs		360	184	128	179	227	144	128	97
SSTO +KL (post proce- ssing)	match	77	99	52	47	66	30	22	27
	precision	0.2139	0.5380	0.4063	0.2626	0.2907	0.2083	0.1719	0.2784
	recall	0.3407	0.5928	0.4298	0.3760	0.3474	0.3000	0.2529	0.3140
	F-Score	0.2628	0.5641	0.4177	0.2334	0.3165	0.2459	0.2047	0.2951
	total	360	184	128	179	227	144	128	97
SSTO+ LC+KL (post proce- ssing)	match	61	79	37	39	52	26	12	15
	precision	0.1694	0.4293	0.2891	0.2179	0.2291	0.1806	0.0938	0.1546
	recall	0.2699	0.4731	0.3058	0.3120	0.2737	0.2600	0.1379	0.1744
	F-Score	0.2082	0.4501	0.2972	0.2566	0.2494	0.2131	0.1116	0.1639
	total	360	184	128	179	227	144	128	97
SSTO+ LC+KL (fall back)	match	44	12	8	8	18	13	7	9
	precision	0.3964	0.3333	0.5000	0.2424	0.3273	0.3939	0.2500	0.6923
	recall	0.1947	0.0719	0.0661	0.0640	0.0947	0.1300	0.0805	0.1047
	F-Score	0.2611	0.1182	0.1168	0.2855	0.1469	0.1955	0.1217	0.1818
	total	111	36	16	33	55	33	28	13

the original algorithms themselves (Table 5.1). However, the SSTO+LC+KL with fall-back offers improvement over all the previous variants of SSTO in terms of the Frobenius norm.

To dissect the improvement further, Table 7.2 shows how the dialog labeling generated from various algorithms agrees with the gold standard notations produced by a human labeler. Since TO only produces undirected links, we do not include it in the comparison. A comparison with Table 5.2 shows that the precision and recall values are worse than in SSTO alone (Table 5.2) for all the cases.

Thus, overall we see that KL divergence author similarity information can be used as a cue to prune off the links for the TO to great effect. However, the additional information fails to produce substantiative improvements for the SSTO variants, when compared for both Frobenius norm and precision and recall. This means that semantic information present in the dialogs can be used in addition to the network only information as an additional cue to guide the algorithm for TO, but since SSTO already relies on a more specific semantic cues, the topic modeling fails to offer any improvements on top of that.

CHAPTER 8

INFERRING THE CONTEXT OF COMMUNICATION

In this section, we describe our method for identifying the users' regional groups. Table 8.1 contains a description of the Second Life region categories that we used for this task. Our goal is to identify each user's region-based group based on a combination of network, community, and conversational content features. Our study includes three basic types of regions: 1) orientation areas, 2) sandboxes for scripting and building, and 3) scenic areas. Many regions include multiple attraction types, but we categorized each region by the dominant attraction near the bot's position.

Given the network obtained using the SSTO algorithm (described in Section 3.2.2), we can perform higher level analysis on this network, to extract the network and community features used in the classification task. The output of SSTO is a to/from labeling for the chat dialogs with directed links between users. These social networks were used as the basis for both the network

Table 8.1: Second Life region descriptions

Region	**Region Description**
Help Island Public	Orientation area
Help People Island	Orientation area
Mauve	Sandbox
Kuula	Sandbox
Moose Beach	Scenic area
Pondi Beach	Scenic area

and community features used to train a set of supervised classifiers. The procedure is described in detail in the following sections.

8.1 Network Features

For the network features, we calculate measures of centrality for each node in the network using the UCINET software [BEF02]. The centrality measures of nodes in the network correspond to how well-connected nodes are to other nodes (and therefore more central or influential). We use three measures of centrality, degree, closeness and betweenness (unnormalized), as the set of network features [Lin79].

8.1.1 Degree

The degree measure of centrality indicates the strength of relationship for the actor in a network. The higher the value, the greater amount of communication exists between the actor and other participants in the network. This may indicate how advantaged an actor is, since possessing more links enables an actor to satisfy their needs and be more independent of other actors. They may act as point of linkage between two networks or between two other actors. The degree of a individual node is given by the number of other nodes that node is connected to.

There is a subtle difference in degree based on the symmetry of the graph. In a symmetric graph, the number of vertices adjacent to a given vertex is the degree of that vertex. For non-symmetric data the in-degree of a vertex u is the number of ties received by u and the out-degree is the number of ties initiated by u. In addition if the data is valued then the degrees (in and out) will consist of the sums of the values of the ties. For a given binary network with vertices $v_1....v_n$ and maximum degree centrality c_{max}, the network degree centralization measure is $\sum(c_{max} - c(v_i))$ divided by the maximum value possible, where $c(v_i)$ is the degree centrality of vertex v_i [Lin79]. We use the un-normalized individual degree centrality measure for our non-symmetric valued network data.

8.1.2 Closeness

The closeness measure gives the distance of the actor to all other actors in the network and thus unlike degree, gives a metric for comparing indirectly connected actors. This metric can be used to identify actors in a local neighborhood. UCINET calculates the closeness centrality of a vertex as the reciprocal of the sum of the lengths of the geodesics to every other vertex. The normalized closeness centrality of a vertex is the closeness divided by the maximum possible closeness in the graph expressed as a percentage. As an alternative to taking the reciprocal after the summation, the reciprocals can be calculated before. In this case, the closeness is the sum of the reciprocated distances such that infinite distances contribute a value of zero. This can also be normalized by

dividing by the maximum value. For a given network with vertices $v_1....v_n$ and maximum closeness centrality c_{max}, the network closeness centralization measure is $\sum(c_{max} - c(v_i))$ divided by the maximum value possible, where $c(v_i)$ is the closeness centrality of vertex v_i [Lin79]. We use the un-normalized individual closeness centrality measure for our analysis.

8.1.3 Betweenness

Betweenness, in the context of binary data gives a view of the actor based on its presence on the geodesic paths between pairs of other actors in the network. This implies that the more people the actor ends up being getting connected the greater its betweenness. Moreover, the value is discounted for all the non-critical links, that is, if two actors have one or more geodesic path, other than the one involving the actor in question, it ends up reducing the betweenness value for that actor.

The UCINET algorithm for calculating betweenness is as follows: if b_{jk} is the proportion of all geodesics linking vertex j and vertex k which pass through vertex i, the betweenness of vertex i is the sum of all b_{jk} where i, j and k are distinct. The normalized betweenness centrality is the betweenness divided by the maximum possible betweenness expressed as a percentage. For a given network with vertices $v_1....v_n$ and maximum betweenness centrality c_{max}, the network betweenness centralization measure is $\sum(c_{max} - c(v_i))$ divided by the maximum value possible,

where $c(v_i)$ is the betweenness centrality of vertex v_i [Lin79]. We use the un-normalized individual betweenness centrality measure for our analysis.

8.2 Community Features

Based on previous work [TL09], we hypothesize that community features could be valuable for our user classification problem. Community membership has been successfully used to identify latent dimensions in social networks using techniques such as eigenvector-based modularity optimization [New06b] which allows for both complete and partial memberships. We use the approach proposed by Newman [New06b], where modularity (denoted by Q below) measures the chance of seeing a node in the network versus its occurrence being completely random. It can be defined as the sum of the random chance $A_{ij} - \frac{k_i k_j}{2m}$ (where A_{ij} is the entry from adjacency matrix and $m = \frac{1}{2}\sum_i k_i$ the total edges in the network) summed over all pairs of vertices i, j that fall in the same group, where s_i equals 1 if the two vertices fall in the same group and -1 otherwise:

$$Q = \frac{1}{4m} \sum (A_{ij} - \frac{k_i k_j}{2m}) s_i s_j. \tag{8.1}$$

If B is defined as the modularity matrix given by $A_{ij} - \frac{k_i k_j}{2m}$, which is a real symmetric matrix and s column vectors whose elements are s_i then Equation 8.1 can be written as $Q = \frac{1}{4m} \sum_{i=1}^{n} (u_I^T s)^2 \beta_i$, where β_i is the eigenvalue of B corresponding to the eigenvector u. u_i are the normalized eigenvectors of B such that $s = \sum_i^n a_i u_i$ and $a_i = u_i^T s$.

We use the leading eigenvector approach to modularity optimization to perform a loose community partitioning (s being a real number). For the maximum positive contribution to the modularity we use all the eigenvectors that make positive contribution to the modularity (obtained through cross-validation) and take the coefficients for them as the membership information. We use a membership threshold of 0.1 to determine whether the contribution is sufficient for community membership. The final feature used for classification is simply the number of communities that the actor belongs to, instead of the membership coefficient or the (strict) community to which the actor belongs. This is a more reasonable feature for community structure since it is not really possible to generalize the specific communities across different hours as the networks are entirely separate (unless we have the same actors in both - rare occurrence) — that is a community in one region for one time period is not linked to a community in another region for a time period under consideration. This might not even be true for the same region unless the communities have the same actors in both the time instances for the time period under consideration. This partial membership approach makes it different to the full (and thereby mutually exclusive) community membership as described in Section 5.1, where it was used to improve upon the link mining algorithm and explore the evolution of groups over time.

8.3 Content Features

In addition to the network and community features mined from the link structure of the networks, we believe that the language content present in the dialogs offers additional clues about the user's regional identity. We hypothesize that there is sufficient information present in the language that can aid in classification, signalling that there are specific topics that are being discussed more frequently in each type of region. Due to the dissimilarities between written and chat data, it is not feasible to use a tagger and parsers trained on corpus data consisting of written text to analyze the chat data. The task is further complicated by the users' frequent use of acronyms, nicknames and emoticons. Consequently, we opt for a shallow approach using n-gram analysis.

The vector space model [SWY75] represents each document as a vector of features using the terms (e.g. n-grams), with the individual value of each feature denoting the presence or absence of the term within the document (N features for an N word document). A variation is to use term frequency-inverse document frequency (TF-IDF) [SM86] which discounts terms occurring frequently in all the documents. [MS00] gives the following formula for calculating TF-IDF $w_{ij} = tf_{ij} \times log(\frac{N}{n_i})$, where term i's weight in a document j is given by its frequency in j and the log of its inverse document frequency in all the documents, with n_i representing the documents in the collection that contain the term i and N the total number of documents. This approach has been successfully used for topic identification in IRC data by the MIT Butterfly agent [DLM99].

To extract word features for our dataset, we followed the procedure outlined below:

1. First we partition the data into chunks consisting of all spoken dialogs for the time period under consideration. Documents in our dataset simply correspond to all the utterances from each unique user within the time period (for each region), since we are not using the semantics, constructing documents in this manner lends itself nicely to the n-grams approach taken here (where each n-gram is considered independent). For the term frequency, we take the term frequency for all the dialogs spoken by the user, and inverse document frequency is computed from all the dialogs from all the users within the same region.

2. Next, we tokenize the dialogs to extract the tokens from this chunk for each user from a region using unigrams.

3. A stop words list is used to eliminate commonly occurring articles, pronouns, helping verbs and salutations and words that have a high frequency of occurrence or lack of meaningful information.

4. The Snowball stemmer [Sno01] is used to stem the words to their root removing duplicates and thus creating unique terms for each region that can be used as a feature vector.

5. The top k terms from each region are selected based on classification cross-validation (using various algorithms as described in Section 8.4); terms are removed until the classification performance declines significantly. Adding bigrams and trigrams drops the classification performance, possibly because of the large number of typos, acronyms, and emoticons present in our dataset.

Table 8.2: Hourly token counts

Hour	**Number of Tokens**
1	3433
2	3175
3	3215
4	2261
Total	11,994
Reduced	603

6. We evaluated two feature variants: 1) binary encoding of the presence/absence of terms and 2) weighting the terms according to a TF-IDF measure. Both feature variants resulted in approximately equivalent classification accuracy so we used the binary encoding for our feature set.

The dataset considered for the classification consists of over 500 hours of data for a period of 4 days across 6 regions (mentioned in Table 8.1). We randomly selected one hour of data from the four different days for the term vector analysis. Table 8.2 summarizes the token counts over the dataset. There were 11,994 tokens overall, which after processing reduced to 603 terms that were then used for classifier training.

8.4 Classifier Training

For our classification task we evaluated the performance of four supervised learning algorithms using the Weka machine learning workbench: 1) decision-trees 2) Bayesian belief nets 3) k-nearest

neighbor and 4) Naive Bayes [Wek09]. The basic decision-tree algorithm minimizes the entropy measure (maximizing the information gain) over a set of attribute values in order to generate a classification tree based on the attributes that best predict the class from the training data. In our system, we employ the C4.5 implementation from Weka, known as J48. For the Bayesian belief network, we learn both the structure and the conditional probability distributions directly from the data employing the Weka BayesNet implementation. K-nearest neighbor (kNN) is a popular instance-based algorithm that predicts the class label for the test instance using majority voting among a local neighborhood composed of the k training instances that lie closest to the query. We employ the Weka implementation of KNN, with Euclidean distance on normalized attributes and the choice of k determined using leave-one-out cross validation. The Naive Bayes classifier is based on the simplifying assumption that the attribute values are conditionally independent given the target value such that given the target value the probability of observing the conjunction $a1, a2, ..a_n$ is the product of the probabilities for the individual attributes.

8.5 Results

A general description of the activities that the users tend to perform in each region is shown in Table 3.2. The regions fell into three different general categories: 1) orientation areas for new users to learn how to interact with Second Life, 2) sandbox areas that permit users to experiment with building construction, and 3) general entertainment areas (e.g., beaches). The number of

Table 8.3: Network user counts

	Help Island Public	Help People Island	Mauve	Kuula	Pondi Beach	Moose Beach	Total
Hour 1	34	19	7	25	27	25	137
Hour 2	30	18	5	16	15	23	107
Hour 3	32	15	6	19	18	24	120
Hour 4	28	22	5	19	16	21	109

actors in each region for the four hours under consideration is shown in Table 8.3. To determine the stability of our regional comparisons, we evaluated networks formed in the same region over multiple days.

We consider two networks for analysis: one the original network obtained using our SSTO algorithm and the other obtained by randomly choosing an equivalent number of actors from this original network. This was done to unbias the affect of network size on the centrality measures. The classification task is to identify the region given the feature set. We evaluated the following feature sets:

- all the centrality measures (betweenness, closeness and degree),
- the most predictive centrality measure, closeness, only,
- community membership, expressed as the number of communities each user belongs to,
- the top-k words from the term vector model.

We included the centrality measures (as explained in Section 8.1) as a feature for classification, since these provide a picture of the properties of the network based on the connectivity infor-

Table 8.4: Classification accuracy by feature set (random chance level=16.6% for the 6 class problem)

Feature Set	**Classification Accuracy**
All Centrality	25.2%
Closeness Only	25.8%
Community #	27.5%
Words (unigrams)	34%
All Centrality and Community	21.4%
All Centrality and Words	33.9%
Closeness and Community	**44.4%**
Closeness and Words	38.1%
Community and Words	34.9%
All Features	**54.3%**

mation for individual actors. We report the results for using all the centrality measures (Degree, Betweenness and Closeness) and also using the closeness separately, since we found that most of the predictive power for the centrality features comes from the closeness alone.

The statistics shown in the Table 8.4 were obtained by combining the results from classification for leave-one-out validation for the best classifier. For the leave-one-out validation we made four splits, where the data from three hours was used for training the classifier and the data from the fourth was used as the test set. We selected this scheme to evaluate whether the learned patterns persist over time and generalize to completely different sets of actors and social networks from the same region. The same goes for the term vector, where we wanted to see if the topics discussed are sufficiently repetitive to be able to aid in classification task. We performed the classification for each of these four splits using all the four algorithms and selected the best performing algorithm for each feature set.

We can make the following conclusions based on the results shown in Table 8.4:

- Unsurprisingly, using a combination of all the features provides the best classification performance, 54.3%
- Using any of the features individually, or any combination, provides classification performance better than random chance levels (16.6% for the six class problem), showing all the features carry important information.
- For individual features, the best performance is obtained from using the words (34%)
- For the combined features, the best performance is obtained by combining closeness with community membership information.

The confusion matrices (number of misclassified instances by category) for the best two combined feature sets, all and community plus closeness are shown in Table 8.5 and Table 8.6. The columns are organized by the prevalent activity in the region:

orientation areas: HIP (Help Island Public) and HPI (Help Public Island)

sandboxes: Kuula and Mauve

scenic areas: M. Beach (Moose Beach) and Pondi Beach (P. Beach)

Looking at both the confusion matrices, we fail to observe any activity-based trends resulting in increased confusion between regions of a similar type. This indicates that the network structure, communities, and topics of discussion in regions with similar activities are quite different. Note

Table 8.5: Confusion matrix (all measures)

	HIP	HPI	Kuula	Mauve	M. Beach	P. Beach
HIP	58	1	10	0	8	5
HPI	7	35	4	1	13	6
Kuula	10	1	34	0	7	3
Mauve	2	0	0	17	2	1
M. Beach	7	6	3	3	21	9
P. Beach	14	0	5	0	9	53

Table 8.6: Confusion matrix (closeness and community)

	HIP	HPI	Kuula	Mauve	M. Beach	P. Beach
HIP	46	8	13	0	3	5
HPI	10	21	14	4	2	1
Kuula	2	12	24	0	6	3
Mauve	0	0	0	15	0	0
M. Beach	15	5	18	0	12	10
P. Beach	3	7	14	1	2	26

that there are always multiple activities available even in regions that are predominantly of one type; for instance, Mauve also has a shopping area in addition to the sandbox. Hence it is possible that the bots are simultaneously hearing public chat from users participating in different activities.

8.6 Using Topic Modeling to Improve Classification

One of the challenges we faced when doing the n-gram analysis was feature reduction. Using the classical approach, we started with about 6000 n-grams and kept eliminating features until there was a significant hit in classification performance. This yielded about 200 n-grams that were then used as features. Using topic modeling, we not only were able to extract a higher level semantic relationship than n-grams, but it also acts as a feature-selection or compression

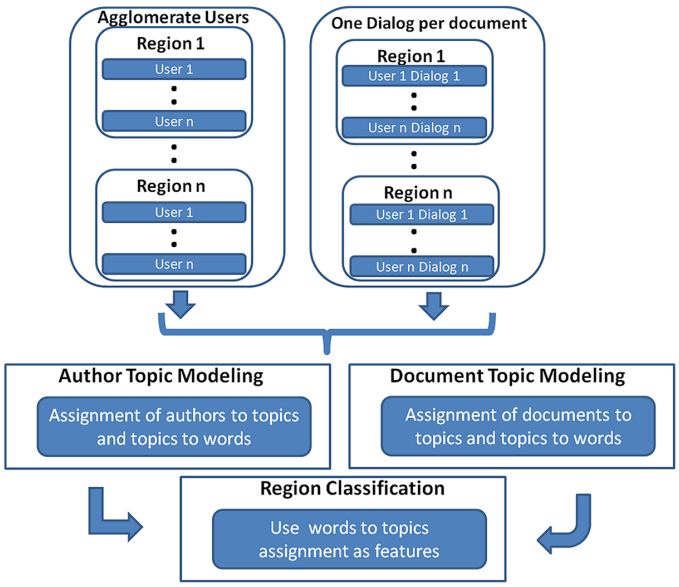


Figure 8.1: Using Topic Modeling to aid in classification.

mechanism, reducing the thousands of n-grams to a few hundred topics, that can then be used as features.

Figure 8.1 shows the adopted approach. We explore the use of both the author-topic and document topic models (as described in Section 7). We evaluated two variations on what constitutes a document:

1. In one setting, we treated each utterance as a document of its own.

2. For the second setting we agglomerated all the dialogs from each user into a document of its own, that is one document per user per region.

Using topic models for chat data classification has been attempted before by [TT04]. However, they used a simple multinomial PCA [Bun02] instead of LDA. Also, they only used the network or content features to predict the topics and the topic distribution probability vectors were not used in

the classification paradigm itself. Our use of topic based features is similar to Blei [BNJ03] who demonstrated the feature reduction properties of topic modeling for a document classification task.

8.6.1 Classifier Training and Dataset

Again, we use the same dataset described in Section 8.3. LDA topic modeling requires one to specify the number of topics, the per document topic distribution and per word topic distribution as priors on the dataset. Value for these parameters were selected using a iterative procedure, where the topic models were re-run until the topic word list seem to be related. It is to be noted that for a MCMC model like LDA, approaches like likelihood maximization of the corpus occurrence, while seeming like an attractive option, don't always work, since it is the interpretation that matters most. Furthermore, the Bayesian approach to parameter estimation doesn't apply for this reason as well, since what constitutes ground-truth (document and word memberships) is again subjective. Figure 8.2 shows the topics learned from the data alongside the probabilities of occurrence of the topic. The values used for the hyper-parameters (defined in Section 7.2) were $\alpha = 0.02$, $\beta = 0.0001$, number of iterations 2000, number of chains 10 and number of topics 200. The assignment of authors to the topics model and the assignment of the topics to the documents, were used as an additional 200 features, in combination with the features described in Section 8.

We evaluated the performance of the same four supervised learning algorithms using the Weka machine learning workbench as utilized in Section 8.4: 1) decision-trees 2) Bayesian belief nets 3)

TOPIC_1	0.00513	TOPIC_2	0.00293	TOPIC_3	0.00406	TOPIC_4	0.00573
play	0.12994	awww	0.18198	ll	0.19685	rain	0.12797
items	0.07797	kk	0.06826	great	0.09843	check	0.06981
beach	0.06498	hon	0.04551	week	0.09843	friend	0.06981
babe	0.05198	mines	0.04551	key	0.04922	walk	0.06981
features	0.03899	ummm	0.04551	muzzle	0.04922	azulle	0.06981
face	0.03899	tu	0.02277	button	0.03282	gotta	0.05817
wait	0.03899	christabelles	0.02277	lock	0.03282	ball	0.05817
@	0.03899	peu	0.02277	piers	0.03282	ahh	0.03491
laugh	0.03899	poofed	0.02277	club	0.03282	arent	0.03491
games	0.02600	soulstar	0.02277	clikc	0.01642	spanish	0.02328

TOPIC_5	0.00446	TOPIC_6	0.00313	TOPIC_7	0.00480	TOPIC_8	0.00453
gonna	0.11948	crashed	0.06390	whats	0.16676	nardok	0.14715
wanted	0.10455	hehe	0.06390	show	0.06949	rlv	0.10301
scared	0.08961	ghost	0.06390	bad	0.06949	drink	0.08829
ewww	0.05975	f0	0.06390	vampire	0.06949	gor	0.07358
means	0.05975	beautiful	0.04261	dog	0.04170	yeah	0.04415
shoulder	0.05975	poking	0.04261	news	0.04170	glass	0.04415
weapons	0.05975	pregnant	0.04261	meds	0.02780	empty	0.02944
dunno	0.04481	wha	0.04261	light	0.02780	alcohol	0.02944
things	0.02988	zac	0.04261	display	0.02780	strictly	0.02944
alive	0.02988	arm	0.04261	finally	0.02780	worry	0.02944

Figure 8.2: Topics learned using LDA.

k-nearest neighbor and 4) Naive Bayes [Wek09]. We also provide the classification performance for SVM as an additional measure. To obtain the most conservative measures, we performed ten-fold cross validation on the data.

8.6.2 Results

In this section, we show the results for the classification performance using:

- Document-Topic Model

  We evaluated two variants, one combining all dialogs from a user into a single document and the other where we have each dialog folded into its own document. Another variation we

Table 8.7: Classification accuracy by feature set using document-topic model (one document per user) (random chance level=16.6% for the 6 class problem)

Feature Set	**Classification Accuracy**
Topics Only	34.6%
All Centrality and Topics	41.7%
Closeness and Topics	44.7%
Community and Topics	40.7%
All Features	46%

tried was using eight hours instead of limiting the data to the four hours under study to see if adding more data improves the classification performance.

- Author-Topic Model

  We specified one dialog per user per document for the author-topic model, limiting ourselves to the hours considered for the classification purposes.

8.6.2.1 Using Document Topic Modeling

Tables 8.8 and 8.7 show the classification performance results using the document-topic model. Table 8.7 shows a better classification performance using the second scheme (for the topics) over the first in Table 8.8. Also, the results improve upon those obtained using the n-grams approach, save for the case where all features are combined.

Additionally, we wanted to see if adding more hours (content) improves the classification performance using the topic models. For evaluating this task, we limited ourselves to using the topics

Table 8.8: Classification accuracy by feature set using document-topic model (one document per dialog) (random chance level=16.6% for the 6 class problem)

Feature Set	**Classification Accuracy**
Topics Only	40.4%
All Centrality and Topics	51%
Closeness and Topics	52%
Community and Topics	44%
All Features	47%

Table 8.9: Classification accuracy by feature set using document-topic model (eight hours of data) (random chance level=16.6% for the 6 class problem)

Feature Set	**Classification Accuracy**
Topics Only	47%

only and used eight hours for training the learning algorithms. The results from the this approach (Table 8.9) show as much as a 7 percent performance improvement for classification against those obtained using fewer hours. Hence it seems that adding more data is of key importance to improving the utility of topic-based features.

8.6.2.2 Using Author Topic Modeling

For the author topic model we used the approach where we group together all the dialogs from the user in the same document. The results for the classification performance presented in Table 8.10 show that the performance is a little worse than using the same scheme for the document-topic model as shown in Table 8.7, but still better than the n-gram words alone in Table 8.4.

Table 8.10: Classification accuracy by feature set using author-topic model (with random base chance being approx. 17% for the 6 class problem)

Feature Set	**Classification Accuracy**
Topics Only	38.1%
All Centrality and Topics	46.7%
Closeness and Topics	46.7%
Community and Topics	37.1%
All Features	48.7%

We can make the following conclusions based on the results:

- The document-topic model seems to give better performance over using the author topic model.
- Using one document per dialog per user yields better performance than one document per user for the document-topic model.
- Using multiple hours for the document topic model greatly improves upon the classification performance (using only topics).
- The topics yield classification performance at least as good as using the n-gram words or better. Even in the case where the classification performance is equal they are useful because of their feature reduction properties.
- The combination of topics with the centrality or community measures, results in a classification performance that is much better than combining words with any of these features, with the combination of closeness with the topics almost equaling (in the best case) the performance obtained using all the features in the words case.

- The combination of all the features results in a performance that is actually inferior to that obtained from combining words with all other other features. This is the only case where the topics don't seem to make a difference, and it could be due to the large amount of features.

CHAPTER 9

CONCLUSION

The principal contributions of our work are:

1. the creation of an agent architecture suitable for mining social interactions in a variety of massively-multiplayer online games with minimal modification;

2. a set of algorithms for robust conversational partitioning and social network extraction on unstructured dialog data;

3. a longitudinal analysis of the persistence of communities over time in Second Life;

4. a classification procedure for using network, community, and content features to identify the chat context;

5. a set of algorithms for combining temporal, topic, and other semantic information for link prediction. Depending on the type of information available and privacy constraints, our algorithms can either leverage the timing information or the content information to predict friendship connections.

Our approach to link mining is unsupervised, scalable and avoids the inherent ambiguity in determining the shifting topic of communication in an open-ended environment, where the actors

can come and leave at any time. For our dataset even humans have trouble in conversation partitioning tasks (as described in [SS11]). While we have demonstrated our technique for a virtual world dataset, we believe that it is very applicable to real-world situations and provides a way to deal with the type of exchanges that occur in today's social networking environments.

Our empirical study of region-based user groups in the Second Life virtual world reveals that the combination of network and community features can be more predictive of user groups than the actual content of the conversations. One explanation for this phenomena is that more meaningful dialogs might be exchanged on private chats; we notice from our dataset that the conversations occurring in the public chat forums are very similar even in regions where the users are participating in different activities.

Due to the volume of user traffic and the variable duration of user stays, the network structure and the number of communities differ substantially across regions, apparently resulting in more predictive classification features. This holds true even across data sampled from the same region on different days. Even though many of the actors are different from day-to-day, the network and community features retain some similarity. Our hypothesis that the form of the network follows the function that the users need to pursue their activities is not supported by the observed confusion matrices since the classifier mispredictions do not follow simple activity based trends.

Although most earlier studies on group dynamics [SH04] have been conducted on individuals connected by long-standing social interactions, humans can form groups that exhibit group behavior patterns and biases within a few seconds of minimal interaction, even without face-to-face contact or prior history;Second Life is an interesting research testbed since it contains a large num-

ber of groups of this nature. From our regional analysis, it is apparent that many of the commonly used network measures differ substantially across days as well as regions, even if there is continuity across groups of actors. This indicates that studying the effects of virtual worlds on social groups is a challenging problem since many observable group measures can change substantially without any environmental modifications.

APPENDIX A: A SOCIAL RECOMMENDATION SYSTEM FOR SECOND LIFE

Here we describe the design and training of a social recommender system that assists user navigation in Second Life (SL), a massively multi-player online environment. Second Life allows users to create a virtual avatar and explore areas constructed by other users. However, unlike the real world, virtual attractions can be constructed within hours and previous ones can fall rapidly into disuse. Without recent information about the state of regions in the world, it is impossible to accurately assist users' searches in the virtual world. Second Life's default search mechanism relies strictly on meta-data voluntarily provided by land owners, which is not always reliable. We suggest that the best solution to this problem is to leverage information gleaned directly from the user population about their explorations in the virtual world. Since Second Life supports both physically-based explorations and hyperlink teleportation through URLs, it is unclear whether the recommender system should rely on techniques that work well in physical environments such as destination prediction and location ratings or function in a manner similar to web searches. In this article, we describe our evaluation of two different recommender paradigms: 1) collaborative filtering based on user ratings of locations and 2) tag-based search using WordNet to automatically expand user-provided labels. Our system runs in real-time and presents the user with information via a virtual heads-up display (HUD). We demonstrate significant improvements in user satisfaction levels over the default Second Life search mechanism; moreover users exhibit a definitive preference for our tag-based search mechanism over item-based collaborative filtering.

With the increasing number of places to visit and things to do in Second Life, it is difficult for a user to explore all the possibilities and places to visit. The game supports a number of personal modes of travel (walking, flying, teleporting) in addition to enabling users to create their own

vehicles. Although Second Life offers 3D visualization, the keyword search mechanism offered by the SL user interface is fairly limited and more appropriate for searching text-based information sources. Net lag can often deter users from doing extensive physical explorations of Second Life in areas with large numbers of user-created scripts and objects, which are paradoxically the most interesting locations. Users typically find new places through personal exploration, tips from their friends, and chancing upon good keywords. This motivates the need for a social recommender system that can suggest places to visit, personalized with the user's destination preferences.

Although it is easier to collect data on users' travel patterns in a virtual world than in the real world due to the relative ease of constructing virtual sensors, compared to using cameras, GPS, or wireless beacons to track user movements, analyzing users' destinations in virtual worlds offer a unique set of challenges:

- the lack of consistent geocoding information;
- the ability of users to teleport instantly to destinations;
- the lack of constraining lifestyle factors such as a need to sleep or go to work at a regular time.

These challenges combined with the rapidly shifting nature of the SL landscape suggest that a regularly-updated, data-driven model is required.

Here, we describe our framework for labeling, predicting, and recommending user destinations within Second Life. By leveraging location information gleaned directly from the user population

(a) User can label the place or request a suggestion.

(b) User labeling a place using categories and tags.

(c) User requesting a recommendation using category labeling.

(d) User requesting a recommendation using tags.

(e) Returned results displayed as SLurls.

Figure .1: Screenshots of a user's avatar in Second Life using our social recommender system displayed in the HUD (bottom left).

from their explorations in the virtual world, our social recommender system offers users two different options: 1) a text-based search that utilizes user-provided tags generalized with the WordNet lexical database [Mil95] and 2) item-based collaborative filtering based on previous user ratings.

.1 Related Work

Second Life is a unique test bed for research studies, allowing scientists to study a broad range of human behaviors. The ease of creation and interaction with the objects in virtual world enables the rapid exploration of new product designs and customer appreciation studies [Rhe07]. Additionally, social scientists are using Second Life to study norms and etiquette in dressing and meeting people [FSS07]. Several studies on user interaction in virtual environments have been conducted in SL including studies on augmented reality [LMZ08], conversation [WTR08], gestures [KRA08], collaborative construction [KA08] and virtual agents [BSE08, PV08].

Navigating in Second Life environment is simultaneously similar to browsing through web pages and moving through 3D game environments. It is possible to use a variety of tools to move from place to place including Second Life's inbuilt text-based search, as well as personal transportation modalities such as walking, running and flying. Users can also instantly teleport to known locations using hyperlinks (SLurls) or a map-based interface.

The problem of navigation assistance in 3D real-world environments has been addressed by researchers creating GPS-based driver assistance systems such as [LKH06]. Approaches such as

explicitly modeling transportation modality [LFK04] or using inverse reinforcement learning to learn the driver's reward function [ZMB08] have proven fruitful in predicting driver future routes from GPS data and offering. However, these systems depend on mapping the user's movement to a road network, which is not feasible within Second Life. Within computer games, there has been some research on predicting players' future quest goals using dynamic Bayesian networks [AZN98]. Mott *et al.* (2006) demonstrated a similar goal recognition system for interactive narrative environments using scalable n-gram models. However, these systems only examined the user's long-term goals, rather than trying to make predictions or recommendations about immediate destinations.

Recognizing user search patterns has been studied extensively in the context of text-based search interfaces. For instance, Teevan *et al.* (2005) created a web search assistance agent that builds a rich user profile by monitoring the user's document-related activity (search history, bookmarks, and email). Although Second Life destination searches can be conducted using a text-based search tool, most users use the 3D visualization to move between locations and rely on the text-search tools only for browsing across regions. Due to the anonymous nature of Second Life, it is most feasible to utilize in-game information to improve user searches, since many users are reluctant to divulge any information about their real-world identifies.

Given that most of the attractions in Second Life are user-constructed, the problem of searching within SL can be considered a 3D analog to searching user-generated text content such as blogs. In blogs, there is an increasing reliance on the "wisdom of the crowds" [Sur05] to supplement text-based search with user-provided tags. Tag extraction from user queries has been examined within

the context of information retrieval for question answering (e.g., [PH01]) and various TREC (Text REtrieval Conference) competitions [SGS06, SKB07] using shallow parsing for semantic analysis of the queries. The objective of these systems was to extract the most relevant text from long text documents with constraints on time and size; due to the availability of a huge corpus from past competitions, they were also able to employ machine learning. In contrast, our system relies much smaller database of user-provided labels which we augment using WordNet to provide the best set of locations in response to the user query.

With the recent Netflix prize competition, the problem of increasing the accuracy of recommender systems has received renewed research interest. There are two principal approaches to collaborative filtering: user-based and item-based [SKK01]. User-based systems rely on finding a neighborhood of similar users; memory-based or model-based methods are then used to generate a prediction of the ratings for the current user [BSH98]. In contrast, item-based systems compute the similarity between items and estimate the user's preferences based on user rating histories for similar items. In this recommender system, we use an item-based similarity approach since it can be computed in advance and requires little or no user information. There are many methods for calculating item-based similarity, including the commonly used cosine-based similarity and the weighted Pearson-correlation coefficient [SKK01]. For our implementation, we implemented a modified version of Slope One recommendation [LM05], which often has superior performance over other regression based approaches. Due to privacy reasons, we do not collect any demographic data on our profiles so our system must rely on item-based similarity measures. Also our

recommender system must contend with sparse data since we do not have recommendations for most areas, nor do we have many recommendations from any single user.

.2 Software System

To acquire data on users' travel patterns, we developed a custom tracker object using the Linden Scripting Language (LSL). Users carrying the object are periodically prompted to enter information describing their current location. The tracker object appears as an HUD that can be worn on the right or left of the avatar that monitors the user's current (x, y, z) location, as shown in Figure .1. Additionally the tracker estimates the local population density by counting the number of other users within 10m of the user. The tracker commences operation when the user clicks on the object and periodically prompts the user to provide information about his/her current location. The operation of our tracker is described in [SBS09].

Users have the option of marking a place as belonging to seven possible categories (artistic, camping, educational, entertainment, shopping, residential, other). Our GUI allows users to designate a location as belonging to multiple categories and to enter additional descriptive tags through a text field. Users are prompted to rate their interest in the location on a five star scale. The information is sent as a web request to the web server, where it is stored in a MySQL database. The data collection and recommendation can be performed simultaneously for multiple users using our multi-threaded Java server application.

Our recommender system was implemented as a second custom object in the Linden Scripting Language (LSL), with the same monitoring properties as the tracking object but with a different HUD (heads-up display). Through the HUD, users can request a destination using a text-based query that describes the place they are looking for and receive recommendations in the form of SLurls, an in-world analog of the hyperlink references used in WWW. When the user clicks on the SLurl link, he/she is teleported directly to that destination. To build a model of the user's preferences, we allow users to provide ratings for the SLurl. The user is presented with an option to correct the labeling and optional tagging annotation for the location if he/she thinks that the location does not fit the category.

.3 Learning the Map

We studied the performance of a variety of supervised classifiers on the first part of the task—learning a mapping between Second Life locations and destination categories. Note that users were allowed to assign multiple labels to locations so points were not classified as belonging exclusively to a single category. Using the tracker we collected the following features:

- Second Life username;
- (x, y, z) coordinates;
- region name;

Table .1: Classification performance for category prediction

Classifier		Accuracy (%)	Learning Time (sec)
Trees	C4.5	85.48%	0.23
Lazy schemes	KNN (K=11)	85.38%	0
Bayesian schemes	Bayes Net	82.31%	0.59

- user population count for the local area;
- date and time stamp;
- user-supplied category annotations

We conducted a study evaluating the performance of various supervised classifiers. Classification accuracy was estimated using 10-fold cross-validation, and a corrected t-test (adjusted for folds) was performed. The time required to train each classifier was also recorded. C4.5 and kNN (K=11) achieved 85% accuracy with no significant difference between the two; we opted to use kNN for our current system, due to the slightly superior learning time (see Table .1).

.4 Predicting Users' Future Destination

To predict the user's movement we trained three separate models, each employing the M5P [Qui92] algorithm for numeric prediction. Our basic procedure for making the destination prediction is summarized in Figure .2. All user models were constructed in real-time. Data collected from all user logs is consolidated incrementally into a single unnormalized table for classification. This data

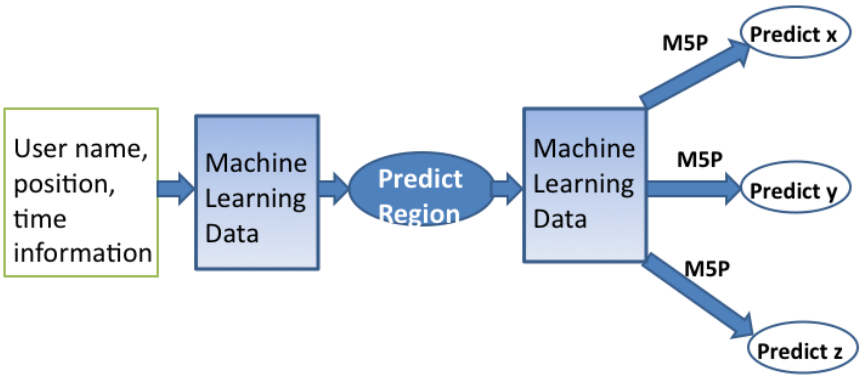


Figure .2: Prediction method. The user's destination (x, y, z) is predicted using three independent models, each employing the M5P algorithm for numeric prediction.

is used to create, incrementally update the learning model, and predict the user's next location. To evaluate our destination prediction approach, we evaluated the correlation coefficient of our x,y,z predictions using M5P and KNN (K=11). We see that there is excellent correlation between predicted and ground-truth values in all dimensions for both classifiers (no significant difference) with the z prediction being the most accurate since most users only visited ground-level destinations.

.5 Cluster-based Collaborative Filtering

Despite the reasonably good accuracy of our destination prediction system, many users indicated that they preferred a user-driven recommender system to allow them to request destination options

Table .2: Destination prediction performance

Destination prediction performance		
Classifier		Correlation Coefficient
KNN (K=11)	x	0.77
	y	0.69
	z	0.94
M5P	x	0.72
	y	0.63
	z	0.92

rather than having the destination prediction system autonomously suggest alternate destinations. To do this, we implemented an item-based collaborative filtering algorithm to generate SLurl recommendations based on user ratings. Raw labeled data points (x,y,z coordinates) were partitioned according to category using unsupervised k-means clustering (Euclidean distance minimization). Cluster centers (referred to as items here on) were indexed according to the average user-rating of the points. This ranking was used to make the first recommendation to new users without any rating history.

After collecting some ratings from the user, we leverage them to make a more personalized recommendation based on item similarities and rating history by calculating a Slope One recommendation [LM05] with the bipolar method. For the bipolar method, we consider both the likeness and dislikeness of the user to the other users' based on the ratings. In our system, likeness is defined as a rating of three or above on scale of five and dislikeness otherwise. Cluster centers in the same category are considered to be similar. The recommendation is returned to the request HUD as a SLurl. If the user clicks on the SLurl link, he/she is teleported directly to that destination. To

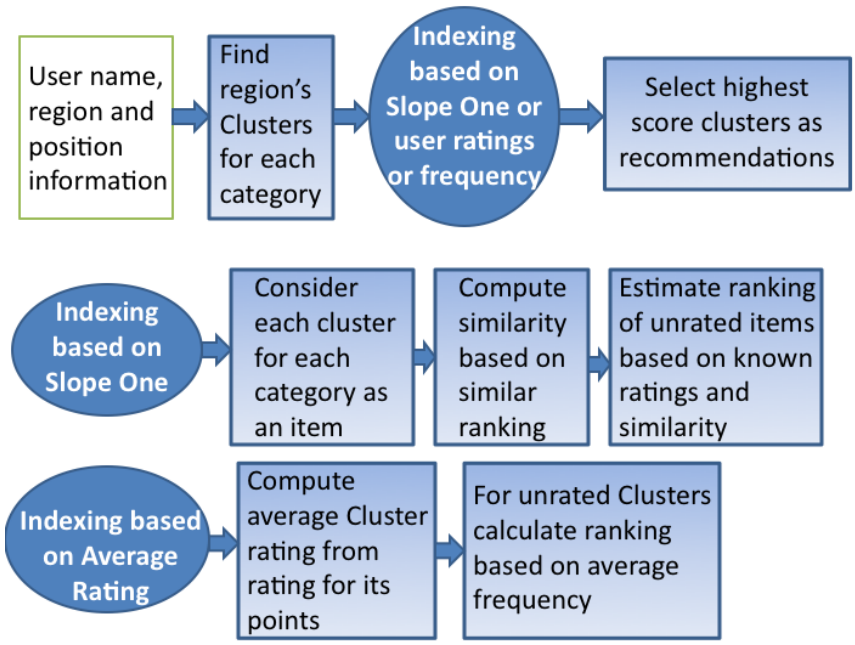


Figure .3: Item-based collaborative filtering

capture user feedback, user provide ratings on a five point scale for the link he/she has teleported to.

Our procedure for making the recommendation is as follows:

1. First, we smooth out the raw data, using k-NN to assign points to cluster centers. Each cluster center is calculated using a category-wise partition of the data.

2. Cluster centers are ranked in the order of the user rating after calculating the average score of the cluster.

3. In the case of no ratings, cluster centers are ranked by the frequency of the points that belong to each of the clusters.

4. The system responds to each request for recommendations by providing the next five suggestions in order of computed index, using one of the above two methods, such that the first three recommendations are from the same region and the last two from other regions.

5. Once we have some initial ratings from the user, we use the bipolar Slope One scheme to make a recommendation based on the similarity or dissimilarity of the user rating to other users that rated similar items similarly.

6. We consider items that belong to the same category as being the same and compute the likeness assuming that if the rating for the item was above the average rating for that user, it is liked, otherwise it is disliked.

7. The ratings for the user are then predicted from the average rating of the user and the similar user ratings for similar items, sorted on this rating. The recommendation changes as more user ratings become available.

Formally, the deviation matrix (the deviation of item i with respect to item j) is calculated as shown in the equation .1:

$$dev_{j,i}^{\text{like}} = \sum_{u \epsilon \mathcal{S}_{j,i}^{\text{like}}(\chi)} \frac{u_j - u_i}{|\mathcal{L}_{j,i}(\chi)|} \tag{.1}$$

where u_i is the user rating for item i. The prediction of item j based on rating of item i is either $p_{j,i}^{\text{like}} = dev_{j,i}^{\text{like}} + u_i$ or $p_{j,i}^{\text{dislike}} = dev_{j,i}^{\text{dislike}} + u_i$ depending on whether i is taken from $\mathcal{L}(u)$ or $\mathcal{D}(u)$

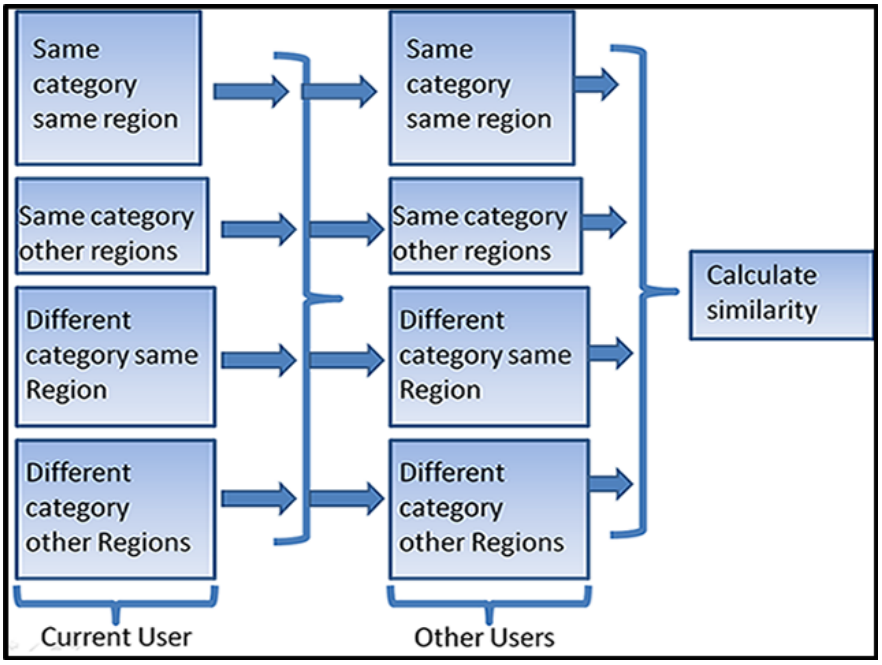


Figure .4: Item similarity computation

respectively, where $\mathcal{L}(u) = \{i \in \mathcal{S}(u)|y_i > u\}$ and $\mathcal{D}(u) = \{i \in \mathcal{S}(u)|y_i \leq u\}$ are the two sets of items. The bipolar prediction is then given as:

$$p(u)_j = \frac{\sum_{i \in \mathcal{L}(u)-\{j\}} p_{j,i}^{\text{like}} c_{j,i}^{\text{like}} + \sum_{i \in \mathcal{L}(u)-\{j\}} p_{j,i}^{\text{dislike}} c_{j,i}^{\text{dislike}}}{\sum_{i \in \mathcal{L}(u)-\{j\}} c_{j,i}^{\text{like}} + \sum_{i \in \mathcal{D}(u)-\{j\}} c_{j,i}^{\text{dislike}}}$$

where the weights $c_{j,i}^{\text{like}} = |\mathcal{L}_{j,i}|$ and $c_{j,i}^{\text{dislike}} = |\mathcal{D}_{j,i}|$ are weighted based on the number of items rated for each. For the user similarity computation based on the ratings for similar items multiple scenarios may arise; these are summarized in Figure .4. Figure .5 shows the overall system architecture.

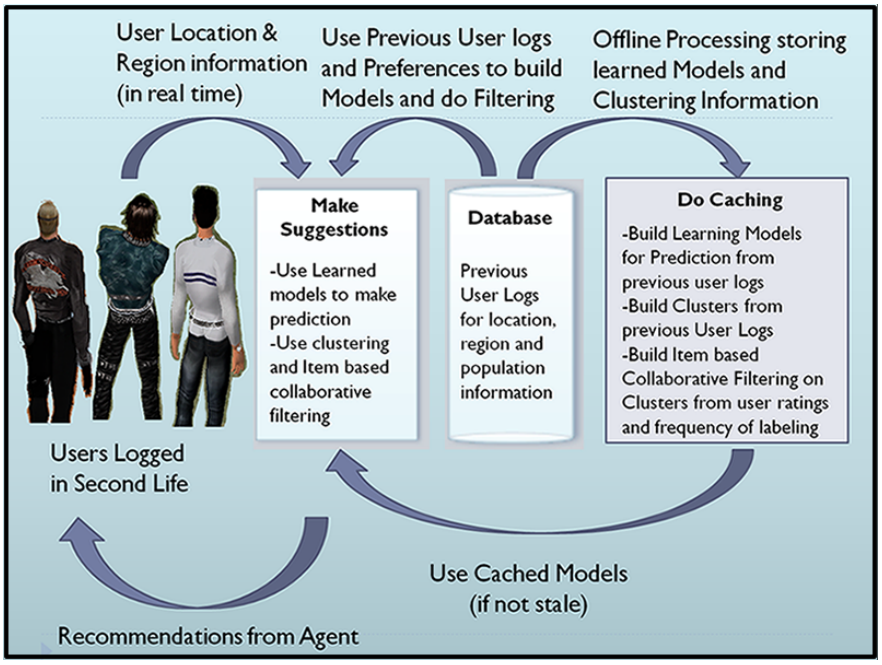


Figure .5: Architecture for item-based collaborative filtering

.6 Tag Based Search

To build a searchable human-tagged index of Second Life, we collected data during a free-form study in which we asked the users to specify tag descriptors for interesting areas. To generalize the small set of user-provided tags, we use the WordNet lexical database [Mil95]. WordNet groups words into related sets called synsets (cognitive synonyms), each of which is a distinct concept. Semantic and lexical relations among synsets are organized into hyponym/hypernym, synonymy/antonymy hierarchy and holonymy/meronymy relationships. Our WordNet-based similarity match simply returns all the hypernyms, hyponyms and synonyms for a concept using the MIT Java Wordnet Interface [Wor09]. Table .3 shows some user-provided tags, along with the related concepts generated by WordNet. Note that our recommender system does not utilize all

Table .3: Tags and Related Concepts from WordNet

Tag	**Related Concepts**
shack	hovel,hutch,iglu,shelter,shanty,hut,shack,igloo
pants	underpants,bloomers,pants,drawers,knickers
camping	encampment,camping,habitation,inhabitation,bivouacking,inhabitancy,tenting
hangout	haunt,area,stamping ground,hangout,resort,repair,gathering place,country

the hyponyms, hypernyms and synonyms from WordNet, but only the subset of words that occur at least once in the user-provided tags. We collected 390 different tags from users which were reduced to 111 related concepts using the following offline procedure. To calculate the similarity, we examine each of the tags to determine whether it is already accounted for in the concept cache. If not, we add an entry for that tag in the concept cache, make a separate table for the tag and add all subsequent related concepts into the same table. This results in a similarity-based agglomeration of the region annotations that can be used to perform k-means clustering on the raw three-dimensional location coordinates using a variation of the unsupervised K-means clustering algorithm with a Euclidean distance minimization measure. This has the positive effect of reducing the noise in the data, as the nearby points for the same concept are clustered together. Clusters are sorted by user frequency such that the more frequently visited locations (concept clusters with a greater number of data points) have higher indices.

Algorithm 1 shows how the concepts are used to make recommendations based on both membership similarity measure and user frequency indexing. We match the tags specified by the user and calculate all the similar concepts such that the highest number of matching tags with the concept and its related concepts percolate to the top. We choose from each of the regions, matching

cluster centers, one by one in order of frequency indexing, for the most related concept. This step is repeated as many times until 30 recommendations are generated and this sorted cluster center list is returned to the in response to user queries, five recommendations at a time.

input : Clustering of the data points into concepts and related sub-concepts across all regions, user_request_tag
output: Recommendation of locations

```
concepts_to_make_recommendation_for;
no_concept_matches_in_each;
ForEachtag in user_request_tag
ForEachconcept in concept_cache if ∼ concept = tag then
    ForEachsubconcept in concept if subconcept = tag then
        concepts_to_make_recommendation_for.add(
        concept);
        no_concept_matches_in_each.put(concept,
        no_concept_matches_in_each.get(concept,
        count)+1);
    end
end
Else concepts_to_make_recommendation_foràdd(concept);
no_concept_matches_in_each.put(concept,1);
sort(concepts_to_make_recommendation_for
no_concept_matches_in_each);
```

Algorithm 1: User request processing

.7 Results

To evaluate the utility of our social recommender system, we performed a user study with fifty-eight users. The purpose was to evaluate the performance of the recommender system at assisting users' with a navigation task. The users were asked to perform three sessions: 1) labeling/marking

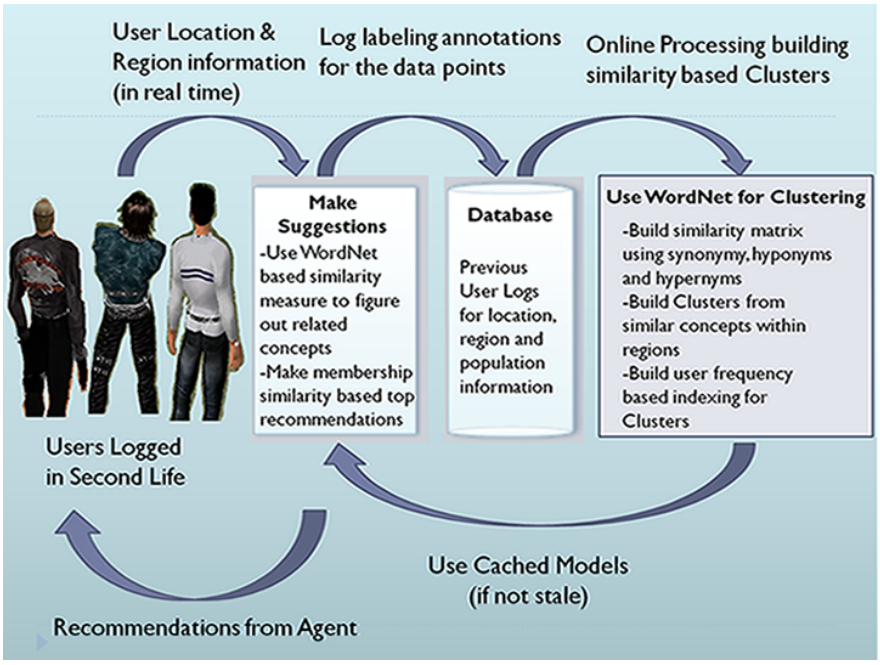


Figure .6: Architecture for tag-based search

2) searching using category-based collaborative filtering 3) searching using tag-based search augmented with WordNet. We did not impose any restrictions on the search or require them to visit particular places within Second Life. During the study, users were asked to do the following:

- Label five locations by specifying the category label and the tags for the place
- Perform at least five category-based searches using collaborative filtering and rate the results after visiting the location
- Perform at least five searches using the text-based search and rate the results after visiting the location.

Table .4: User annotations across destination categories

Category	Instances
Artistic	726
Camping	25
Educational	19
Residential	23
Recreational	117
Shopping	71
Other	83
Unlabeled	1567

Our initial labeled category dataset spanned 94 Second Life regions and included 2463 entries based on previous data collected from 24 users. 29 users provided 997 tag entries over 165 regions; this resulted in 632 tags and 506 concepts.

Note that we assume that the users are already familiar with the default location search options available in Second Life and solicit their opinions on the inbuilt game search without explicitly requiring them to perform searches without our interface. Users performed pre and post session questionnaires for all three sessions in which users were asked to respond to paired statements on a Likert scale with 1 indicating complete disagreement and 5 indicating complete agreement. The main section of the questionnaire is given below:

- **I experienced problems with the user interface for the game.**
- **I experienced problems with the user interface for the recommendation HUD.**
- **I would prefer to use the recommendation HUD for navigation rather than doing manual exploration.**

- **I think the recommendation HUD would improve my gaming experience. Examples of improvements include: aiding exploration, enabling better time utilization, highlighting popular areas.**
- **The accuracy of recommendation for the recommendation HUD was good.**
- **The speed of responses for the recommendation HUD was good.**
- **I found the recommendation HUD useful for finding interesting places this time.**
- **The current search is more useful for helping me find what I am looking for.**
- **I think tag-based search improved my search experience.**

Tables .5- .10 lists the Fisher's exact test tabulation and results for the results from the user-study questionnaires.

1. Table .5 and .6 show that using the HUD made a number of users to switch their opinion on the general usefulness of a recommendation HUD that can assist the users in exploration. However the Fisher's exact test shows no significant relationship between using the HUD

Table .5: Fisher's exact test tabulation and results for question: I would prefer to use a recommendation HUD for navigation rather than doing manual investigation. (Default vs. Category-based Collaborative Filtering)

	Before using HUD	After using HUD
I would like to use HUD	30	36
I would not like to use HUD	28	22
P-Value = 0.348603		

Table .6: Fisher's exact test tabulation and results for question: I would prefer to use a recommendation HUD for navigation rather than doing manual investigation. (Second Life Search vs. Tag-based Search)

	Before using HUD	After using HUD
I would like to use HUD	30	38
I would not like to use HUD	28	20
P-Value = 0.186716		

Table .7: Fisher's exact test tabulation and results for question: I think the recommendation HUD would improve my gaming experience.

	Before using HUD	After using HUD
I would like to use HUD	38	44
I would not like to use HUD	20	14
P-Value = 0.307799		

Table .8: Fisher's exact test tabulation and results for question: I think the recommendation HUD would improve my gaming experience. (Second Life Search vs. Tag-based Search)

	Before using HUD	After using HUD
I would like to use HUD	38	49
I would not like to use HUD	20	9
P-Value = 0.0309680		

Table .9: Fisher's exact test tabulation and results for question: The current search is more useful for helping me find what I am looking for. (Change in perception after each of the two searches)

	Before using HUD	After using HUD
I would like to use HUD	43	48
I would not like to use HUD	15	10
P-Value = 0.366670		

Table .10: Fisher's exact test tabulation and results for question: I think tag-based search improved my search experience. (Change in perception after each of the two searches)

	Before using HUD	After using HUD
I would like to use HUD	0	58
I would not like to use HUD	58	0
P-Value = 0.0000000		

and the change in opinion. This might be due to the fact that more than half of the users are already initially positive about its usefulness just from the description of the system. Another possibility is that the users were happier with the new search options provided but preferred the SL default viewer over the HUD visualization which seemed to confuse some of the users.

2. Tables .7 and .8 show that the category-based collaborative filtering search did not convince the users of the utility of expanded search options but using the tag-based search modified the users' opinions.

3. Table .9 show that there was no significant relationship between the users' preference for the HUD based recommendations to the default search in SL for each of the two experimental conditions. Users preferred *both* search options over the default search within SL.

4. Table .10 shows drastic improvement in user perception after the use of the tag-based search vs. category-based collaborative filtering.

Overall the feedback from the users was very positive and many of them requested that extended use of the HUD. Some of the comments provided by the users were:

- User 1: i liked it
- User 2: ok, so i use now a guide of someone else ? could be interesting, to explore places, you don't know :-) it works well but if i'm right , the places the hud show to me, will be changing all the time depends on, how many people integrate the system, right ? wow, that's a good idea from you. thxs than for the hud :-)) it's also depending, how good people are descriping the places, that's not always easy :-)) but i will have much to explore again , fantastic ! anyway, a very good initiative
- User 3: Definitely a better keyword search than SL's search.
- User 4: this is a neat. Eventually you could put in a random suggestion option would be cool, just an idea, I know how additive stumbleupon is, would love there to be one for sl
- User 5: yes...your are proud on this product!!! and you should be!!

A few examples where the users were confused and could not continue :

- User 6: it sounds complicated
- User 7: but when i want to search i can only search the categories

.8 Conclusion

Many people use the Internet to create a social presence, through blogs, avatars, and social networking sites; this presents an opportunity for researchers to collect rich user data from these interactions and research the problem of effectively creating a personalized and user-friendly experience. While we have thus far remained narrowly focused on the problem of destination recommendation for Second Life, our framework provides a rich user interface that is being used to explore the following directions as part of our ongoing research work:

1. Creating social networks of users with similar interests;

2. Supplementing Second Life data with information measures that describe the user's real-life interests.

The main contribution of this work was the creation and demonstration of a powerful social recommender system for Second Life that incorporates destination prediction, collaborative filtering, and tag-based search. Unlike the default search which relies on the metadata provided by land-owners, our system leverages the large and active user population to provide current labels and ratings for attractions within Second Life. The results for the study showed that the users not only liked the system but were impressed by its performance on non-exact tag matches. We believe that there are benefits to bringing the richness of search and natural-language processing to virtual worlds such as Second Life, allowing the users to enjoy the cost and time savings already available on the WWW. In future work, we are interested in moving toward automatic tag extraction to elicit rele-

vant tags directly from user reviews; we believe that this will us leverage additional data sources rather than relying completely on the efforts of volunteer user tagging.

APPENDIX B: IRB APPROVAL OF HUMAN RESEARCH: AGENT RECOMMENDATION SYSTEMS FOR MASSIVELY-MULTIPLAYER GAME ENVIRONMENTS

University of Central Florida Institutional Review Board
Office of Research & Commercialization
12201 Research Parkway, Suite 501
Orlando, Florida 32826-3246
Telephone: 407-823-2901 or 407-882-2276
www.research.ucf.edu/compliance/irb.html

Approval of Human Research

From: **UCF Institutional Review Board #1**
FWA00000351, IRB00001138

To: **Gita Reese Sukthankar and Syed Fahad Allam Shah**

Date: **December 11, 2009**

Dear Researcher:

On December 11, 2009, the IRB approved the following human participant research until 12/10/2010 inclusive:

Type of Review:	Submission Correction for UCF Initial Review Submission Form
Project Title:	Agent Recommendation Systems for Massively-Multiplayer Game Environments
Investigator:	Gita Reese Sukthankar
IRB Number:	SBE-09-06564
Funding Agency:	University of Central Florida
Grant Title:	Psychological Models for Intent Inference
Research ID:	1048486

The Continuing Review Progress Report must be submitted 2 – 4 weeks prior to the expiration date for studies that were previously expedited, and 8 weeks prior to the expiration date for research that was previously reviewed at a convened meeting. Do not make changes to the study (i.e., protocol, methodology, consent form, personnel, site, etc.) before obtaining IRB approval. A Modification Form **cannot** be used to extend the approval period of a study. All forms may be completed and submitted online at https://iris.research.ucf.edu .

If continuing review approval is not granted before the expiration date of 12/10/2010, approval of this research expires on that date. When you have completed your research, please submit a Study Closure request in iRIS so that IRB records will be accurate.

Use of the approved, stamped consent document(s) is required. The new form supersedes all previous versions, which are now invalid for further use. Only approved investigators (or other approved key study personnel) may solicit consent for research participation. Participants or their representatives must receive a copy of the consent form(s).

In the conduct of this research, you are responsible to follow the requirements of the Investigator Manual.

On behalf of Joseph Bielitzki, DVM, UCF IRB Chair, this letter is signed by:

Signature applied by Janice Turchin on 12/11/2009 10:57:02 AM EST

IRB Coordinator

Figure .7: IRB Approval for the Human Subjects Research (year 2009).

University of Central Florida Institutional Review Board
Office of Research & Commercialization
12201 Research Parkway, Suite 501
Orlando, Florida 32826-3246
Telephone: 407-823-2901 or 407-882-2276
www.research.ucf.edu/compliance/irb.html

Approval of Human Research

From: **UCF Institutional Review Board #1**
FWA00000351, IRB00001138

To: **Gita Reese Sukthankar** and **Syed Fahad Allam Shah**

Date: **December 13, 2010**

Dear Researcher:

On December 10. 2010, the IRB approved the following modifications/human participant research until 12/09/2011 inclusive:

Type of Review:	IRB Continuing Review Application Form
Project Title:	Agent Recommendation Systems for Massively-Multiplayer Game Environments
Investigator:	Gita Reese Sukthankar
IRB Number:	SBE-09-06564
Funding Agency:	University of Central Florida
Grant Title:	Agent Recommendation Systems for Massively-Multiplayer Game Environments
Research ID:	1048486

The Continuing Review Application must be submitted 30days prior to the expiration date for studies that were previously expedited, and 60 days prior to the expiration date for research that was previously reviewed at a convened meeting. Do not make changes to the study (i.e., protocol, methodology, consent form, personnel, site, etc.) before obtaining IRB approval. A Modification Form **cannot** be used to extend the approval period of a study. All forms may be completed and submitted online at https://iris.research.ucf.edu .

If continuing review approval is not granted before the expiration date of December 9, 2011, approval of this research expires on that date. When you have completed your research, please submit a Study Closure request in iRIS so that IRB records will be accurate.

Use of the approved, stamped consent document(s) is required. The new form supersedes all previous versions, which are now invalid for further use. Only approved investigators (or other approved key study personnel) may solicit consent for research participation. Participants or their representatives must receive a copy of the consent form(s).

In the conduct of this research, you are responsible to follow the requirements of the Investigator Manual.

On behalf of Joseph Bielitzki, DVM, UCF IRB Chair, this letter is signed by:

Signature applied by Janice Turchin on 12/13/2010 09:39:03 AM EST

Figure .8: IRB Approval for the Human Subjects Research (year 2010).

University of Central Florida Institutional Review Board
Office of Research & Commercialization
12201 Research Parkway, Suite 501
Orlando, Florida 32826-3246
Telephone: 407-823-2901, 407-882-2012 or 407-882-2276
www.research.ucf.edu/compliance/irb.html

Acknowledgement of Study Closure

From **: UCF Institutional Review Board #1**
FWA00000351, IRB00001138

To : **Gita Reese Sukthankar** and Co-PI: **Syed Fahad Allam Shah**

Date : **September 13, 2011**

Dear Researcher:

On 9/13/2011 the IRB conducted an administrative review of the FORM: Study Closure Request that you submitted in iRIS. The study has been closed within the system.

This report is in regards to:

Type of Review:	Study Closure
Project Title:	Agent Recommendation Systems for Massively-Multiplayer Game Environments
Investigator:	Gita Reese Sukthankar
IRB Number:	SBE-09-06564
Funding Agency:	University of Central Florida
Grant Title:	
Research ID:	1048486

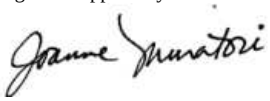
As part of this action:

- The research is permanently closed to enrollment.
- All participants have completed all research-related interventions.
- Collection of private identifiable information is completed.
- Analysis of private identifiable information is completed.

Thank you for notifying the IRB of this modification.

On behalf of Sophia Dziegielewski, Ph.D., L.C.S.W., UCF IRB Chair, this letter is signed by:

Signature applied by Joanne Muratori on 09/13/2011 10:18:26 AM EDT



IRB Coordinator

Figure .9: IRB Approval for the Human Subjects Research (year 2011).

LIST OF REFERENCES

[AC05] Ahmed Abbasi and Hsinchun Chen. "Applying Authorship Analysis to Extremist Group Web Forum Messages." *Journal of Virtual Worlds Research*, **20**, 2005.

[AD51] T. W. Anderson and D. A. Darling. "Asymptotic theory of certain "goodness-of-fit" criteria based on stochastic processes." *The Annals of Mathematical Statistics*, **23**(2), 1951.

[aGM05] Seyit Ahmet amtep, Mark Goldber, Malik Magdon-Ismai, and Mukkai Krishnamoort. "Detecting conversing groups of chatters: A model, algorithms, and tests." In *Proceedings of the IADIS International Conference on Applied Computing*, IADIS 2005, pp. 89–96. IADIS, 2005.

[AKW09] Muhammad Aurangzeb Ahmad, Brian Keegan, Dmitri Williams, Jaideep Srivastava, and Noshir Contractor. "Mining for Gold Farmers: Automatic Detection of Deviant Players in MMOGs." In *International Conference on Computational Science and Engineering*, volume 4 of *CSE '09*, pp. 340–345. IEEE Computer Society, 2009.

[AM08] Paige H. Adams and Craig H. Martell. "Topic Detection and Extraction in Chat." In *Proceedings of the 2008 IEEE International Conference on Semantic Computing*, pp. 581–588. IEEE Computer Society, 2008.

[ARS03] Paul M. Aoki, Matthew Romaine, Margaret H. Szymanski, James D. Thornton, Daniel Wilson, and Allison Woodruff. "The Mad Hatter's Cocktail Party: A Social Mobile Audio Space Supporting Multiple Simultaneous Conversations." In *Proceedings of the ACM Conference on Computer Human Interaction*, CHI 2003, pp. 425–432. ACM, 2003.

[AZN98] A. Albrecht, I. Zukerman, and A. Nicholson. "Bayesian Models for Keyhole Plan Recognition in an Adventure Game." *Journal of User Modeling and User-Adapted Interaction*, **9**:5–47, 1998.

[Bai07] William Sims Bainbridge. "The Scientific Research Potential of Virtual Worlds." *Science*, **317**(5837):472–476, 2007.

[BEF02] S. P. Borgatti, M. G. Everett, and L. C. Freeman. "UCINET 6 For Windows: Software for Social Network Analysis.", 2002.

[BHS06] P. Boer, M. Huisman, T.A.B. Snijders, C.E.G. Steglich, L.H.Y. Wichers, and E.P.H. Zeggelink. "StOCNET: An open software system for the advanced statistical analysis of social networks. Version 1.7.", 2006. Retrieved December 2010 `http://stat.gamma.rug.nl/s_man400.pdf`.

[BLM06] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. Hwang. "Complex networks: Structure and dynamics." *Physics Reports*, **424**(4-5):175–308, February 2006.

[BNJ03] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. "Latent Dirichlet Allocation." *Journal of Machine Learning Research*, **3**:933–1022, 2003.

[Boe08] Tom Boellstorff. *Coming of Age in Second Life: An Anthropologist Explores the Virtually Human*. Princeton University Press, 2008.

[BSE08] Anton Bogdanovych, Simeon Simoff, and Marc Esteva. "Virtual Institutions: Normative Environments Facilitating Imitation Learning in Virtual Agents." In *International Working Conference on Intelligent Virtual Agents*, 2008.

[BSH98] Breese, John S., Heckerman, David, and Carl Kadie. "Empirical Analysis of Predictive Algorithms for Collaborative Filtering." In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, 1998.

[BSH10] E. Bakshy, M. Simmons, D. Huffaker, C-Y Teng, and L. Adamic. "The Social Dynamics of Economic Activity in a Virtual World." In *International Conference on Weblogs and Social Media (ICWSM)*, 2010.

[Bun02] Wray L. Buntine. "Variational Extensions to EM and Multinomial PCA." In *ECML '02 Proceedings of the 13th European Conference on Machine Learning*. Springer-Verlag, 2002.

[Car91] K. Carley. "A Theory of Group Stability." *American Sociological Review*, **56**(3):331–354, 1991.

[Cas05] Edward Castronova. "On the Research Value of Large Games: Natural Experiments in Norrath and Camelot." In *CESifo*, volume 1621 of *working paper*. Social Science Research Network (SSRN), 2005. available at http://ssrn.com/abstract=875571.

[CCD09] Kathleen Carley, Dave Columbus, Matt DeReno, Michael Bigrigg, Jana Diesner, and Frank Kunkel. "AutoMap User's Guide 2009." Technical Report CMU-ISR-09-114, Carnegie Mellon University, School of Computer Science, Institute for Software Research, 2009.

[CCH09] Thomas Chesneya, Swee-Hoon Chuaha, and Robert Hoffmann. "Virtual world experimentation: An exploratory study." *Journal of Economic Behavior and Organization*, **72**:618–635, 2009.

[Chi11] Oliver Chiang. "Twitter Hits Nearly 200M Accounts, 110M Tweets Per Day, Focuses On Global Expansion.", 2011. Retrieved May 15 2011 http://blogs.forbes.com/oliverchiang/2011/01/19/twitter-hits-nearly-200m-users-110m-tweets-per-day-focuses-on-glo

[Cho00] N. Chomsky. *New Horizons in the Study of Language and the Mind*. Cambridge University Press, 2000.

[CL90] Philip R. Cohen and Hector J. Levesque. "Intention is choice with commitment." *Artificial Intelligence*, **42**, 1990.

[Cli10] J. Clippinger. “Human Nature and Social Networks.”, 2010.

[CWS09] Edward Castronova, Dmitri Williams, Cuihua Shen, Rabindra Ratan, Li Xiong, Yun Huang, and Brian Keegan. “As real as real? Macroeconomic behavior in a large-scale virtual world.” *New Media and Society*, **11**, 2009.

[CYB03] E. Castano, V. Y. Yzerbyt, and D. Bourguignon. “The impact of entiativity on social identificiation.” *European Journal oF Social Psychology*, **33**:735–754, 2003.

[Dai08] Nicole Norfleet. “UNC buys in to virtual world.”, 2008. Retrieved July 2009 `http://www.dailytarheel.com/2.1393/welcome-to-your-second-life-1.164341`.

[DDF90] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. “Indexing by latent semantic analysis.” *Journal of American Society for Information Science*, **41**(6):391–407, 1990.

[DGD05] Leon Danon, Albert D. Guilera, Jordi Duch, and Alex Arenas. “Comparing community structure identification.” *Journal of Statistical Mechanics: Theory and Experiment*, **2005**(9):P09008–09008, September 2005.

[DLM99] Neil W. Van Dyke, Henry Lieberman, and Pattie Maes. “Butterfly: A Conversation-Finding Agent for Internet Relay Chat.” In *International Conference on Intelligent User Interfaces*, 1999.

[DM03] S. N. Dorogovtsev and J. F. F. Mendes. *Evolution of Networks: From Biological Nets to the Internet and WWW*. Oxford University Press, USA, 2003.

[DM04] N. Ducheneaut and R. J. Moore. "Let me get my alt: digital identiti(es) in multiplayer games." Technical Report 6, CSCW 2004 Workshop on Representation of Digital Identities, 2004.

[Dro10] Game Drone. "Top 10 Most Successful MMOGs of 2010.", 2010. Retrieved May 15 2011 `http://gamedrone.net/2010/06/30/top-10-most-successful-mmogs-of-2010/`.

[EC10] Micha Elsner and Eugene Charniak. "Disentangling chat." *Computational Linguistics*, **36**:389–409, 2010.

[EK10] David Easley and Jon Kleinberg. *Networks, Crowds, and Markets: Reasoning about a Highly Connected World*. Cambridge University Press, 2010.

[ESK11] Thomas Erickson, N. Sadat Shami, Wendy A. Kellogg, and David W. Levin. "Synchronous Interaction Among Hundreds: An Evaluation of a Conference in an Avatar-based Virtual Environment." In *Proceedings of the 2011 annual conference on Human factors in computing systems*, CHI '11, pp. 503–512. ACM, 2011.

[FLG02] G.W. Flake, S. Lawrence, C.L. Giles, and F.M. Coetzee. "Self-organization and identification of Web communities." *Computer*, **35**(3):66–70, march 2002.

[FM07] E.N. Forsyth and C.H. Martell. "Lexical and Discourse Analysis of Online Chat Dialog." In *International Conference on Semantic Computing*, pp. 19–26, 2007.

[For05] Donelson R. Forsyth. *Group Dynamics*. Wadsworth Publishing, 2005.

[FS86] O. Frank and D. Strauss. "Markov graphs." *Journal of the American Statistical Association*, **81**(395), 1986.

[FSS07] Doron Friedman, Authory Steed, and Mel Slater. "Spatial Social Behavior in Second Life." In *Proceedings of Intelligent Virtual Agents*, volume 4722, 2007.

[GA05] Roger Guimera and Luis A. Nunes Amaral. "Functional cartography of complex metabolic networks." *Nature*, **433**(7028):895–900, February 2005.

[GK99] B. Grosz and S. Kraus. "The evolution of SharedPlans." In Michael J. Wooldridge and Anand Rao, editors, *Foundations and Theories of Rational Agency*, Applied Logic series. Kluwer Academic, 1999.

[GL96] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. JHU Press, 3rd edition, 1996.

[GN02] M. Girvan and M. E. J. Newman. "Community structure in social and biological networks." *Proceedings of the National Academy of Sciences of the United States of America*, **99**(12):7821–7826, June 2002.

[GS99] Marijtje A.J. Van Duijn Gerhard G. Van De Bunt and Tom A.B. Snijders. "Friendship Networks Through Time: An Actor-Oriented Dynamic Statistical Network Model." *Computational and Mathematical Organization Theory*, **5**(2), 1999.

[GS04] Thomas L. Griffiths and Mark Steyvers. "Finding scientific topics." *Proceedings of the National Academy of Science*, **101**:5228–5235, 2004.

[GS07] Thomas L. Griffiths and Mark Steyvers. *Probabilistic Topic Model*. Lawrence Erlbaum Associates, 2007.

[HH07] M. Huss and P. Holme. "Currency and commodity metabolites: their identification and relation to the modularity of metabolic networks." *Systems Biology, IET*, **1**(5):280–285, september 2007.

[HHJ03] Petter Holme, Mikael Huss, and Hawoong Jeong. "Subnetwork hierarchies of biochemical pathways." *Bioinformatics*, **19**(4):532–538, March 2003.

[Hof99] Thomas Hofmann. "Probabilistic latent semantic indexing." In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1999.

[HS03] Mark Huisman and Tom A. B. Snijders. "Statistical Analysis of Longitudinal Network Data with Changing Composition." *Socialogical Methods and Research*, **32**:253–287, 2003.

[HSW09] Yun Huang, Cuihua Shen, Dimitri Williams, and Noshir Contractor. "Virtually There: Exploring Proximity and Homophily in a Virtual World." In *International Conference on Computational Science and Engineering*, volume 4. IEEE, 2009.

[IAR09] IARPA. "IARPA-BAA-09-05: Reynard.", 2009.

[Int08] DFC Intelligence. "DFC Intelligence Forecasts Video Game Market to Reach 57 Billion dollars in 2009.", 2008. Retrieved May 15 2011 `http://www.dfcint.com/wp/?p=222`.

[KA08] Michael Kriegel and Ruth Aylett. "Emergent Narrative as a Novel Framework for Massively Collaborative Authoring." In *International Working Conference on Intelligent Virtual Agents*, 2008.

[KAW10] Brian Keegan, Muhammad Aurangzeb Ahmad, Dmitri Williams, Jaideep Srivastava, and Noshir Contractor. "Dark Gold: Statistical Properties of Clandestine Networks in Massively Multiplayer Online Games." In *Proceedings of the 2010 IEEE Second International Conference on Social Computing*, SOCIALCOM '10, pp. 201–208. IEEE Computer Society, 2010.

[KAW11] Brian Keegan, Muhammad Aurangzeb Ahmad, Dmitri Williams, Jaideep Srivastava, and Noshir Contractor. "What can gold farmers teach us about criminal networks?" *XRDS: Crossroads, The ACM Magazine for Students*, **17**, 2011.

[KL51] S. Kullback and R. A. Leibler. "On Information and Sufficiency." *Annals of Mathematical Statistics*, **22**(1):79–86, 1951.

[KN09] Indika Kahanda and Jennifer Neville. "Using Transactional Information to Predict Link Strength in Online Social Networks." In *Proceedings of the Third International Conference on Weblogs and Social Media*, 2009.

[KRA08] Tomoko Koda, Matthias Rehm, and Elisabeth Andr. "Cross-Cultural Evaluations of Avatar Facial Expressions Designed by Western Designers." In *International Working Conference on Intelligent Virtual Agents*, 2008.

[Lab10] Linden Labs. "LindeX™Market Data.", 2010. Retrieved May 15 2011 `http://secondlife.com/whatis/economy-market.php`.

[Lab11] Linden Labs. "The Second Life Economy in Q4 2010.", 2011. Retrieved May 15 2011 `http://community.secondlife.com/t5/Featured-News/The-Second-Life-Economy-in-Q4-2010/ba-p/674618`.

[LFK04] L. Liao, D. Fox, and H. Kautz. "Learning and inferring transportation routines." In *Proceedings of National Conference on Artificial Intelligence*, 2004.

[Lin79] Freeman C. Linton. "Centrality in Social Networks: Conceptual clarification." *Social Networks*, **1**, 1979.

[LKH06] Julia Letchner, John Krumm, and Eric Horvitz. "Trip Router with Individualized Preferences (TRIP): Incorporating Personalization into Route Planning." In *Proceedings of National Conference on Artificial Intelligence*, 2006.

[LM05] Daniel Lemire and Anna Maclachlan. "Slope One Predictors for Online Rating-Based Collaborative Filtering." In *SIAM Data Mining*, 2005.

[LML10] Miranda J. Lubbers, Jos Luis Molina, Jrgen Lerner, Ulrik Brandes, Javier vila, and Christopher McCarty. "Longitudinal analysis of personal networks. The case of Argentinean migrants in Spain." *Social Networks*, **32**(1):91–104, 2010.

[LMZ08] Tobias Lang, Blair MacIntyre, and Iker Jamardo Zugaza. "Massively Multiplayer Online Worlds as a Platform for Augmented Reality Experiences." In *IEEE Virtual Reality*, 2008.

[Mas51] F. J. Massey, Jr. "The Kolmogorov-Smirnov test of goodness of fit." *Journal of the American Statistical Association*, **46**(253), 1951.

[MGS08] Paul R Messinger, Xin Ge, Eleni Stroulia, Kelly Lyons, Kristen Smirnov, and Michael Bone. "On the Relationship between My Avatar and Myself." *Journal of Virtual Worlds Research*, **1**(2), 2008.

[MH09] Mary McGlohon and Matthew Hurst. "Community Structure and Information Flow in Usenet: Improving Analysis with a Thread Ownership Model." In *Proceedings of the Third International Conference on Weblogs and Social Media*, 2009.

[Mil95] George A. Miller. "WordNet: a lexical database for English." *Communications of the ACM*, **38**, 1995.

[MPS08] Mark S. Manger, Mark Pickup, and Tom Snijders. "Plugged into the Network? A longitudinal social network analysis of PTA formation." In *Harvard Conference on Networks in Political Science,*, 2008.

[MS99] C. D. Manning and H. Schutze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, Massachusetts, first edition, 1999.

[MS00] C. D. Manning and H. Schutze. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* Prentice Hall PTR, Upper Saddle River, NJ, 2000.

[MS07] Rosa Mikeal Martey and Jennifer Stromer-Galley. "The Digital Dollhouse : Context and Social Norms in The Sims Onlin." *Games and Culture*, **2**(4):314–334, 2007.

[MSS10] L. Merckena, T.A.B. Snijders, E. Steglichd, E. Vartiainene, and H. de Vriesa. "Dynamics of adolescent friendship networks and smoking behavior." *Social Networks*, **32**:72–81, 2010.

[MW03] James Moody and Douglas R. White. "Social Cohesion and Embeddedness: A hierarchical conception of social groups." *Journal of the American Statistical Association*, **68**(1), 2003.

[NBW06] Mark Newman, Albert-Laszlo Barabasi, and Duncan J. Watts. *The Structure and Dynamics of Networks: (Princeton Studies in Complexity)*. Princeton University Press, USA, 2006.

[net02] S. P. Borgatti. "NetDraw: Graph Visualization Software.", 2002. Retrieved July 2009 `http://www.analytictech.com/downloadnd.htm`.

[New03] M. E. J. Newman. "The Structure and Function of Complex Networks." *SIAM Review*, **45**(2):167–256, 2003.

[New04] M. E. J. Newman. "Detecting community structure in networks." *The European Physical Journal B - Condensed Matter and Complex Systems*, **38**(2):321–330, March 2004.

[New06a] M.E.J. Newman. "Finding community structure in networks using the eigenvectors of matrices." *Phys. Rev. E 74, 036104*, 2006.

[New06b] M.E.J. Newman. "Modularity and community structure in networks." In *Proceedings of the National Academy of Sciences*, volume 103, pp. 8577–8582, 2006.

[Nie10] June2009 June2010 Nielsen NetView. "What Americans Do Online: Social Media And Games Dominate Activity.", 2010. Retrieved May 15 2011 `http://blog.nielsen.com/nielsenwire/online_mobile/what-americans-do-online-social-media-and-games-dominate-activity`

[Ope09] openmetaverse.org. "LibOpenMetaverse.", 2009. Retrieved July 2009 `http://openmetaverse.org/projects/libopenmetaverse`.

[OR07] Jeff Orkin and Deb Roy. "The Restaurant Game: Learning Social Behavior and Language from Thousands of Players Online." *Journal of Game Development*, **3**(1):39–60, 2007.

[PDF05] Gergely Palla, Imre Derenyi, Illes Farkas, and Tamas Vicsek. "Uncovering the overlapping community structure of complex networks in nature and society." *Nature*, **435**(7043):814–818, June 2005.

[PH01] Marius A. Pasca and Sandra M. Harabagiu. "High performance question/answering." In *Proceedings of the 24th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2001.

[Pin94] S. Pinker. *The Language Instinct*. Harper Collins, 1994.

[Pow04] K. Powell. "Brains sniff out scam artists: Evolution may have programmed us to compute fairness.", 2004. Retrieved Feb 2011 `http://www.nature.com/nsu/020812/020812-1.html`.

[Pr07] Nataša Pržulj. "Biological network comparison using graphlet degree distribution." *Bioinformatics*, **23**(2), 2007.

[PSG11] Kyle W. Prier, Matthew S. Smith, Christophe Giraud-Carrier, and Carl L. Hanson. "Identifying health-related topics on twitter: an exploration of tobacco-related tweets as a test topic." In *Proceedings of the 4th international conference on Social comput-*

ing, behavioral-cultural modeling and prediction, SBP'11, pp. 18–25. Springer-Verlag, 2011.

[PSS06] Michael Pearson, Christian Steglich, and Tom Snijders. "Homophily and assimilation among sport-active adolescent substance users." *Connections*, **27**:51–67, 2006.

[PV08] Claudio Pedica and Hannes Vilhjlmsson. "Social Perception and Steering for Online Avatars." In *International Working Conference on Intelligent Virtual Agents*, 2008.

[Qui92] J. Ross Quinlan. "Learning with Continuous Classes." In *Australian Joint Conference on Artificial Intelligence*, pp. 343–348. World Scientific, 1992.

[RCG10] Michal Rosen-Zvi, Chaitanya Chemudugunta, Thomas Griffiths, Padhraic Smyth, and Mark Steyvers. "Learning author-topic models from text corpora." *ACM Transactions on Information Systems (TOIS)*, **28**(1):4:1–4:38, 2010.

[RDL10] Daniel Ramage, Susan Dumais, and Dan Liebling. "Characterizing Microblogs with Topic Models." In *Proceedings of the Fourth International Conference on Weblogs and Social Media*, 2010.

[Rel10] Blizzard Entertainment Inc. Press Releases. "WORLD OF WARCRAFT SUBSCRIBER BASE REACHES 12 MILLION WORLDWIDE.", 2010. Retrieved May 15 2011 `http://us.blizzard.com/en-us/company/press/pressreleases.html?101007`.

[Rep09] Nielson Reports. "World Of Warcraft, Playstation 2 Continue Most Played Gaming Trend.", 2009. Retrieved May 15 2011 http://blog.nielsen.com/nielsenwire/consumer/world-of-warcraft-playstation-2-continue-most-played-gaming-trend

[Rhe07] Howard Rheingold. *Space Time Play*. Springer Publishers, 2007.

[RSW06] Garry Robins, Tom Snijders, Peng Wang, and Mark Handcock. "Recent developments in exponential random graph (p*) models for social networks." *Social Networks*, **29**(3):192–215, 2006.

[SB99] T.A.B Snijders and Stephen P. Borgatti. "Non-parametric standard errors and tests for network statistics." *Connections*, **22**:161–170, 1999.

[SBS09] Fahad Shah, Phil Bell, and Gita Sukthankar. "Identifying User Destinations in Virtual Worlds (poster)." In *International Florida Artificial Intelligence Research Society Conference*, 2009.

[SBS10] T. Snijders, G. van de Bunt, and C. E. G. Steglich. "Introduction to Actor-Based Models for Network Dynamics." *Social Networks*, **32**:44–60, 2010.

[SD97] Tom A.B. Snijders and M.A.J. van Duijn. *Simulation for statistical inference in dynamic network models*. Springer, 1997.

[Sec06] sun.com. "Sun Microsystems To Launch Presence in Virtual World Second Life.", 2006. Retrieved July 2009 http://www.sun.com/aboutsun/media/presskits/secondlife.

[Sec07] techshoutdotcom. "AMD joins the virtual world of Second Life.", 2007. Retrieved July 2009 http://www.thestreet.com/story/10339429/amd-opens-virtual-space.html.

[Sec09a] Princeton University. "Second Life at Princeton University.", 2009. Retrieved April 2009 http://etc.princeton.edu/sl.

[Sec09b] simteach.com. "Second Life: Universities and Private Islands.", 2009. Retrieved July 2009 http://www.simteach.com/wiki/index.php?title=Second_Life:_Universities_and_Private_Islands.

[SGS06] N. Schlaefer, P. Gieselman, and G. Sautter. "The Ephyra QA System at TREC 2006." In *The Fifteenth Text REtrieval Conference (TREC 2006) Proceedings*, 2006.

[SH04] L. Shi and W. Huang. "Apply Social Network Analysis and Data Mining to Dynamic Task Synthesis to Persistent MMORPG Virtual World." In *Proceedings of Intelligent Virtual Agents*, 2004.

[SKB07] N. Schlaefer, J. Ko, J. Betteridge, M. Pathak, E. Nyberg, and G. Sautter. "Semantic Extensions of the Ephyra QA System for TREC 2007." In *The Sixteenth Text REtrieval Conference (TREC 2007) Proceedings*, 2007.

[SKK01] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. "Item-based collaborative filtering recommendation algorithms." In *Proceedings of the 10th International Conference on the World Wide Web*, 2001.

[SM86] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill Inc., 1986.

[Sni95] Tom A.B. Snijders. "Methods for longitudinal social network data." *New Trends in Probability and Statistics*, **3**:211–227, 1995.

[Sni96] Tom A.B. Snijders. "Stochastic actor-oriented models for network change." *Journal of Mathematical Sociology*, **21**:149–172, 1996.

[Sni01a] Tom A. B. Snijders. "The statistical evaluation of social network dynamics." *Sociological Methodology*, **31**:361–395, 2001.

[Sni01b] Tom A.B. Snijders. "The Statistical Evaluation of Social Network Dynamics." *Sociological Methodology*, **31**(1):361–395, 2001.

[Sni05] T. A. B. Snijders. *Models and methods in social network analysis*. Cambridge University Press, New York, 2005.

[Sni06] T.A.B. Snijders. "Statistical Methods for Network Dynamics." In *Proceedings of the XLIII Scientific Meeting, Italian Statistical SocietyInternational Conference on Weblogs and Social Media*, pp. 281–296, 2006.

[Sno01] M.F. Porter. "Snowball: A language for stemming algorithms.", 2001. Retrieved Feb 2011 `http://snowball.tartarus.org/texts/introduction.html`.

[SPR06] T.A.B. Snijders, P.E. Pattison, G.L. Robins, and M.S. Handcock. "New Specifications for Exponential Random Graph Models." *Sociological Methodology*, **36**(1), 2006.

[SR10] Tom A.B. Snijders and Ruth M. Ripley. "Manual for SIENA version 4.0.", 2010.

[SS11] Fahad Shah and Gita Sukthankar. "Constructing Social Networks from Unstructured Group Dialog in Virtual Worlds." In *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction (SBP11)*, 2011.

[SSB10] Samira Shaikh, Tomek Strzalkowski, Aaron Broadwell, Jennifer Stromer-Galley, Sarah Taylor, and Nick Webb. "MPC: A Multi-Party Chat Corpus for Modeling Social Phenomena in Discourse." In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*. European Language Resources Association (ELRA), 2010.

[SSR04a] Mark Steyvers, Padhraic Smyth, Michal Rosen-Zvi, and Thomas Griffiths. "The author-topic model for authors and documents." In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*. AUAI, 2004.

[SSR04b] Mark Steyvers, Padhraic Smyth, Michal Rosen-Zvi, and Thomas Griffiths. "Probabilistic author-topic models for information discovery." In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2004.

[SSS07] T. Snijders, C. E. G. Steglich, and M. Schweinberger. *Longitudinal models in the behavioral and related sciences*. 2007.

[Sur05] James Surowiecki. *The Wisdom of Crowds*. Anchor, 2005.

[SWY75] G. Salton, A. Wong, and C. S. Yang. "A vector space model for automatic indexing." *Communications of the ACM*, **18**:613–620, 1975.

[SYS06] Dou Shen, Qiang Yang, Jian-Tao Sun, and Zheng Chen. "Thread detection in dynamic text message streams." In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pp. 35–42. ACM, 2006.

[Thu03] Crispin Thurlow. "Generation Txt? The sociolinguistics of young people's text-messaging.", 2003. Retrieved Dec 2010 `http://faculty.washington.edu/thurlow/papers/Thurlow(2003)-DAOL.pdf`.

[TL09] Lei Tang and Huan Liu. "Relational learning via latent social dimensions." In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 817–826. ACM, 2009.

[Too07] Lars Toomre. "Second Life and Corporate Virtual Presence.", 2007. Retrieved July 2009 http://www.toomre.com/Second_Life_and_Corporate_Virtual_Presence.

[TT04] Ville H. Tuulos and Henry Tirri. "Combining Topic Models and Social Networks for Chat Data Mining." In *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*, WI '04, pp. 206–213. IEEE Computer Society, 2004.

[Tuc65] Bruce W. Tuckman. "Developmental sequence in small group." *Psychological Bulletin*, **63**(6):384–399, 1965.

[WDX06] Dmitri Williams, Nicolas Ducheneaut, Li Xiong, Yuanyuan Zhang, Nick Yee, and Eric Nickell. "From Tree House to Barracks: The Social Life of Guilds in World of Warcraft." *Games and Culture*, **1**(4):338–361, 2006.

[Wek09] Univeristy of Waikato. "Weka.", 2009. Retrieved July 2009 http://www.cs.waikato.ac.nz/ml/weka/.

[WJC08] Yi-Chia Wang, Mahesh Joshi, William Cohen, and Carolyn Ros. "Recovering Implicit Thread Structure in Newsgroup Style Conversations." In *International Conference on Weblogs and Social Media (ICWSM)*, 2008.

[WO09] Lidan Wang and Douglas W. Oard. "Context-based message expansion for disentanglement of interleaved text conversations." In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association*

for Computational Linguistics, NAACL '09, pp. 200–208. Association for Computational Linguistics, 2009.

[Wor09] George A. Miller et. al. "WordNet.", 2009. Retrieved July 2009 `http://wordnet.princeton.edu/wordnet/`.

[WP96] S. Wasserman and P. Pattison. "Logit models and logistic regressions for social networks: I. An introduction to Markov graphs and p." *Psychometrika*, **61**(3), 1996.

[WTR08] Erik Weitnauer, Nick M. Thomas, Felix Rabe, and Stefan Kopp. "Intelligent Agents Living in Social Virtual Environments: Bringing Max into Second Life." In *International Working Conference on Intelligent Virtual Agents*, 2008.

[YBU07] N. Yee, J.N. Bailenson, M. Urbanek, F. Chang, and D. Merget. "The unbearable likeness of being digital: the persistence of nonverbal social norms in online virtual environments." *Cyberpsychol Behavior*, **10**(1):115–121, 2007.

[Yee05] Nick Yee. "WoW Gender-Bending.", 2005. Retrieved May 15 2011 `http://www.nickyee.com/daedalus/archives/001369.php`.

[Yee08] Nick Yee. "Our Virtual Bodies, Ourselves?", 2008. Retrieved May 15 2011 `http://www.nickyee.com/daedalus/archives/001613.php`.

[Yok10] T. Yokosuka. "Second Life Statistics.", 2010.

[ZMB08] B. Ziebart, A. Maas, J. Bagnell, and A. Dey. "Maximum Entropy Inverse Reinforcement Learning." In *Proceedings of National Conference on Artificial Intelligence*, 2008.