Confidentiality and Integrity Management in Online Systems

by

Amin Ranj Bar

School of Computer Science

McGill University, Montreal, Canada

January 2013

A Thesis Submitted to McGill University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

© Amin Ranj Bar, January 2013

This document is dedicated to my parents and my love.

Acknowledgement

I would like to take this opportunity to express my sincere appreciation and gratitude to my supervisor, Professor Muthucumaru Maheswaran who has enlightened and guided me throughout my doctoral studies. His support has always been generous and timely. It was his invaluable advice that made me accomplish this achievement. Nothing can be said enough to express my indebtedness to him.

My appreciation also goes to my colleagues in ANRL for their help and support. In particular, I would like to thank Arash Nourian for helping me during my studies at McGill. Also, many thanks to Roxane Lefebvre-Labelle and Aryan Bayani for helping me in translating the abstract to French.

I would also like to thank all the members of School of Computer Science for creating such a dynamic and collaborative environment for research and study.

Last, but not least, I would like to express my warmest and deepest thanks to my parents and my love for their self-giving love and support through the duration of my studies. Without them, I would never be able to complete my study at McGill University.

Abstract

The dominant role of social networking in the web is turning human relations into conduits of information flow. This means that the way information spreads on the web is determined to a large extent by human decisions. Consequently, information security, confidentiality and integrity of shared data, relies on the quality of the collective decisions made by the users. Recently, many access control schemes have been proposed to control unauthorized propagation and modification of information in online systems; however, there is still a need for mechanisms to evaluate the risk of information leakage and unauthorized modifications within online systems. First, the thesis focuses on the confidentiality of information in online social networks. A novel community-centric confidentiality control mechanism for information flow management on the social web is presented. A Monte Carlo based algorithm is developed to determine the potential spread of a shared data object and to inform the user of the risk of information leakage associated with different sharing decisions she can make in a social network. The scheme also provides a facility to reduce information flowing to a specific user (i.e., black listing a specific user). Second the thesis focuses on the integrity of artifacts in crowdsourcing systems. A new approach for managing the integrity of contents created in crowdsourcing repositories named Social Integrity Management (SIM) is presented. SIM integrates two conflicting approaches to manage integrity in crowdsourcing systems: owner-centric and owner-free schemes. The ownership bottleneck

is relaxed by including co-ownerships and having multiple versions. Finally, the thesis presents a thorough analysis of the Stack Exchange sites as an example of widely used crowdsourcing question answering systems. The dump datasets are used to analyze various user behaviors in crowdsourcing question answering systems by considering the effect of tagging, user reputation and user feedback. Observed characteristics from the studies are used in the modeling and evaluation of social integrity management.

Résumé

Le rôle prépondérant des réseaux sociaux sur le web change les relations humaines en conduits d'échange d'information. Ainsi, l'information qui est véhiculée sur le web est déterminée en grande partie par les prises de décisions humaines. Conséquemment, la sécurité de l'information, la confidentialité et lintégrité de l'information partagée dépendent de la qualité des décisions prises collectivement par les utilisateurs. Récemment, plusieurs schémas de contrôle daccès ont été proposés pour contrôler la propagation non autorisée et la modification de l'information dans les systèmes en ligne. Par contre, il y a encore un besoin de mécanismes dévaluation des risques de fuites d'information et de modifications non autorisées à l'intérieur des systèmes en ligne. Premièrement, la thèse se concentre sur la confidentialité de l'information dans les réseaux sociaux en ligne. Un nouveau mécanisme de contrôle de la confidentialité axé sur la communité pour la gestion de circulation de l'information est présenté. Un algorithme basé sur le modèle Monte Carlo est développé pour déterminer la possibilité de la diffusion des éléments de données partagés ainsi que pour informer l'utilisateur des risques de fuite d'information associés aux différentes décisions de partage que l'utilisateur pourra faire dans un réseau social. Le schéma fournit également une installation pour réduire l'échappement de l'information à un utilisateur spécifique (ex. mettre un utilisateur sur une liste noire). Deuxièmement, la thèse se concentre sur l'intégrité des objets des systèmes de crowdsourcing. Une approche

nouvelle pour gérer l'intégrité du contenu créé par les archives de crowdsourcing appelé Gestion de l'intégrité sociale (Social Integrity Management) est présentée. Cette approche intgre deux approches contradictoires pour gérer l'intégrité des systèmes de crowdsourcing: les schémas basés sur le propriétaire et les schémas sans propriétaires. La problématique de la propriété est détendue en incluant la copropriété et la possibilité d'avoir plusieurs versions. Finalement, la thèse présente une analyse complète des sites d'échange comme exemple de système de réponses aux questions par le crowdsourcing qui sont grandement utilisés. L'ensemble de données déchargées est utilisé pour analyser le comportement de différents utilisateurs dans les systèmes de réponses aux questions basés sur le crowdsourcing en considérant les effets d'étiquetage, la réputation des utilisateurs et les commentaires des utilisateurs. Les caractéristiques observées par les études sont utilisées dans la simulation et l'évaluation de la gérance de l'intégrité sociale.

Contents

List of Figures

List of Tables

List of Algorithms

1

Introduction

1.1 Overview

Online social networks (OSNs) can be categorized into three major categories: friendship networks, common interest networks and interaction networks. In friendship networks, people create profile pages that describe themselves and that explicitly link them to their friends' profiles. There are many OSNs that fall into this category such as Facebook, MySpace, and LinkedIn. In common interest networks, social networks emerge as users collaboratively create some online content such as photo albums, wikis, and blogs. In interaction networks, social networks are defined by the communication patterns used in instant messaging services or email. OSNs are dynamic networks with topological changes that are caused by edge and node creations and deletions.

As online social networks (Facebook counts more than one billion users) increase in size and more people use them as their primary Internet portal, the volume of information shared in OSNs keeps on growing. Information is created by different sources in OSNs including people posting information in their profile pages, relational information generated by people initiating connections among themselves, and data feeds generated by sensing people's activities such as gaming and purchasing. In any sharing activity, OSNs store

and process different pieces of information: picture files, relationships among people, and sharing preferences regarding data objects. This means that the way OSNs are architected and the security primitives built into them play key roles in defining information security in the social web. Consequently, many research thrusts have examined wide-ranging security issues in the context of OSNs [1, 2, 3, 4, 5, 6, 7, 8].

While information sharing is vital for socializing online, many security and privacy issues have been raised such as confidentiality and integrity violations of shared data objects. The main issue is to ensure users that their privacy and access control requirements are preserved when information sharing occurs within OSNs. Recently, users in OSNs started to become more aware of the risk of unauthorized propagation of their information through social networking sites. To partially answer users concern, several topology-based access control mechanisms for OSNs were proposed in order to identify authorized users by specifying some constraints on the social graph [9, 3, 10, 11, 12, 13]. In these schemes, to regulate information sharing, access control rules are defined by identifying the relationships that users must have in order to access the shared data.

Because existing techniques only deal with information release, a user might not be able to precisely identify who is authorized to have access to her data. Even in small social networks, it is difficult for a user to keep track of the topology of her constantly changing social network and to identify users who are actually authorized even with simple access rules such as "friends-of-friends." In addition, the user's privacy requirements are constantly changing [14, 15, 16]. Users can lose control of their shared data and risks of unauthorized dissemination of their data escalates with increasing number of social interactions [17]. Specially, a user may not be able to track how her private information is handled by her friends after she has released the information to them [17]. The topology

based access control mechanisms give a static control scheme based on particular friendship configurations. Therefore, it is necessary to have new access control mechanisms in OSNs in order to evaluate the potential risks and to make users fully aware of the possible consequences of their decisions in specifying access rules.

This thesis research takes a complementary approach to address the challenges identified above by introducing a novel community-centric confidentiality management for OSNs. With the assumption that usage control is hard in OSNs, this work focuses in developing a new strategy where the eventual information distribution is shaped by the initial release of objects into the network. Because initial release is completely controlled by the owner, she could shape the information distribution by making appropriate release decisions to preserve the confidentiality of the shared information.

The second part of this thesis research focuses on the integrity problem in crowdsourcing systems. Crowdsourcing is a powerful approach for building information artifacts used in popular systems such as Wikipedia and Linux on the Internet [18, 19]. To meet its integrity objectives, Wikipedia encourages contributions by making it easy for the contributors to create and update any article. The integrity of the contributions are checked and flagged by subsequent readers. For highly trafficked articles, this model of integrity enforcement works very well. In the Linux kernel, integrity is given very high priority. All updates submitted by the development community need the final approval of the project originator (Linus Torvalds) before they are included in the official software release. Community feedback and importance of the contribution are some of the factors that can influence Linus Torvalds' decision to include or exclude the contribution.

Several Wikipedia-like projects that do not have the popularity of Wikipedia use a

model similar to Linux to manage the integrity of the articles maintained by them. However, instead of relying on a single person for the whole project, these sites [20, 21] decentralize the integrity management task such that an article creator is responsible for accepting or rejecting community updates received on topics within the scope of the article. While having a central figure per article facilitates integrity maintenance, it can create lots of workload for the maintainer if there is a large number of small updates from the community.

Mainly, there are three problems to preserve integrity of articles on online crowdsourcing systems. The leading problem is the lack of authority in large-scale collaborative content sharing websites such as Wikipedia [22]. For instance, readers of Wikipedia cannot know who has written or modified the article they are reading, it may or may not have been written by an expert. The second problem is the lack of content verification on specialized topics. Someone should report the problem; otherwise, inaccurate information that is not obviously false may exist in Wikipedia for a long time before it is challenged [23, 24]. Lack of fact checking may result in biased articles with various contentions that will need further resolutions. Most of the solutions to these problems were propositions to have multiple versions and the ownership. Although these techniques are essential for resolving contentions, they also presents a third challenge which is the duplication of effort. Users would have difficulties in discovering relevant artifacts in the existing ones. To address problems observed above, we propose Social Integrity Management (SIM), a new approach for managing the integrity of contents created in crowdsourcing repositories. In SIM, existing online social networks are leveraged to determine the trustworthiness of users. The design of SIM enjoys the benefits of the two existing styles, i.e., any user can be a potential writer to create a new article (Wikipedia Style) while the integrity is enforced using ownership (Linux style). The

bottleneck created by ownership is resolved in SIM by including co-ownerships and having multiple versions. Observed characteristics from the studies done in Chapter 5 are used in the modeling and evaluation of social integrity management.

Finally, in the last part of the thesis, we present a thorough analysis of the StackExchange sites that provide question answering on topics ranging from programming to cooking. StackExchange sites provide free services to its users, where the answerers volunteer their time by contributing answers. In return, the answerers gain reputation as they provide acceptable answers. Question answering is a fundamental pattern that exists in many computer based systems. Crowdsourced question answering systems use human intelligence to obtain the information the questioner is seeking. In recent years, many crowdsourced question answering systems have emerged on the Internet and there has been a lot of research on the measurement and analysis of Question Answering sites [25, 26, 27, 28, 29].

More investigation into understanding the characteristics of such sites is necessary to evaluate the effectiveness of current systems and to design future systems. We present our analysis on user behavior in crowdsourced question answering systems by considering the effect of tagging, user reputation and user feedbacks. We also conducted a survey among actual users of the StackExchange sites. The survey results support some of the observations we made in data analysis.

1.2 Thesis Contributions

The research work undertaken as part of this thesis makes the following three groups of contributions.

First, for the community-centric confidentiality control mechanism presented in Chapter 3, the major contributions are:

- Proposed a novel community-centric confidentiality control scheme for online social networks based on the risk of information leakage involved in the sharing activity.
- Developed a Monte Carlo based model for computing the set of potential users who could receive the data objects belonging to a data owner and provided algorithms for preventing information from reaching certain users by shaping the initial release set.
- Analyzed different sharing situations and estimated information leakage values considering that our algorithms controlled information sharing.

Second, the major contributions for the social integrity management presented in Chapter 4 are:

- Analyzed integrity management in Wikipedia and examined the role of contributors in an article becoming a featured one.
- Proposed a social integrity management scheme for preserving integrity in online crowdsourcing systems based on the observed characteristics from the studies done in Chapter 5.
- Analyzed the effects of the social network structure on the features of the proposed scheme.

Finally, the main contributions of analysis provided in Chapter 5 are:

- Analyzed one of the widely used collection of question answering sites.
- Characterized user behavior in crowd sourced question answering systems by considering the effect of tagging, user reputation and user feedback.
- Conducted a survey among users of the question answering sites to collect their feedback in order to support our results.

1.3 Organization of the Thesis

The rest of this thesis is organized as follows. In Chapter 2, we provide some background discussion on the current state of online social networks, crowdsourcing systems, privacy and security challenges in online systems, access control, and the confidentiality and integrity management problem. Chapter 3 defines the secure information sharing problem for online social networks and presents a detailed design and analysis of the community-centric confidentiality control mechanism for online systems. In Chapter 4, an overview of the integrity management problem and the challenges facing online crowdsourcing systems are provided. Our proposal for social integrity management is presented in this chapter with a system design, theoretical analysis, and simulation studies of various elements of the overall system. Chapter 5 provides details of our study on user behavior in a crowdsourced question answering system. Chapter 6 gives an overview of the existing literature relevant to the problems we addresses in this research. A summary of the thesis with the important contributions is presented in Chapter 7. Possible future extensions of our research are also briefly indicated in the same chapter.

2

Background Material

2.1 Overview

In this chapter, we provide background information on topics related to the problems addressed in this thesis. We discuss general social networks and web-based social networks in Section 2.2 and Section 2.3. Section 2.4 discusses security challenges in online systems. In Section 2.5, we review the access control problem in online social networking services. Background information on the crowdsourcing systems is provided in Section 2.6. Finally, Section 2.7 reviews the integrity management problem in crowdsourcing systems.

2.2 Social Networks

Social networks are social structures that consist of social entities (e.g., individuals, groups, organizations) that are connected to one another by social relationships [30]. Social relationships can be quite broad; examples include friendships, behavioral interactions, biological relationships, or affiliations.

Social network analysis focuses on studying the different patterns of relationships among

social entities along with their implications on the behavior and decisions of the social entities [31]. Based on concepts used in graph theory, a social network is represented by a graph consisting of a set of nodes and edges. The nodes in the graph represent the social entities, while the edges represent the social ties that link those entities. The resulting graph structures are often complex where the social entities are considered interdependent rather than independent units. This means that in social network analysis the discrete unit of analysis is the combination of social entities and the relationships among them.

Social network analysis has been widely used in recent decades in such diverse areas as sociology, anthropology, biology, economics, and information science [32, 33, 34, 35, 36, 37]. For example, in the area of epidemiology, social network analysis has been used to study the effect of different patterns of social contacts on the spread of human diseases and viruses [32], and also to study the relationship between social and community ties and mortality among people [33]. In the field of sociology, social network analysis played an important role in understanding how information spreads on social networks [34] and how individuals are connected in the physical world [35, 36]. In economics, the influence of social structures on the outcomes of the labor market are analyzed using the tools of social network analysis [37].

The last few years have witnessed the emergence of the second generation of the world wide web, i.e., "Web 2.0." Facilitating online collaboration and information sharing for people, Web 2.0 has enabled the development and evolution of online based social networks (communities). This has resulted in an increased use of social network analysis to study the underlying structures of these communities and address the problems and challenges that arise within these online systems. Due to their importance, we introduce online social networks and discuss the different properties associated with them in the next section.

2.3 Online Social Networks

Online social networks are online communities of individuals who share common interests or activities. The majority of these online communities develop on different web-sites that offer different means for their users to interact and socialize. Boyd [38] defines today's social networking web sites as "web-based services that allow individuals to (1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system."

While the definition for social networking websites provided in [38] presents those sites as being mainly profile-based, there exist many social networking sites that offer other types of services. The research report in [39] attempts to categorize the different social networking services that exist today. It identifies eight main types, among which are the popular profile-based social networks like `myspace.com` and `facebook.com`. Content-based social networks are also among the most popular sites, where the main form of interactions and relationships between users are established through the creation of user content. Examples of such sites include `flickr.com`, a photo-sharing site, `youtube.com`, a site for sharing user created videos, and `delicious.com`, a social bookmarking and tagging site. In addition, other social networks provide micro-blogging services, where the users post status messages allowing other people on their social network to track their status; an example of such a service is `twitter.com`. What makes these social networking sites interesting is that they eliminate the physical limitations of the traditional social networks, allowing their participants to extend and build their personal social networks by meeting new individuals from across the globe. As a result, we are witnessing the rise of new and different relationship structures that are not related to the offline world.

Some suggest that online social networking can be traced back to 1997 with the launch of the first blogging site [40] and social networking website `sixdegrees.com` [38]. Since then, the number of social networking sites has increased dramatically, attracting many users and generating high web traffic. Based on the information provided by Alexa (`alexa.com` – a database of information about sites that includes various statistics), many of the existing social networking web sites are ranked in the top 500 web sites in terms of traffic generated on the web [41]. Reports from Nielsen Online [42], a company that provides measurement and analysis of online audiences, indicates that nearly half of the biggest social networking sites are also among the fastest growing, with still room for potential future growth.

2.4 Privacy and Security Challenges

The popularity of social networking services has attracted the attention of researchers due to the various privacy and security risks involved. The success of any social networking site can be judged based on the number of its participants and the size of the activities taking place on the site. Therefore, these websites are always competing to come up with new services and designs that would make them more appealing to their users. Unlike the physical world, people participating in online social networking services are, in certain cases, willing to form relationships with others they know little about, and thus providing strangers access to their private and sensitive information.

A study in [43] examined the patterns of information revelation and the usage of privacy settings in Facebook to show that the users appear unconcerned about the privacy risks associated with OSNs. The study shows that while personal data is generally provided by the users, adjusting the privacy preferences to limit the access to personal data is rarely

used. This is a major cause for concern, since the information revealed online by users can expose them to various physical and cyber risks.

In [44], authors highlight a comprehensive set of privacy-, identity-, and social-related security risks associated with today's social networks. Examples of the privacy-related risks include digital dossier aggregations of personal data from third parties, secondary data collection of data not included in user profiles (e.g., members disclose information related to length of connections, other user profiles visited and messages sent), and linkability of user profiles from image metadata. In identity-related risks, social network phishing and identity theft are big threats. Social network phishing is a phishing attack targeted at social network users facilitated by the easily accessible user profiles available on those networks. In identity theft, fake profiles are created based on other existing identities in order to benefit from their reputation or otherwise slander people's reputation.

In addition to the privacy and identity related security concerns, security issues related to confidentiality and integrity of user-created content are equally important. A key problem related to the confidentiality and integrity of user data is online sharing of this information. Users participating in OSNs use various applications provided on these sites to create and share public or private content for personal or professional purposes. While information sharing is vital for socializing online, many challenges have been raised because of the unregulated sharing situations in OSNs.

The primary challenge for information sharing in OSNs is the impreciseness of the problem itself. In a typical corporate computing system, information sharing is dictated by the overall organizational policies, which are formulated based on the corporate agenda. The information sharing problem in OSNs, however, is not governed by a precise policy. The need to socialize in OSNs dictates that users should share information. However, the

privacy concerns can reduce the overall information spread in OSNs. Another challenge is the diverse user populations in OSNs. It is generally accepted that users in OSNs desire to have effortless ways of controlling information sharing. While retaining simplicity, users want mechanisms that minimize unintentional release of data. Yet another challenge is the mismatch of goals among the different stakeholders of OSNs. The OSN operators want unhindered information flow so that they could extract sufficient business intelligence on the user population. The users, on the other hand, want to control information flow to suit their needs and privacy preferences. This means that the control of the information sharing should balance the need for privacy with the need for publicity.

Due to its importance, there has been a number of proposed solutions to the problem of online information sharing on OSNs based on different access control mechanisms. Since this research focuses on the problem of preserving the confidentiality and integrity of shared data, I believe it is worthwhile to review the major access control techniques along with some of the solutions treating the problem of online sharing.

2.5 Access Control in Online Social Networks

Confidentiality is defined in [45] as "ensuring that information is accessible only to those authorized to have access". Similarly, Bishop describes integrity as "preventing improper or unauthorized modifications" [45]. Access control mechanisms are an essential part of information security because they provide the necessary means for preserving information confidentiality and integrity. Today, due to the popularity of online information sharing, there is a pressing need for new access control techniques that provide a secure environment for online information sharing. In this section, we first give a brief introduction to the major access control techniques used in security systems then, we review some of the recent work

addressing the problem of information sharing in OSNs.

2.5.1 Major access control techniques

There are three major access control policies that have emerged since the 1970s: Discretionary Access Control (DAC) [46], Mandatory Access Control (MAC) [47, 48], and more recently, Role Based Access Control (RBAC) [49]. These access control policies are the among the most commonly used in computer systems.

1. Discretionary Access Control (DAC), is an owner-centric based policy where the owner of the protected data dictates the different access policies for the data. Many of the implemented access control policies are related to DAC in some form or another [50]. DAC consists of three main entities: the protected objects, subjects, and access rights. In the system, each object is assigned an owner who is initially the creator of the object. The owner of an object has complete control over the access rights and permissions assigned to other subjects in the system. Subjects are granted access only if the access rights authorize them to perform the requested operation.

   Advantages of DAC is its simplicity, flexibility, and ease of implementation [50]. The main drawback of DAC policies is that the access restrictions can be easily bypassed [51]. A subject who has been granted access can easily pass the object to non-authorized subjects without the owner's knowledge. This is because there is no restriction imposed upon the dissemination of information once a subject has gained access. DAC is usually implemented by Access control list (ACL) and capability based access control systems [51].

2. Mandatory Access Control (MAC), is based on the security classifications of subjects

and objects in the system. In MAC, subjects are assigned security classification (or clearance) that corresponds to the trustworthiness of that entity, while the security sensitivity label of an object corresponds to the trust level required for the subjects in order to have access. In contrast to DAC, object owners do not make policy access decisions or assign security attributes [50]. Access is granted only if a subject has the necessary clearance to access an object. MAC aims to enforce lattice-based information flow constraints to establish high assurance information systems [52].

This is usually achieved through the following two principles [51]:

- **Read down**: A subject's clearance must dominate the security level of the object being read
- **Write up**: A subject's clearance must be dominated by the security level of the object being written

The rules above ensures a one way flow of information. In MAC some degree of centralization exists. Typically, a security policy administrator is responsible for maintaining the security levels of subjects or objects. The main disadvantage of MAC is its rigidity and the centralized architecture makes it difficult to adapt it for distributed systems.

3. Role-Based Access Control (RBAC), has emerged in the past decade as the most widely discussed alternative to DAC and MAC. RBAC assigns permissions to well-defined abstractions called roles. Roles can be defined as a set of actions and responsibilities associated with a particular working activity [51]. Users then take on different roles in order to gain access to protected objects. RBAC allows a user to attain different permissions by switching to different roles in a given session [53].

RBAC makes the process of managing access rights easier by splitting the task of user authorizations into two parts namely, the assignment of access rights to roles and the assignment of roles to users. This makes the assigning and revocation of access rights a convenient task. An advantage of RBAC is that it enables the creation of a hierarchy of roles which makes it appealing to many highly structured systems. However, context is not fully considered in the activation, deactivation, and management of roles.

The access control policies discussed above are the most widely used policies in many security systems today. However, in the context of OSNs, where the social relationships play an important role in shaping the access policies, applying traditional access control techniques to deal with the information sharing problem is not a trivial task. Still, there have been a number of studies developing different social networking based access control techniques. We review some of those papers next.

2.5.2 Access control for online social networks

Sharing of personal data is considered a major issue on OSNs and content sharing systems. Although some social networking sites, like Facebook, Flickr, and Google Knol, have started to incorporate basic access control features into their sites, these controls are often limited and incomplete. Recently, a number of social networking based access control models have been developed to address the sharing problem.

The work in [54, 2] presents a social networking based access control scheme suitable for online information sharing. In the proposed approach, users identities are established through key pairs. Social relationship between users are represented as social attestations issued from one user to another and are used to gain access to friends personal content.

Access control lists (ACLs) are employed to define the access rights for the users based on the social relationships. To gain access to a particular object, a person must hold an attestation that satisfies the access policies specified for the object.

Another model has been proposed in [55]. The authors introduce a rule-based access control mechanism for web-based social networks. The approach is based on the enforcement of complex policies expressed as constraints on the type, depth, and trust level of relationships existing between users. The model makes use of certificates to grant relationships authenticity. The authors propose the use of client-side enforcement of access control according to a rule-based approach, where a subject requesting to access an object must demonstrate that it has the rights of doing that.

In [56], Villegas *et al.* present (PDAC), a personal data access control scheme for protecting personal data stored online. PDAC introduces a trusted distance measure that is computed based on the social network structural information and the experiential data of the users. Using the trusted distance, a data object owner defines three protection zones. PDAC uses a collaborative computing approach to map other users to the data protection zones defined by the owner of a data object. Based on the zone a user is mapped into, her requests to access the data objects of another user will be accepted, attested, or rejected. Attestation involves another round of evaluation by attesters designated by the owner of the data object.

In addition to the above, an ongoing research project is represented by PLOG [57]. The goal of PLOG is to facilitate access control that is automatic, expressive and convenient. The authors are interested in exploring content based access control to be applied in social networking sites. Furthermore, Relationship-Based Access Control (ReBAC) was proposed in [6] to express the access control policies in terms of interpersonal relationships

between users. ReBAC captures the contextual nature of relationships in OSNs. Relationships are articulated in contexts, and accesses are authorized also in contexts. Sharing of relationships among contexts is achieved in a rational manner through a context hierarchy. The authors also present a policy language based on modal logic in order to express ReBAC policies. The language provides means for composing complex policies from simple ones.

2.5.3 Access control implementations in online social networks

Most of the social networking services (e.g., Facebook, Google Knol, Flickr, etc.) incorporate basic access control features into their sites. The protection mechanisms implemented in most of the social networking sites are often very limited, allowing their users to set the confidentiality and integrity level for a given item as public, private, or accessible to friends. Certain social networking sites have incorporated variants of this simple protection scheme in order to provide their users with more flexibility in specifying the confidentiality and integrity level.

For example, in addition to the basic access control options, Bebo (`bebo.com`) and Facebook (`facebook.com`) support friend-of-friend (2nd degree contacts) and "customized" (i.e., selected friend) access control options. Orkut (`orkut.com`) provides support for friends-of-friends. Myspace (`myspace.com`) and Google Knol (`knol.google.com`) only support the basic options of public, private and 1st degree contacts. LinkedIn (`linkedin.com`) supports the option "my network," which is defined as the user's network of trusted professionals and includes 1st degree, 2nd degree, and 3rd degree connections in addition to members belonging to the user's LinkedIn groups. Flickr (`flickr.com`) supports public, private, and 1st degree connections (friends, family, or both).

It is important to note that all these simple access control schemes mentioned above,

or variations of them, have several drawbacks. Many of these drawbacks stem from the following assumptions made by the simple schemes that are not always applicable.

1. **All friends are equal**. Access control schemes that use the hop distance to categorize friends assume that all friends at a particular hop distance are equal. While this makes access control simple, it lacks the flexibility of differentiating among the friends who are at the same distance when setting access options.

2. **Omniscient users**. Access control schemes that expect users to define access control lists to explicitly deny and allow accesses to data items assume that users are all knowing about their social neighbourhood and able to make the appropriate protection decisions. With large and dynamic social neighbourhoods (friends and friend-of-friends) such an assumption is not practical.

3. **No impact on friendships**. The simple hop-based protection scheme assumes that access control decisions do not impact the topology of the social network, which is often not valid. If Alice is unwilling to provide access to Trudy for data that she is already sharing with Bob, then it can indicate a lack of trust on Trudy.

4. **Friendships are static**. Because the nature of friendships can change from time to time, this can impact the owner's desire for certain users to access his or her data at certain times. In order to deal with this scenario, access control schemes utilizing only hop distance would require breaking and restoring the friendship link. Access control lists would require users to add and remove specific peers from each list; therefore it is a hard task for users.

In addition to these points, access control mechanisms used or developed for online social networking systems focus on protecting information resources within the system. In

online social networks, information is often shared with other users for different purposes. Access controls techniques on social networks attempt to guarantee the protection of the information stored within the network, which means that once information is shared, those who gain access are able to use the content in anyway they want. Therefore in online social networks, there is a need for new access control schemes where the eventual information distribution is shaped by the initial release of the data items into the network.

2.6 Crowdsourcing Systems

Crowdsourcing systems enlist a crowd of users to collaborate to build a wide variety of artifacts. Over the past decade, a large number of crowdsourcing systems have been proposed on the Internet such as Wikipedia, Linux, Yahoo! Answer, Stack Exchange and much effort is being directed toward developing many more. Since this effort is an emerging area, it has appeared under many names, including peer production [58], user-powered systems [59], user-generated content [59], collaborative systems [60], community systems [61], collective intelligence [62], wikinomics [63], crowd wisdom [64], smart mobs [65], mass collaboration [66], and human computation [59].

Crowdsourcing systems can be classified in many different contexts. One of the classifications can be the nature of collaboration, explicit or implicit. Explicit collaboration systems (e.g., Wikipedia or Linux) allow users to collaborate explicitly to build artifacts. On the other hand, implicit collaboration systems let users collaborate implicitly to solve a problem of the system owners. For instance, the ESP game [67] allows users to implicitly collaborate to label images as a side effect while playing the game.

A second type of classification is based on the type of the target problem. The target problem can be any problem defined by the system owners, from building temporary or

permanent artifacts to executing tasks. Another dimension can be the degree of manual effort. When building a crowdsourcing system, the system owners must decide how much manual effort is required to maintain the system. This can range from relatively little (e.g., combining ratings) to substantial (e.g., combining code), and also depends on how much the system is automated. The system owners must decide how to divide the manual effort between the users and themselves. Some systems ask the users to do relatively little and the owners a great deal. For instance, to detect malicious users, the users may simply click a button to report suspicious behaviors, whereas the owners must carefully examine all relevant evidence to determine if a user is indeed malicious. Some systems do the reverse. For example, most of the manual burden of merging Wikipedia edits falls on the users who are currently editing, not the owner.

The other criteria can be the role of human users. Here, we can consider four basic roles for humans in a crowdsourcing system. *Slaves*: humans help solving the problem in a divide-and-conquer fashion, and minimizing the resources (e.g., time, effort) of the owners. Examples are ESP and finding a missing boat in satellite images using Mechanical Turk-based systems [68]. *Perspective providers*: humans contribute different perspectives, which when combined often produce a better solution than with a single human. Examples are reviewing books and aggregating user bets to make predictions [64]. *Content providers:* humans contribute self-generated content (e.g., videos on YouTube or images on Flickr). *Component providers:* humans function as components in the target artifact, such as a social network, or simply just a community of users (e.g., the owner can sell ads). Humans often play multiple roles within a single crowdsourcing system (for example, slaves, perspective providers, and content providers in Wikipedia) [59].

2.6.1 Sample crowdsourcing systems

In this section, we focus on introducing the most widely used crowdsourcing systems by categorizing them in four different groups: collective knowledge management, collective creativity, collaborative gaming and collaborative voting. We provide definitions and descriptions for each category.

Collective knowledge management

This type of systems allows users to build artifacts, often merges user inputs tightly, and requires users to edit and merge one another's inputs. A well-known artifact is textual knowledge bases (KBs). To build such KBs, users contribute data such as sentences, paragraphs, Web pages, then edit and merge one another's contributions. The two main examples of knowledge management crowdsourcing systems are Wikipedia and Yahoo! Answers.

Wikipedia is an online encyclopedia that is freely available online. The notion of open editing in Wikipedia encourages many people to collaborate in a distributed manner to create and maintain a repository of information artifacts. Wikipedia has more than 17 million registered authors and more than four million articles [69]. It has become a valuable resource and many people cite it as a credible information source. However, the open process that provides popularity to Wikipedia makes it difficult for readers to be sure about the reliability of the content. Similar to other crowdsourcing systems, Wikipedia articles are constantly changed by contributors who can be nonexpert or even vandals. On the other hand, Yahoo! Answers is a general question-answering forum to provide automated collection of human reviewed data at Internet-scale. These human-reviewed data are often required by enterprise and web data processing.

Collective creativity

The role of human in creativity cannot be replaced by any advanced technologies. The creative tasks, such as drawing and coding, can only be done by humans. Here, crowdsourcing is used to tap into online communities of thousands of users to develop original products and concepts, including photography, advertising, film, video production, graphic design, apparel, consumer goods, branding concepts and different software. As a result, some researchers sought for crowdsourcing users to do some creative tasks to reduce the production costs. An example is the Sheep Market. The Sheep Market is a web-based artwork to implicate thousands of online workers in the creation of a massive database of drawings. It is a collection of 10,000 sheep created by MTurk workers, and each worker was paid US$0.02 to draw a sheep facing left [70, 71].

Another example is Threadless which is a platform of collecting graphic T-shirt designs created by the community [72]. Although technology advances rapidly nowadays, humans can innovate creative ideas in a product design process but computers cannot. A computer has no clue about how to solve a specific problem for developing a new product. Different individuals may create different ideas such as designing a T-shirt [72]. Moreover, Leimeister [73] proposed crowdsourcing software development tasks as ideas for competitions to motivate more users to support and participate. Well-known software such as Apache, Linux, Hadoop was produced and maintained by crowdsourcing systems.

Collaborative gaming

The concept of "Social Game" was pioneered by Luis Von Ahn and his colleagues, who created games with a purpose [74]. The games produce useful metadata as a by-product. By taking advantage of people's desire to be entertained, problems can be solved efficiently by

online game players. The online ESP Game [67] was the first human computation system, and it was subsequently adopted as the Google Image Labeler. Its objective is to collect labels for images on Web. In addition to image annotation, the Peekaboom system [75] can help determine the location of objects in images, and provide complete outlines of the objects in an image. The concept of the ESP Game has been applied to other problems. For instance, the TagATune system [76], MajorMiner [77] and TheListen Game [78] provide annotation for sounds and music which can improve audio searches.

Collaborative voting

In this type of crowdsourcing systems, a user is required to select an answer from a number of choices. The answer that the majority selected is considered to be correct. Voting can be used as a tool to evaluate the correctness of an answer from the crowd. An example of popular crowdsourcing websites with collaborative voting is Amazon Mechanical Turk (or MTurk) [79]. A large number of applications or experiments were conducted on Amazon's MTurk site. It can support a large number of voting tasks.

2.7 Integrity Management in Crowdsourcing Systems

Integrity management is the review or establishment of different mechanisms to ensure the long-term integrity of artifacts in crowdsourcing systems. Crowdsourcing systems deal with two main challenges in order to preserve the integrity of their artifacts: assigning different capabilities to users, and evaluating users and their contributions.

To assign different capabilities to users, crowdsourcing systems often classify users into different groups, such as guests, regulars, editors, admins, and "dictators". Low-ranking users (e.g., guests, regulars) usually have few capabilities: answering questions, editing

small part of artifacts, or flagging an incorrect data piece. On the other hand, high-ranking users (e.g., editors, admins) have a wide variety of capabilities, from small contributions to resolving controversial issues. This type of user classification is necessary in order to control the impact of a contribution. The potential impact of a contribution can be measured by considering how the contribution potentially affects the crowdsourcing system. For instance, editing a sentence in a Wikipedia page can only affect that page, whereas revising a code in a software such as Linux may potentially affect millions of users. Quantifying the potential impact of a contribution in complex crowdsourcing systems may become nontrivial [80, 81]

The main idea for evaluating users and their contributions is to detect spam and out of scope contributions and also differentiate low quality modifications from appropriate ones. Crowdsourcing systems utilize a combination of techniques in order to block, detect, and deter malicious users. First, crowdsourcing systems can block any malicious user by limiting who can make what kinds of contributions. For instance, Wikipedia blocks the IP address of a malicious user who attempts to add irrelevant and inaccurate materials to an article multiple times. As another example, anyone can submit an update for the Linux operating system, but only certain people such as the project originator have the capabilities to include or exclude a given update from the kernel.

Crowdsourcing systems can detect malicious users and contributions using two main approaches: content-based analysis and user-driven evaluations. Content-based analysis is an automatic method that typically involves some test. For instance, a system can ask users questions for which it already knows the answers, then use the answers of the users to compute their reliability scores [81, 82]. Many other schemes to compute users' reliability, trust, fame, or reputation have been proposed [83, 84]. User-driven approaches are

manual techniques including monitoring the system by the users, distributing the monitoring workload among a set of trusted users, and enlisting ordinary users (e.g., flagging bad contributions).

Finally, crowdsourcing systems can deter malicious users with threats of "punishment". A common punishment is banning. A newer and more controversial form of punishment is "public shaming", where a user U judged malicious is publicly branded as a malicious or "crazy" user for the rest of the community (possibly without U's knowledge). For example, a chat room may allow users to rate other users. If the (hidden) score of a user U goes below a threshold, other users will only see a mechanically garbled version of U's comments, whereas U continues to see his or her comments exactly as written.

2.7.1 User-driven quality evaluation in Wikipedia

Wikipedia uses user-driven approaches in order to differentiate between low and high quality articles. Wikipedia introduced the voting-based quality evaluations to tag articles as "Non-Featured Articles", "Good Articles" and "Featured Articles" [85]. Any user can nominate an article by listing it as a candidate for one of these categories. After the nomination of an article, it is flagged with a special tag. There are particular criteria based on the type of category in order to make the decision. Featured articles have the highest quality standard such as accuracy, completeness and well written. Good articles are also high quality articles, however, slight inconsistencies in the quality are tolerated (e.g., a lack of illustrations or small weaknesses in the writing style). Non-featured articles are the ones containing an unsuitable representation or a lack of relevance for Wikipedia. However, even this type of articles maintains a minimum standard of quality. The articles that are generally uncontroversial for deletion, such as those victimized by vandalism or other

nonsense, are deleted quickly by using the speedy deletion procedure.

After the nomination of an article, the community decides whether or not the article belongs to a certain category via a voting process. The voting period and the voting rule depend on the kind of evaluation. For example, to become a featured article, a voting period of 20 days and a slight modification of the two-third voting rule are necessary. After a successful election, the article will be added to one of the category by adding a special tag on top of the page [85].

2.7.2 Content-based analysis in Wikipedia

Wikipedia uses an anti-vandal detection mechanism called ClueBot [86] which utilizes machine learning techniques to detect user behavior and their vandalism. ClueBot learns to detect vandalism automatically by examining a large list of edits pre-classified as either constructive or vandalism instead of using a predefined list of rules that a human generates. According to the Wikipedia page [86], ClueBot catches approximately 55% of all vandalism correctly. In addition, Wikipedia uses a software called XLinkBot [87] to deal with domains frequently misused by new and anonymous users. The XLinkBot allows established users to add links, while links added by others will be reverted back.

Accordingly, other approaches were proposed to quantify the integrity of artifacts in crowdsourcing systems. These approaches tried to measure the quality of artifacts based on the length, the total number of revisions and the reputation of the editors [88, 89]. Blumenstock [90] demonstrated that the length of an artifact is the most accurate approach to distinguish high quality articles from low quality ones.

3

Confidentiality Management in Online Social Networks

3.1 Overview

There are two main challenges found in defining new access control techniques and controlling confidentiality of information on OSNs. The primary challenge relates to enforcing usage conditions. The typical corporate information networks [31] such as a course management network in a university use predefined roles (e.g. professor, teaching assistant, and student) and policies to regulate information flow. For example, a student does not have access to assignments or exams of other students, while students appointed as teaching assistants have access to all exams and assignments of specified courses. The employment conditions of the teaching assistants require them to keep certain information confidential. Information flow control in such a network breaks down if the users fail to abide by the usage conditions. The sharing problem in OSNs, however, is not governed by precise usage policies. The second challenge is that information sharing in OSNs is not automatically coupled with the level or the direction of interactions. In analog social networks [91], physical contacts remain as the dominant mechanism for sharing information between users.

Therefore, people can implicitly control information sharing by avoiding contact with undesirable friends. The explicit controls are necessary in OSNs to avoid information sharing with undesirable friends. Therefore, there is a need for novel access control techniques that work with minimal user intervention.

In this Chapter, we address the two challenges identified above by presenting a novel community-centric confidentiality control scheme for OSNs. We develop a new strategy where the eventual information distribution is shaped by the initial release of objects into the network. Because the initial release is completely controlled by the owner, she could shape the information distribution by making appropriate release decisions to minimize possible information leakage. To the best of our knowledge, this is the first work proposing an algorithm to closely approximate the exact risk of information leakage associated with user access control decisions. Our scheme uses a Monte Carlo method to compute the set of potential users who could receive the data objects belonging to a data owner. Our algorithm can provide input to a fully- or semi- automatic sharing decision maker that will determine the consequences of accepting or rejecting sharing requests. In addition, we provide algorithms for preventing information from reaching certain users by shaping the initial release set. Using datasets from Facebook and Flickr, we simulate sharing situations in social settings and estimate information leakage values considering that our algorithm controls information sharing.

The rest of the chapter is organized as follows: Section 3.2 discusses the secure information sharing problem and associated security challenges in social networking. In Section 3.3, a new community-centric confidentiality control mechanism for online systems is presented. One of the main applications of the new scheme is discussed in Section 3.4. Statistical, complexity and security analysis of our scheme is examined in Section 3.5. Section

3.6 presents an analysis of the experiments performed on information sharing patterns in Facebook and Flickr. Finally, Section 3.7 concludes the main points of the chapter.

3.2 Information Sharing Model

One of the important characteristics of OSNs is the private information space it provides for the users joining a network. After joining, users, at their discretion provide access to their friends using simple mechanisms provided by the OSN operators. Most OSN operators provide facilities to restrict access to subset of friends, friends, friend-of-friends, or public. These controls only deal with information release and expect the user to detect any misuse and modify the release conditions (for example, block an offending user from accessing data) [92, 93].

Information sharing in OSNs takes place without any formal usage restrictions or guidelines, which is an acceptable and preferable approach for OSNs users as shown in Figure 3.1. This survey was conducted to find the value of information sharing on OSNs among 200 McGill University students from various fields of study. Only 24 percent of participants like explicit sharing conditions when they receive data from their friends whereas the majority of the users prefer to attach specific constraints when they provide the information. This makes policy-based access control less suitable for OSNs sharing situations. Because information sharing is not carried out under strict usage conditions in the social networks, information leakage can occur widely. If an unauthorized user has access to the shared data, that object is said to be *leaked* or that *information leakage* occurred. Therefore, it is necessary to be able to compute the risk of information being leaked.

3.2.1 Scenarios

Here, we consider two different use-cases of computing the risk of unauthorized information leakage:

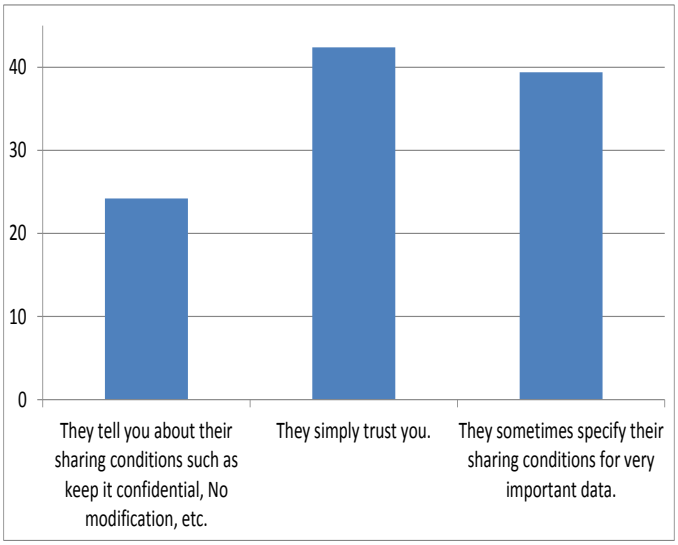
First, if a user needs to share some information with a subset of her friends (S_1), she should simply set the access control to S_1. However, there is a risk of information leakage associated with her decision for which she does not possess any knowledge. If there are intense and frequent interactions between S_1 and another subset of the user's friends named S_2, then the chances that the information shared with S_1 will leak to the members of S_2 is quite high. Therefore, there should be a mechanism to compute the risk of information leakage related to the user's sharing decisions and to provide the subset of the user's friends who will eventually have access to the shared information. Based on this information, the user can shape her access control decisions properly.

Second, if a user attempts to black list a specific user (her adversary), the only thing she can do in existing OSNs is to add the adversary to a black list. However, information she shares with her friends can reach her adversary if the adversary has a significant number of common friends with the user. Similarly, there is a need for a scheme to compute the risk of information leakage to the adversary and also to provide a list of friends who should be blocked in order to minimize the risk.

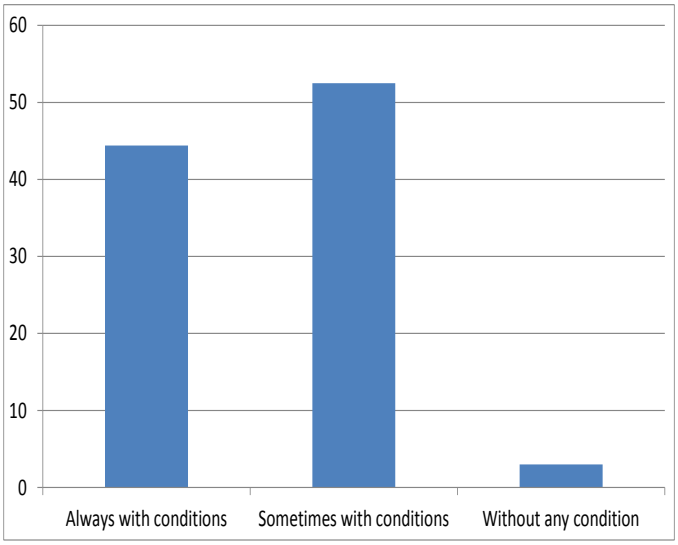
3.2.2 Assumptions

From the above-mentioned scenarios, we can observe that the information leakage problem is similar to the *cover channel [45] problem*. Our approach for information leakage is built on the following assumptions:

1. All information sharing takes place among users who are part of the OSN.



(a) When friends share data with you, do you prefer when



(b) Would you like to share information under specific conditions or under no conditions?

Figure 3.1: Survey of information sharing on OSNs

2. Users forming the OSN are focused on a specific topic (e.g., technology analysts). Therefore, when two users interact they are likely to share information known to them.

3. In OSNs, usage policies are implied and the receivers are trusted to uphold the norms. We assume that users are not equally likely to violate these norms.

3.3 α-myCommunity: A New Grouping Abstraction

In this section, we focus on developing an equivalent set of users for information sharing instead of relying solely on topological information such as friends or friends of friends. This set would be given to a community of friends likely to have access to the shared information only if a quorum of them is given the information. We can compute the equivalent sharing set as a connected component that surrounds a given user based on the communication patterns observed in the OSN. We refer to this set as the α-myCommunity hereafter. For a given communication intensity threshold α, our algorithm gives a subgraph which is likely to contain the information that is being released by the user. If the user intends to actually restrict the spread of information to a subset smaller than the set found in this subgraph, then the risk of information leaking outside the user imposes a restriction that is given by the α value used in the subgraph computation. Afterward, we develop a method to compute a set of friends who would leak the shared information to an adversary. For an acceptable threshold of information leakage, our algorithm provides a minimal subset of friends with whom the owner should stop sharing in order to prevent the shared information from flowing to the adversary.

3.3.1 Real life example

In November 2011, one Apple store employee named Crisp was fired for his critical Facebook posts. He posted negative messages about his employer on a Facebook friend's page assuming it was a private conversation. A coworker enrolled in the private group passed those messages to the Apple store manager [94] who fired Crisp. Although the messages were private, the communication was not protected because one of the friends decided to leak it. With all of the existing access control mechanisms in OSNs, Crisp was not able to prevent his posts from leaking to the manager. However, if Crisp knows about the risk of his information leaking out of the desired circle of friends, he can shape the information spread by not releasing the information to some of his friends. Because Crisp is only capable of controlling the release of information to his friends, he will use those decisions to maximize the information spread while the possibility of undesirable leakage is minimized. Better still, if Crisp could find the friends who should be blocked from getting access to the information, he could prevent it from flowing to his manager (his adversary) and he would never lose his job.

Accordingly, there is a need for a new scheme like α-myCommunity on top of existing access control mechanisms in OSNs in order to compute default sharing sets for users with minimal risk of information leakage. α-myCommunity can be automatically computed for Crisp based on the sharing patterns observed in the OSN and also notify him if there is a high risk of information leakage associated to his sharing decisions. However, our scheme has some limitations: first, our algorithm works only if the adversary is part of OSNs, in this case Facebook. Second, all information sharing should take place through OSNs. A user can share the information with others who are not part of OSNs by other ways than using social networking sites (e.g. physical contacts). Our scheme cannot prevent these

types of information leakage.

3.3.2 Estimating α-myCommunity

The identification of groups and communities inside a network has been one of the major topics in social network analysis [31]. Communities are subsets of users who are densely connected to each other. In other words, community members have high levels of mutual trust and shared interests. Because the concept of groups and communities has been used in different fields, there are various definitions for them in social networks. Here, we try to specify a community from the point of view of an individual user. We define myCommunity as the largest subgraph of users who are likely to receive and hold the information without leaking. That is, myCommunity is defined as a subset of a user's friends among whom there are relatively intense and frequent interactions.

We represent the social network with a directed weighted graph $SG = (U, E)$, where, U is the set of social network users and E is the set of edges representing relationships between the users of the network. In general, users are not equally likely to share the information. Some users are more willing to keep the shared data item confidential. If we denote P_i as the probability that user u_i is willing to share the information with some of her friends, P_i can be computed as follows:

$$P_i = \begin{cases} outflow/inflow & outflow < inflow \\ 1 & outflow > inflow \end{cases} \tag{3.1}$$

where outflow is the number of interactions user u_i has with her friends, and inflow is the number of interactions u_i's friends have with her. The weight on an edge between u_i and u_j ($w_{i,j}$) represents the likelihood of two users sharing information along the given

relationship. Therefore, the probability that user u_i shares her information with u_j is:

$$p_{i,j} = P_i \times w_{i,j} \tag{3.2}$$

Formally, *myCommunity (M_{u_i})* is the largest subgraph of the social graph ($M_{u_i} \subseteq SG$) including user u_i with the highest probability of information flow from u_i to all members. Similarly, we define *α-myCommunity ($MC^{\alpha}_{u_i}$)* as the largest subgraph of the social graph ($MC^{\alpha}_{u_i} \subseteq SG$) including user u_i with the probability of information flow from u_i to all members greater or equal to the threshold α ($MC^{\alpha}_{u_i} = \{\forall u_j \in U | PIF_{i,j}) \geq \alpha\}$). The *probability of information flow* (PIF) is the probability that user u_j will receive the shared data object from the owner u_i. It is important to note that myCommunity is defined for each user independently. Within myCommunity, specialized sub-myCommunities can be defined for specific contexts. It is also possible to develop aggregations of myCommunities in social neighborhoods to form ourCommunities.

Because finding the probability of information flow is an NP-hard problem [95], we propose a Monte Carlo based algorithm in order to determine the α-myCommunity for a specific user u_i [96]. We only consider a graph G_{u_i} as a subgraph of the social graph for u_i ($G_{u_i} \subseteq SG$), where $G_{u_i} = (U_{u_i}, E_{u_i})$ is a graph including u_i and all users with the hop distance equal or less than K from u_i. If we denote u_i and u_j as two users, the hop distance is defined as the smallest number of hops that separate u_i and u_j on the social graph. Hence, $G_{u_i} = \{u_j \in U, e_k \in E | |u_i, u_j| \leq K\}$, where $|u_i, u_j|$ is the hop distance between u_i and u_j.

In this algorithm, an information flow scenario g_s is randomly generated according to the sharing probability on each edge. With N iterations, $\widehat{n_j} = n_j/N$ is an estimation for the probability of information flow between u_i and u_j, where n_j is the summation of all

the random variables generated in the N iterations. That is, n_j represents the number of times u_j receives the shared information from u_i. Let indicator x_k be a random variable indicating whether the information sharing occurred on the edge between u_i and u_j of the graph G_{u_i}. That is

$$x_k = \begin{cases} 1 & \text{with probability } p_{i,j} \\ 0 & \text{with probability } q_{i,j}.\ (q_{i,j} = 1 - p_{i,j}) \end{cases} \tag{3.3}$$

Together, the variables x_1, x_2, ..., x_l generate an information flow scenario g_s of G_{u_i}, $g_s = (U_{u_i}, E_s)$, and $E_s \subseteq E_{u_i}$. The Monte Carlo simulation (MCS) that estimates the α-myCommunity for a specific user u_i is given as follows:

Algorithm 1 Finding α-myCommunity for u_i

Input: the social graph $SG(U, E)$ and a specific user u_i
Output: estimating α-myCommunity for u_i.

1: **Initialize** the variable n_j to zero: $n_j \leftarrow 0$
2: **Extract** the graph G_{u_i}
3: **Simulate** binary random variables x_k for each edge e_k ($e_k \in E_{u_i}$).
4: **Erase** each edge e_k if $x_k = 0$
5: **Set** $Y_{k,j} \leftarrow 1$ for all j, if there is a path between u_i and u_j. Otherwise, set $Y_{k,j} \leftarrow 0$
6: **Put** $n_j \leftarrow n_j + Y_{k,j}$
7: **Repeat** step 3 - 6 N times
8: **Estimate** $\widehat{n_j}$ for all j as $\widehat{n_j} \leftarrow n_j/N$
9: **Set** the probability of information flow between u_i and u_j as $\widehat{n_j}$, for all j: $PIF_{i,j} = \widehat{n_j}$
10: **return** all users with $PIF_{i,j} \geq \alpha$ as members of $MC^{\alpha}_{u_i}$

3.4 Application: Blocking an Adversary

One of the possible applications of Algorithm-1 on OSNs such as Facebook is to find out the minimum set of friends who should be blocked from getting access to the shared

information if the owner u_i attempts to prevent his or her data object from flowing to a specific user u_k. Here, we propose an algorithm in order to prevent the second scenario mentioned in previous section.

Algorithm 2 Finding friends of u_i who has the most effect on the probability of information flow between u_i and u_k

Input: the social graph $SG(U, E)$, u_i and u_k
Output: sorted set of u_i's friends based on their effect on $PIF_{i,k}$.

1: **for all** friends of u_i **do**
2: **Set** the probability of information flow between u_i and u_j to 0: $PIF_{i,j} \leftarrow 0$
3: **Find** the probability of information flow between u_i and u_k: $\alpha_j \leftarrow PIF_{i,k}$
4: **Find** the effect of setting $PIF_{i,j}$ to 0 on $PIF_{i,k}$: $ef_j \leftarrow \alpha - \alpha_j$
5: **Set** the probability of information flow between u_i and u_j to the original value
6: **end for**
7: **return** sorted set of u_i's friends based on ef_j for all j

Algorithm-2 is introduced to find out the most effective friend of u_i on the probability of information flow between user u_i and her adversary u_k. If u_i is willing to decrease the probability of information flow between itself and u_k to some new threshold ($PIF_{i,k} \leq \beta$), Algorithm-3 provides an estimation for determining the minimum set of u_i's friends with whom u_i should stop sharing.

3.5 Analysis: Statistical, Complexity and Security

Now we analyze the statistical properties of our algorithm.

Lemma 3.5.1. *The probability that there is a path between u_i and u_j in one iteration in Algorithm-1 is the probability of information flow between them $PIF_{i,j}$.*

Algorithm 3 Finding the set of u_i's friends with whom u_i should stop sharing to have $PIF_{i,k} \leq \beta$

Input: the social graph $SG(U, E)$, u_i, u_k, and β
Output: set of friends with whom u_i should stop sharing

Call Algorithm-2 to rank u_i's friends based on their effect on $PIF_{i,k}$
repeat
  Set u_j as the most effective friend of u_i on $PIF_{i,k}$
  Set the probability of information flow between u_i and u_j to 0: $PIF_{i,j} \leftarrow 0$
  Add u_j to the output set: $S \leftarrow S \cup u_j$
  Put $PIF_{i,k} \leftarrow PIF_{i,k} - ef_j$
  if for all friends of u_i, $PIF_{i,j} = 0$ **then**
    return S
  else
    continue
  end if
until $PIF_{i,k} > \beta$
return S

Proof.

$$\begin{aligned} Pr[\text{there is a path between } u_i \text{ and } u_j] = \\ Pr[u_i \text{ and } u_j \text{ are connected}] = \\ \prod_{k \in sharing} p_k \prod_{k \notin sharing} (1 - p_k) = PIF_{i,j}. \end{aligned} \tag{3.4}$$

Theorem 3.5.1. *The estimated probability of information flow obtained from Algorithm-1 $\widehat{n_j}$ is unbiased and consistent estimation of the exact probability of information flow $PIF_{i,j}$.*

Proof. In each iteration, we declare $Y_{k,j}$ for a user u_j as an indicator random variable with

$$Y_{k,j} = \begin{cases} 1 & \text{if there is a path between } u_i \text{ and } u_j \\ 0 & \text{otherwise.} \end{cases} \tag{3.5}$$

Because we use an independent and identical method for generating g_s in the N iterations, then $Y_{1,j}$, $Y_{2,j}$, ..., $Y_{N,j}$ are independent identically distributed random variables. By *Lemma 1*, we can achieve:

$$Pr[Y_{k,j} = 1] = PIF_{i,j} \tag{3.6}$$

In the algorithm, n_j is the result of a random experiment. It means that we only consider one particular replica of this random variable in a specific experiment. Therefore, we have:

$$n_j = Y_{1,j} + Y_{2,j} + ... + Y_{N,j} \tag{3.7}$$

where $Y_{k,j}$, $k = 1, ..., N$, are independent identically distributed binary random variables. Also, from *Lemma 1*, we know that the expected value of these binary random variables can be computed as:

$$E[Y_{k,j}] = Pr[Y_{k,j} = 1] = PIF_{i,j} \tag{3.8}$$

Accordingly, we have:

$$E[\widehat{n_j}] = E[n_j/N] = E[(Y_{1,j} + Y_{2,j} + ... + Y_{N,j})/N] = \sum_{k=1}^{N} E[Y_{k,j}(i,j)]/N = PIF_{i,j} \tag{3.9}$$

hence, $\widehat{n_j}$ is unbiased estimator of n_j. In addition, because $Y_{k,j} = Y_{k,j}^2$, so we have:

$$E[Y_{k,j}^2] = E[Y_{k,j}] = PIF_{i,j} \tag{3.10}$$

and

$$Var[Y_{k,j}] = E[Y_{k,j}^2] - E^2[Y_{k,j}] = PIF_{i,j}(1 - PIF_{i,j}) \tag{3.11}$$

Thus,

$$\begin{aligned} Var[\widehat{n_j}] = Var[n_j/N] = Var[\sum_{k=1}^{N} Y_{k,j}/N] = \\ \sum_{k=1}^{N} PIF_{i,j}(1 - PIF_{i,j})/N^2 = PIF_{i,j}(1 - PIF_{i,j})/N \end{aligned} \tag{3.12}$$

Therefore, our algorithm is not only unbiased, but also has variance tending to zero as the number of experiments increases ($\lim_{N\to\infty} Var[\widehat{n_j}] = 0$). In other words, with the increase of N, the value of $\widehat{n_j}$ becomes closer and closer to the exact value of the $PIF_{i,j}$:

$$\lim_{N\to\infty} P\{|\widehat{n_j} - PIF_{i,j}| < \varepsilon\} \tag{3.13}$$

That is, if the replication number N is large enough, any precision requirement ε can be accomplished.

Theorem 3.5.2. *Algorithm-3 is a greedy approximation for finding the minimal subset of u_i's friends who should be blocked in order to have $PIF_{i,k} \leq \beta$. The computational complexity of the algorithm is in order of $O(T)$ where T is the number of u_i's friends.*

Proof. To determine the minimum subset of u_i's friends with whom she should stop

sharing, we should go through all possible subset of u_i's friends, and for every subset, check whether the probability of information flow between u_i and u_k is less than the threshold β. In all possible solutions, the minimum subset would be the answer. The running time for this precise algorithm is $O(2^T)$. Therefore, we need a heuristic to find out the minimum subset in a reasonable time.

Our proposed algorithm, at each stage, considers u_i's friends with the largest effect on $PIF_{i,k}$. Because Algorithm-3, first, ranks the u_i's friends based on their effects on the $PIF_{i,k}$ by calling Algorithm-2, in worst case, it would go through all u_i's friends. Therefore, the Algorithm-3 is a greedy approximation with computational complexity of $O(T)$.

Claim 3.5.1. *The α-myCommunity for user u_i with highest value of α and largest size has the lowest risk of information leakage.*

Proof. Increasing the value of α results in a smaller α-myCommunity with only trustable friends. However, increasing the α value may not always lead to minimizing the risk of information leakage. If Alice wants to share her data object only with her close friends, she should set a high value for α such as 0.9 to only consider high trustable friends. The 0.9-myCommunity includes a few number of her friends (e.g. only five friends). With small change on the α value (e.g. from 0.9 to 0.88), the size of α-myCommunity for Alice increases largely. Therefore, it is better for her to relax the α constraint and share her data with a larger number of her friends.

On the other hand, decreasing the value of α not only increases the size of α-myCommunity, but also results in higher risk of information leakage. In OSNs, users want to control the information flow of their shared data in order to suit their needs and at the same time, privacy preferences. This means that control of the information sharing should be in a way to

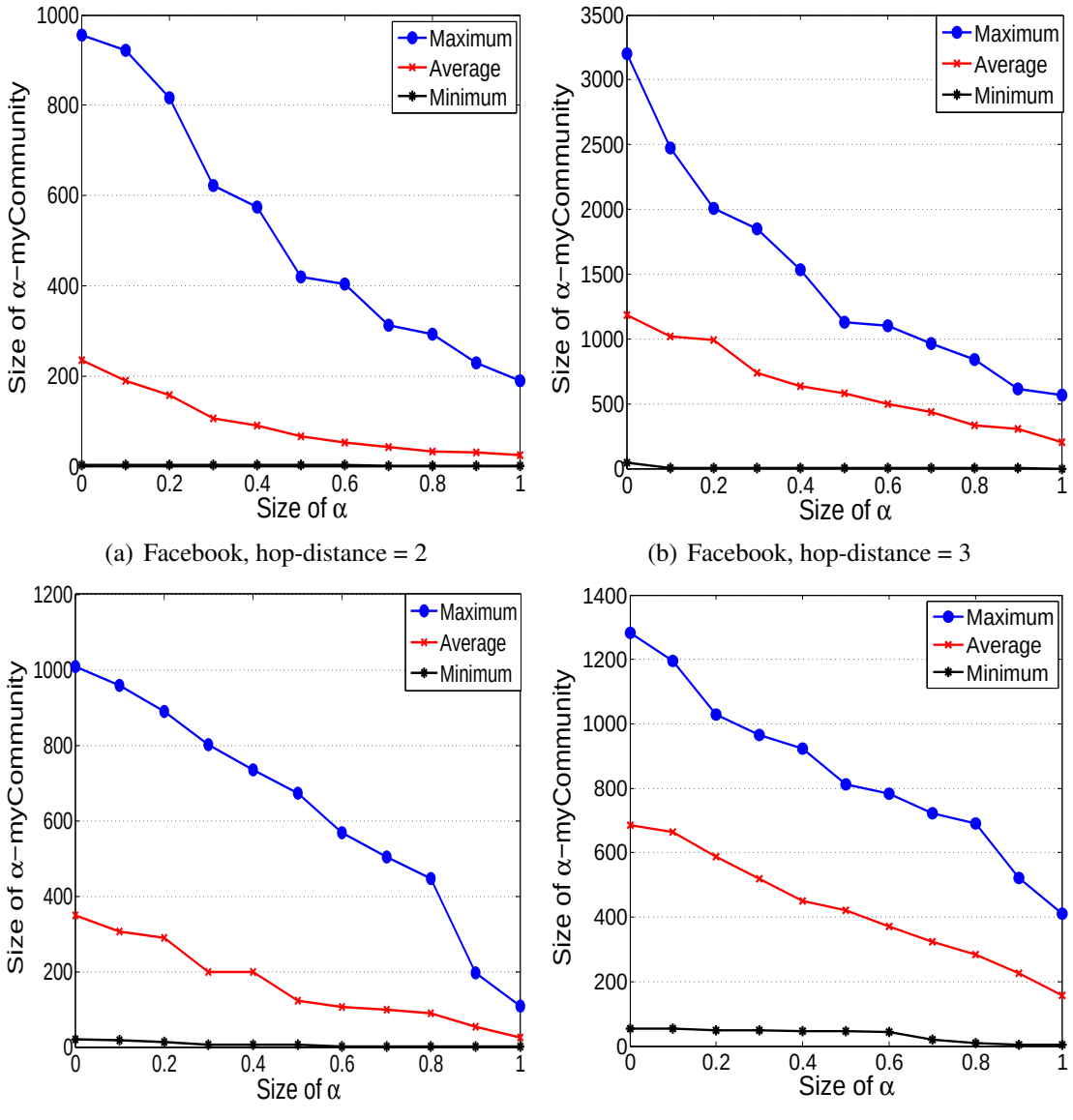
make balance between the need for privacy and the need for publicity. Therefore, it should be a tradeoff between the α value and the size of α-myCommunity. The best α value for user u_i will be achieved as follows:

$$\max\{\alpha \times size(MC_{u_i}^{\alpha})\} \tag{3.14}$$

3.6 Experimental Results

In this section, we use extensive simulations to evaluate α-myCommunity estimation schemes and analyze interactions of users inside α-myCommunities. To setup the simulations, we used topologies extracted from two different datasets, *facebook.com* [97], and *flickr.com* [98]. The Facebook dataset is a collection from New Orleans regional network with around $60,290$ users, $1,545,686$ friendships, and $876,993$ interactions between the users. These interactions can be any wall post such as posting videos, photos, and comments. The other dataset is based on traces collected from Flickr photo sharing site. While Flickr is an online photo album based social network, users connect with each other because of the quality of the photo albums. Therefore, Flickr permits users to have two different types of links: links to friends named contacts and links to favorite photos called favorites [98]. From this dataset, we extracted a subset of data with $100,000$ users, $3,638,215$ friendships, and $10,000,000$ interactions between the users. Interactions can be sending messages to other users, commenting on photos, tagging photos, and choosing favorite photos.

Using the friendship traces from both datasets, we constructed a synthetic social network as an undirected graph. It means that if user u is friend with user v, user v is also friend with user u. Similarly, we built an interaction network as a weighted directed graph using Facebook and Flickr traces. In this network, an edge from node u to node v exists if



(a) Facebook, hop-distance = 2

(b) Facebook, hop-distance = 3

(c) Flickr, hop-distance = 2

(d) Flickr, hop-distance = 3

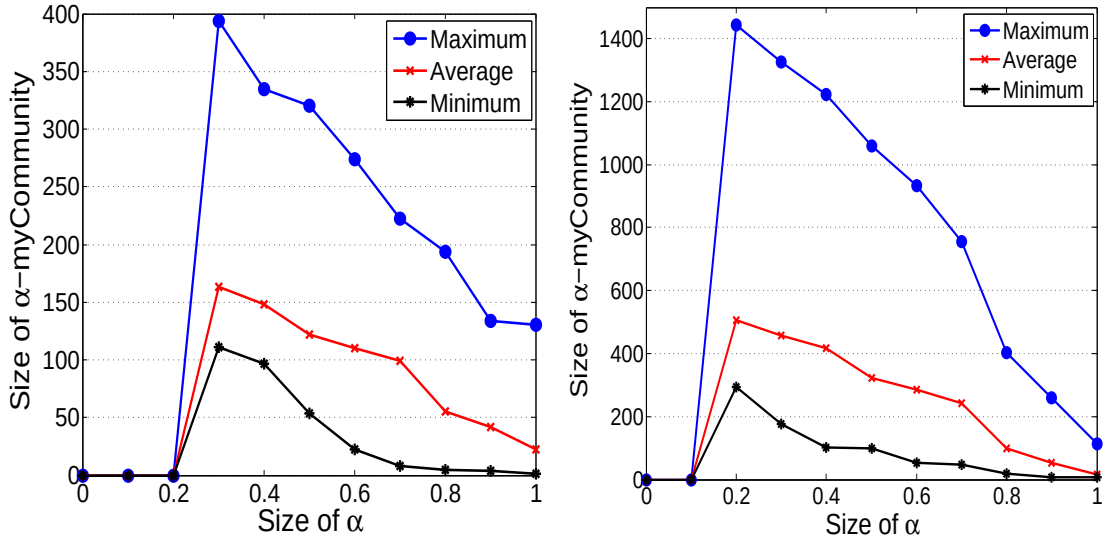
Figure 3.2: Distribution of α-myCommunity's size with equal α value

u has an interaction (wall post) with v. We assign a weight on each connection based on the number of interactions occurred between two users. Therefore, any edge would have a weight at least equal to 1. The larger the weight, the larger the probability would be for two users to share information. We have to mention that we assume all interactions are related to one topic. Analyzing interactions based on different topics is one of the future direction of this thesis.

Accordingly, we perform three different groups of studies. We first focus on the distribution of α-myCommunities with different values for α. Also, we try to find out the best value for α in both networks. Next, we evaluate the information leakage in α-myCommunities and analyze the behavior of users inside and with outside of their communities. Finally, we study the evolution of α-myCommunities over time. We find out how α-myCommunities would change in size and members throughout time. We have to mention that we run our algorithm with different number of iterations ($N = 100, 1000, 10000, 100000$). We notice that there is not a significant difference between the results achieved with N equal or greater than 1000. Therefore, N with value of 1000 leads to a promising estimation for our simulations.

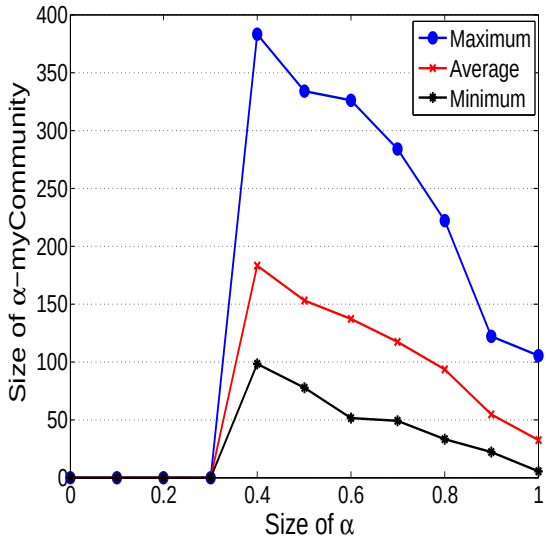
3.6.1 Analysis of α-myCommunity and blocking

In this section, we analyze our algorithms for computing α-myCommunities and blocking lists. We determine the α-myCommunity by considering different hop distance (2-hops or 3-hops) for each individual user. We only consider hop distance equals to 2 or 3, because the hop distance greater than 3 would nearly cover the whole network. We first utilize the same α value for all users to find out the size of α-myCommunities. Figure 3.2 presents the α-myCommunity's size distribution for different α values ($\alpha = 1, 0.9, ..., 0$). This

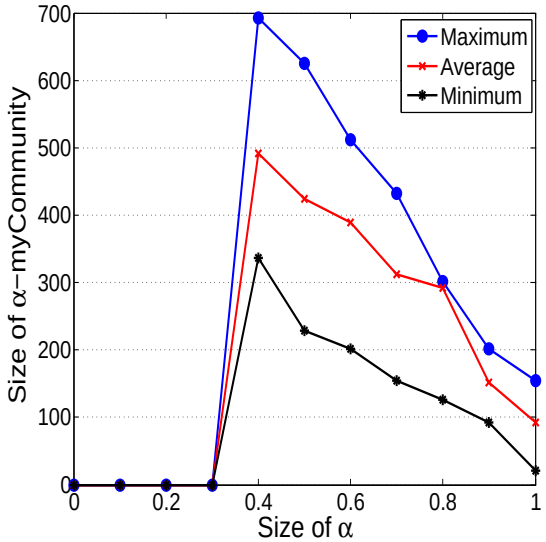


(a) Facebook, hop-distance = 2

(b) Facebook, hop-distance = 3



(c) Flickr, hop-distance = 2



(d) Flickr, hop-distance = 3

Figure 3.3: Distribution of α-myCommunity's size with best value for α

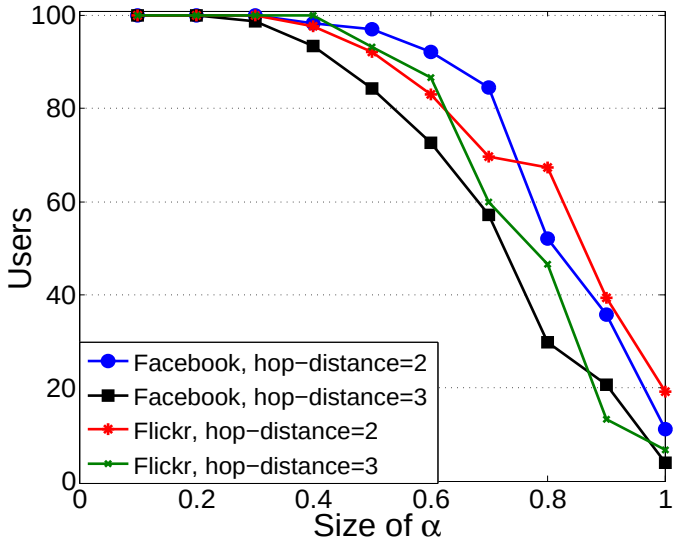


Figure 3.4: Percentage of user with the same α value

figure indicates the maximum, average and minimum size of α-myCommunity with hop distance equals to 2 and 3. The larger the value we choose for α, the smaller the size of α-myCommunity would be. The reason for this is that the smaller number of users has enough interactions to be considered in α-myCommunities with high α values. In addition, the size of α-myCommunities with hop distance equals to 3 is 3 times, for Facebook users, and 2 times, for Flickr users, larger compare to the size of α-myCommunities with hop distance equals to 2.

Second, we calculate the best value for α based on the Equation 3.14 for each individual user. As shown in Figure 3.3, the minimum value for α is 0.3, for Facebook users, and 0.4 for Flickr users. In addition, we find out the percentage of users having the same α value as shown in Figure 3.4. We notice that the best α value for Facebook user is equal to 0.7 as oppose to 0.8 for Flickr users. This is because of the fact that more than 84 percent of Facebook users have the α value equal or greater than 0.7. Similarly, for more than 67 percents of Flickr users, the α value is equal or greater than 0.8.

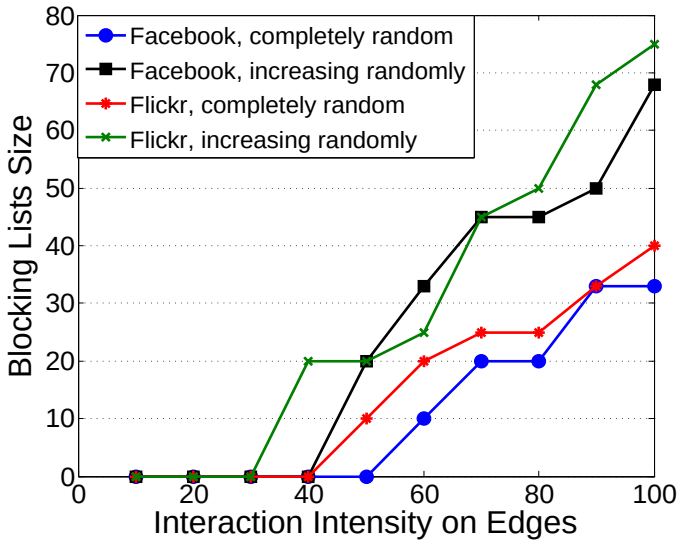


Figure 3.5: The average ratio of variations in the size of blocking lists by changing the interaction intensity on edges

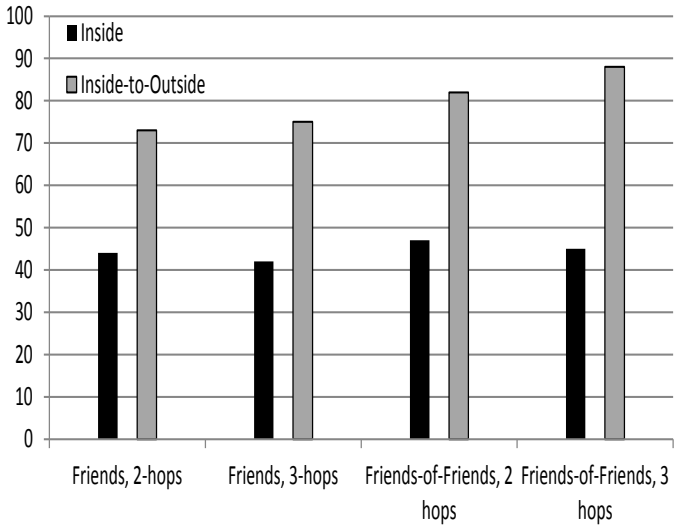
To analyze the blocking list scheme, we attempt to figure out how our algorithm would resist to the changes of the network. We calculate blocking lists for random adversaries for each user, and then change arbitrarily $10, 20, ..., 90, 100$ percentage of interaction intensity on edges in both networks. Figure 3.5 presents the average changes on the size of blocking lists for Facebook and Flickr networks. First, we increase or decrease at random the interaction intensity of edges in both networks. We notice that changes up to 60 percent of Facebook network, and up to 50 percent of Flickr network have no effect in the size of the blocking lists. Second, we only increase arbitrarily the interaction intensity of edges for both networks. In this case, the ratio of changes in the size of blocking lists is two times larger comparing to the first approach.

3.6.2 Information leakage

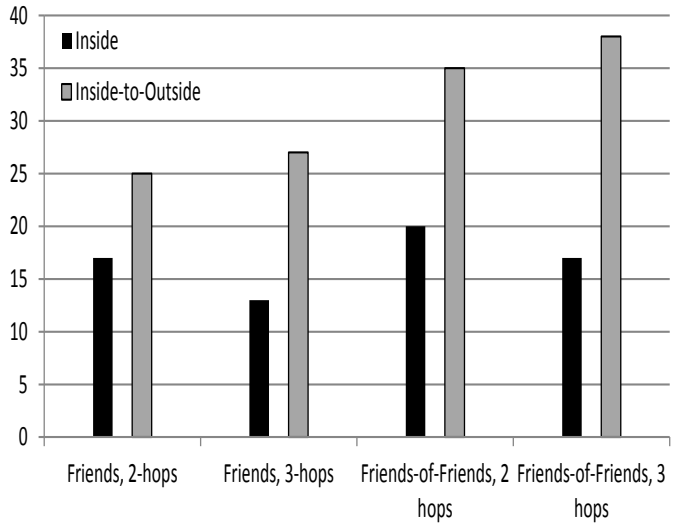
We now focus on determining how effective α-myCommunity would be in preventing information leakage with different values for α. As we mentioned earlier, information leakage occurs when an authorized user has access to the shared information. To evaluate information leakage in α-myCommunities, we analyze the interactions of members with users outside of communities. Since α-myCommunities include users who are likely to hold the shared information without leaking and have relatively intense and frequent interactions among each other, we consider the interactions between members and non-members as the information leakage.

To present how effective the information can be protected by the existing privacy settings in Facebook and Flickr, we try to find the information leakage if we set the privacy settings to friends or friends-of-friends. Figure 3.6 presents the normalized number of interactions which occurred inside the privacy setting (friends and friends-of-friends) and from inside-to-outside of these settings (information leakage). The number of interactions with users outside of the privacy settings is around 2 times more than the number of interactions that occurred within the circle of friends or friends-of-friends for both Facebook and Flicker users.

Accordingly, we try to set the privacy settings for each user to α-myCommunity in order to figure out the effectiveness of our approach. Figure 3.7 indicates the normalized number of interactions between members and from inside-to-outside of α-myCommunities with different values for α. The larger the value we choose for α, the smaller number of interactions would be between members and non-members. For instance, the number of interactions between members of α-myCommunities with α value of 1 is 30 times more than the number of interactions with non-members as shown in Figure 3.7(a). The information



(a) Facebook



(b) Flickr

Figure 3.6: Normalized number of interactions between members and from inside-to-outside with security setting to ”friends” and ”friends of friends”

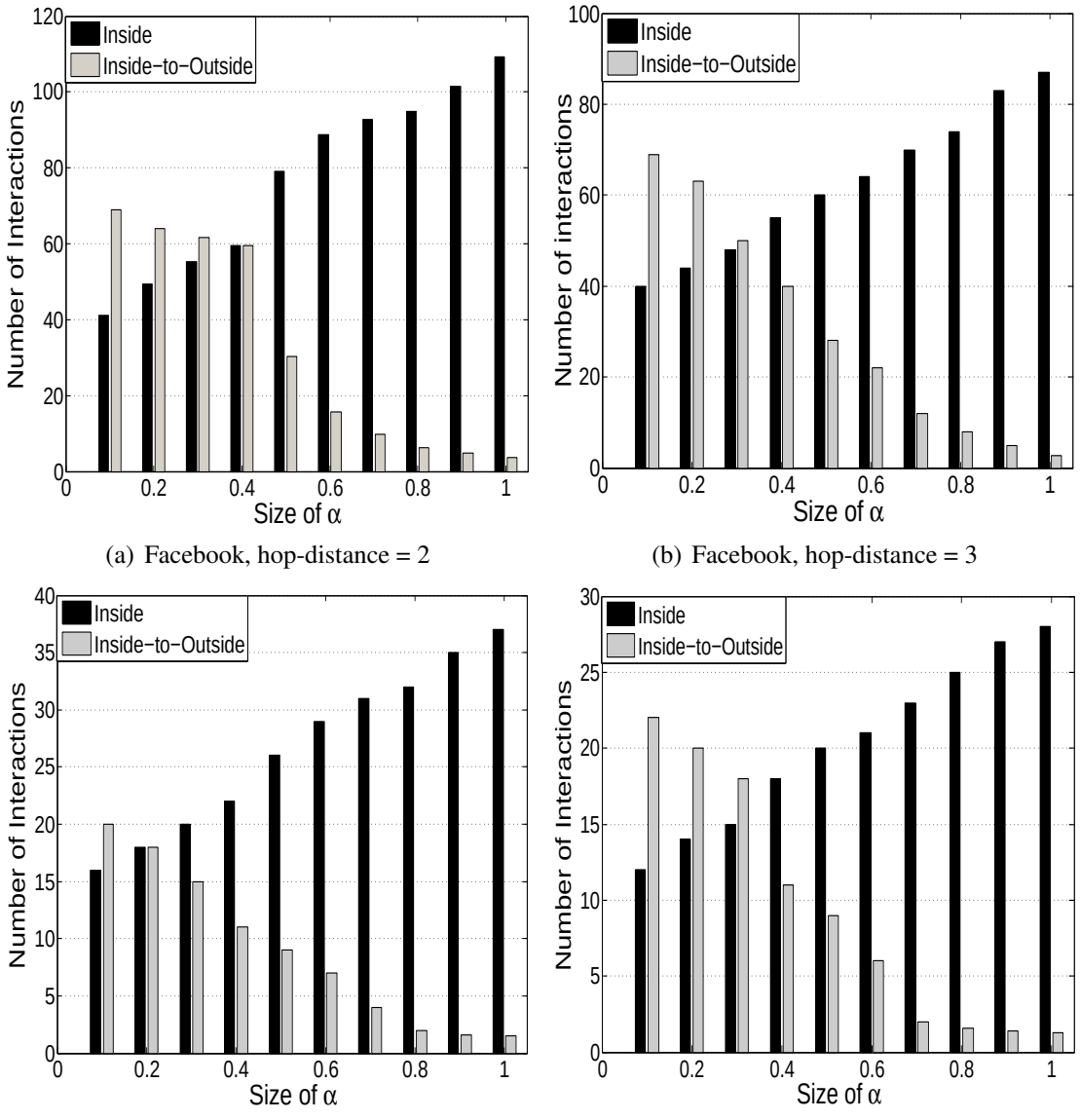
leakage (interactions with outsiders) is less than one percent of whole interactions when α is equal or greater than 0.7 for Facebook users. Similarly, the information leakage is less than one percent for α-myCommunities with α equal or greater than 0.8 for Flickr users. It means that α-myCommunity members have tendency to interact more with each other compare to their other friends and keep the shared information within the community.

3.6.3 Evolution of myCommunity and blocking list

In this part, we study how α-myCommunities and blocking lists evolve over time. To simulate α-myCommunities evolution, a total of 29 timeslots from Facebook traces, collected from September 2006 to January 2009, are considered. Each timeslot contains information about nodes and links in every month during this period. Similarly, we divide the Flickr traces into 20 independent timeslots, respectively containing $50,000$ interactions. We compute the α-myCommunity and blocking list for each user based on interactions that occurred during each time slot individually. We only consider current interactions between users and not their history. Afterwards, we try to find out what fraction of α-myCommunities and blocking lists persist from one timeslot to the next one. We use the notion of *resemblance* to measure the similarity between α-myCommunities and blocking lists in two consecutive timeslots. We define resemblance as the portion of α-myCommunity and blocking list members who remain in the community over two timeslots. Let denote R_t the resemblance of α-myCommunity or blocking list at time t. R_t can be defined as:

$$R_t = \left| \frac{MC^{\alpha}_{u_i}(t) \cap MC^{\alpha}_{u_i}(t+1)}{MC^{\alpha}_{u_i}(t)} \right| \tag{3.15}$$

Where $MC^{\alpha}_{u_i}(t)$ is the α-myCommunity for user u_i at time t. The value of R_t varies



(a) Facebook, hop-distance = 2

(b) Facebook, hop-distance = 3

(c) Flickr, hop-distance = 2

(d) Flickr, hop-distance = 3

Figure 3.7: Normalized number of interactions between members and from inside-to-outside of α-myCommunities

between 0 and 1 where $R_t = 1$ shows the entire α-myCommunity members continued to be part of the community at the next time step, and $R_t = 0$ represents that none of the α-myCommunity members who were part of the community at time t belongs to the community in time $t + 1$. Conversely, we define the ratio of variation in α-myCommunity as the complement of resemblance as follows:

$$V(MC^{\alpha}_{u_i}(t)) = (1 - R_t) * 100 \tag{3.16}$$

Figure 3.8 indicates the average ratio of variations in size of α-myCommunities when all users have the same values for α in both datasets. The larger the value we choose for α, the higher the variation we would have in the size of α-myCommunities. This is because of the fact that α-myCommunities with larger values for α have the smaller number of members who have a relatively large number of interactions through all timeslots. If a user has not enough interactions through one timeslot, she would not be considered as a member of the α-myCommunity with a large α value in the next timeslot. In addition, we notice that α-myCommunities with α equals to 0.7 for Facebook and 0.8 for Flickr have a relatively smaller ratio of variations compare to other values for α. The reason for this is that the best value for α is 0.7 for Facebook users and 0.8 for Flickr users.

Accordingly, Table 3.1 shows membership changes of a specific Facebook user's α-myCommunity with the α value of 0.7 during the first 10 timeslots. We show the intersections of the 10 computed α-myCommunities with each other in this table. For instance, let denote $C_1, ..., C_{10}$ for α-myCommunities in different timeslots. Here, the size of C_2 is 15, and the size of its intersection with C_4 is 10. It means that 5 members of α-myCommunity have not enough interactions in the forth timeslot and they were eliminated from the α-myCommunity. On the other hand, the size of the intersection of C_2 and C_7 is 15. It means

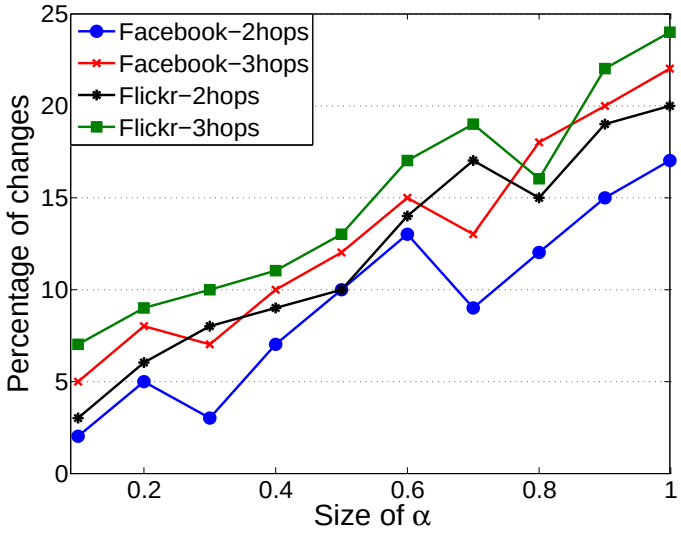


Figure 3.8: The average ratio of changes in size of α-myCommunities by considering timeslots

that those eliminated members joined back into the α-myCommunity. Therefore, it would better to compute α-myCommunities based on both current interactions between users and their history.

As a result, we define a time window including the interactions in current and three prior timeslots. To compute α-myCommunities in one timeslot, we consider the interactions that occurred throughout the time window in which the timeslot belongs. Figure 3.9 indicates the average ratio of changes in size of α-myCommunities with the same α while considering the interaction within time windows. The ratio of variations in the new method is smaller compare to the ratio of the previous scheme, since the history of interactions between users is considered as well. This shows that the resemblance value is sensitive to the size of the window. In addition, we present the membership changes of the same Facebook user's α-myCommunity based on the new method in Table 3.2. For example, the size of C_4 equals to 68 and its intersection with C_6 is 67. It means that from timeslot 4 to 6, only one

	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}
C_1	4	4	3	2	4	4	3	1	0	0
C_2	4	15	15	10	12	12	15	14	12	12
C_3	3	15	23	12	14	12	14	14	12	10
C_4	2	10	12	12	11	10	12	10	9	12
C_5	4	12	14	11	16	11	10	16	11	12
C_6	4	12	12	10	11	13	13	13	10	11
C_7	3	15	14	12	10	13	15	15	10	14
C_8	1	14	14	10	16	13	15	21	12	14
C_9	0	12	12	9	11	10	10	12	12	12
C_{10}	0	12	10	12	12	11	14	14	12	14

Table 3.1: Membership changes in α-myCommunity of a specific Facebook user through 10 different timeslots

member was eliminated and three new members were joined the α-myCommunity.

Similarly, we can define resemblance and ratio of variations for blocking lists. We compute the blocking list for each user by considering both timeslots and time windows. The average ratio of variations in size of blocking lists for timeslots method is 33 percent compare to 10 percent for time window. Finally, we try to find out whether a set of members who are part of α-myCommunity and blocking list exists in every timeslot or time-window. We define this set of users as the *core* of the community. When we consider timeslots, the core for α-myCommunity contains 56 percent of members comparing to 73 percent by considering time-windows. Further, the core for blocking lists in average includes 48 percent of members in the first timeslot and 68 percent of members in the first time-window. Therefore, considering time windows containing the history of interactions between users results in more stable α-myCommunities and blocking lists.

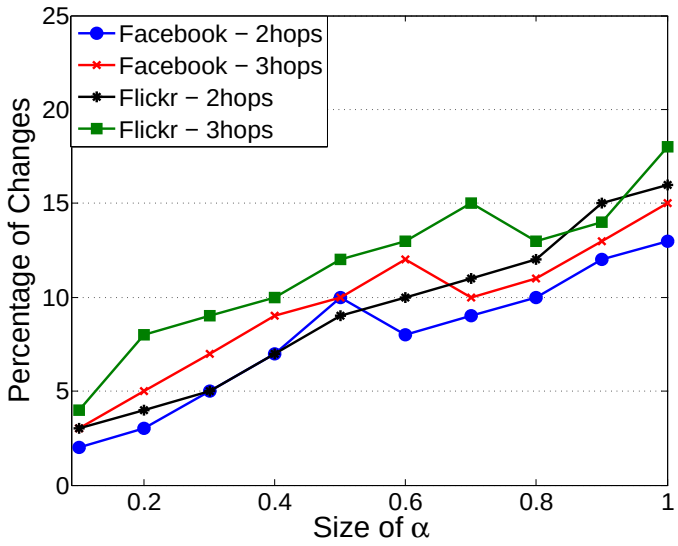


Figure 3.9: The average ratio of changes in size of α-myCommunities by considering time-windows

	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}
C_1	49	49	49	49	45	45	49	40	38	38
C_2	49	58	58	58	55	50	55	55	50	51
C_3	49	58	61	61	58	61	60	58	58	55
C_4	49	58	61	68	68	67	68	60	65	65
C_5	45	55	58	68	75	70	65	59	68	67
C_6	45	50	61	67	70	70	65	60	65	65
C_7	49	55	60	68	65	65	65	60	65	65
C_8	40	55	58	60	59	60	60	60	60	60
C_9	38	50	58	65	68	65	65	60	69	68
C_{10}	38	51	55	65	67	65	65	60	68	75

Table 3.2: Membership changes in α-myCommunity of a specific Facebook user through 10 different time-windows

3.7 Summary

In this Chapter, we considered the problem of controlling confidentiality of information on online social networks. When sharing content, two issues stand out: enforcing usage conditions and ensuring that the conditions are respected once the content is shared. We presented a Monte Carlo based algorithm to compute the sharing subgraph which the information will disperse on the network. We refer to such subgraphs as α-myCommunity, where α specifies the certainty that all members of the subgraph will know about the information.

From the experimental results, we noticed that the larger the value of α the smaller the size of α-myCommunity. This indicates that interactions in OSNs take place in tightly knit groups. These groups, however, may not be cleanly defined by notions such as friends, or friends-of-friends. Also, the best α value for Facebook users is about 0.7. This means by choosing 0.7-myCommunities Facebook users can form communities that are robust in terms of information sharing. The information discharged into such communities are more likely to be contained within them. Similarly, for the Flickr network, we observed an α value of 0.8. Instead of asking the user for an α value the algorithm select the value yielding a robust configuration. In addition, blocking the adversary directly may not prevent flow of information to him. This indicates that users can receive information not only directly from the owner, but also indirect ways from common friends. However, the indirect ways may not be cleanly defined by existing notions in OSNs. Finally, because interactions are time dependent, we evaluated different ways of computing α-myCommunities. By including historical information in the computation process, we were able to compute relatively stable α-myCommunities.

4

Integrity Management in Crowdsourcing Systems

4.1 Overview

Two extreme approaches have been introduced for maintaining integrity in crowdsourcing systems: Wikipedia style and Linux style [59]. With Wikipedia, integrity emerges out of the crowd activity and can be less effective on sections of the encyclopedia that do not gain wide exposure. Incorrect or intentional bias can be introduced into the Wikipedia articles and remain there until subsequent readers flag the problems. On the other hand, Linux-style approaches tightly control the integrity by funnelling all changes through a single authority. This style resolves the inaccuracy problem, yet increases the workload and creates a bottleneck due to having a single authority.

In this chapter, we present Social Integrity Management (SIM), a new approach for managing the integrity of content created in crowdsourcing repositories. Our approach assumes that the users are interconnected by an online social network such as Facebook, where each user specifies the trustworthiness of her friends. Our scheme enjoys the benefits of the two existing styles, i.e., any user can be a potential writer to create a new article (Wikipedia Style) while the integrity is enforced by having ownership (Linux style). SIM

brings ownership as a key feature to control integrity of documents. The bottleneck created by ownerships is relaxed in SIM by including co-ownerships and multiple versions.

The rest of this chapter is organized as follows. The challenges and requirements for preserving integrity in crowdsourcing systems are discussed in Section 4.2. Section 4.3 presents our analysis on Wikipedia's integrity based on dump datasets from its website. In Section 4.4, we detail the design and the rationale of the SIM approach. Section 4.5 presents numerical analysis of SIM using dataset from Facebook. Finally, Section 4.6 summarizes the major features of the SIM.

4.2 Integrity of Crowdsourcing Systems

In a crowdsourcing environment where large number of users are involved in updating large number of documents, integrity needs to be defined with due diligence. In most crowdsourcing repositories (e.g., Wikipedia) only one version of a document is retained. This means contention can arise due to conflicting opinions. To reduce the bias, crowdsourcing systems often include different sections for explaining different viewpoints. If an article is owned by a single user (e.g., Google-Knol), this problem does not arise. However, the article itself can be biased because the owner might be filtering out the opinions he disagrees with. This incurs the possibility of having multiple articles on a single topic, because users concerned about the bias of an existing article might start their own version to provide the alternative opinion.

In Table 5.4, we use different attributes to characterize twelve crowdsourcing information systems. The first attribute is the *type of identity* required by the site. Except for Wikipedia, all other sites in this table require some form of login to create an article or edit contents that are already present in the site. There are two types of identities: (i) real

identity and (ii) pseudo identity. Sites such as Scholarpedia and Citizendium require real identity; i.e., the contributors have to register with their physical identifying information such as their curriculum vitae. Once the physical identity and the accompanying information of a user are verified, the user is given access to the site. Another form of identity that is more commonly required is the pseudo identity. In this case, users accumulate reputation on the pseudo identity and the privileges associated with a user are directly dependent on her reputation.

All sites, except Wikipedia, include ownership in their scheme in order to control the integrity of the generated articles. Therefore, editing capabilities can only be granted by the owner of an article (e.g., Scholarpedia and Citizendium) or by accumulating enough reputation (e.g., Stackoverflow). These systems try to encourage users to contribute more by providing incentives such as different capabilities within the system based on the level of their reputations. However, Squidoo or About sites provide some sort of remunerations to motivate users for contributing in their systems.

Another important characteristic of these sites is the *number of articles per topic*. Depending on the nature of the site, we can have one or many articles for a specific subject. Wikipedia, Scholarpedia and Citizendium as online encyclopedias have only one article per topic. However, Google-knol, Squidoo, Hubpages, Helium, Examiner and DailyTech make it possible to have multiple articles when writers have different opinions on a specific subject. In addition, users can ask the same question multiple times in question answering sites such as Stackoverflow and get different answers.

Name	Type of identity	Ownership	Editing capability	Need for approval	Rating	Scope of topics	Number of articles per topic	Remuneration
Wikipedia	Anonymous	No	Anyone	No	No	All	1	No
Knol	Google account	Yes	No, with owner's permission	No	Yes	All	Many	No
Scholarpedia	Real id	Yes	Yes, approved of author	Yes, expert	No	Scientific	1	No
Citizendium	Real id	Yes	Yes, before approval	Yes	No	All	1	No
Squidoo	Pseudo id	Yes	No	No	No	All	Many	Yes
Hubpages	Pseudo id	Yes	No	No	No	All	Many	Yes
Helium	Real id	Yes	No, owner	No	Yes	All	Many	Yes
Examiner	Real id	Yes	No	No	No	All	Many	Yes
Instructables	Pseudo id	Yes	No, contractor writers	Yes	No	Do-It-Yourself instructions	1	Yes
About	Real id	Yes	No	Yes, guides should accept it	No	All	1	Yes
DailyTech	Pseudo id	Yes	No, owner	No	Yes	Technology news	Many	No
Stackoverflow	Pseudo id	Yes	Yes	No	Yes	Programming	Many	No

Table 4.1: Characteristics of some crowdsourcing websites

4.2.1 Challenges

In this section, we discuss some of the challenges of creating crowdsourced information repositories. We categorize these issues as follows:

1. **Ownership of articles:** Suppose Alice is creating a document X. In a typical publishing scenario, she retains the ownership of document X for its lifetime. If anyone else wants to edit or modify it, they need to get Alice's permission. To this end, they have to submit their modifications to Alice and she has to decide which modification should be accepted or rejected. Obviously, such a publishing model is not very suitable for community-centric or large-scale collaborative publishing. In this model, Alice's ownership becomes a bottleneck because she must validate each and every modification. This is one of the reason that Wikipedia takes the ownership away from its core model. It attempts to provide integrity without requiring ownership.

2. **Accuracy of articles:** The involvement of large number of writers, often non-experts in Wikipedia-style websites, results in unreliable and unprofessional documents. Also, having a single version for each article may result in contentions that will need further resolution. The main problem here is the lack of authority and fact-checking. Someone should report the problem; otherwise, inaccurate information that is not obviously false may persist in Wikipedia for a long time before it is challenged [23].

3. **Duplication of effort:** Some crowdsourcing websites offer an option of having multiple articles on the same topic, each being written by a different author (owner). This can resolve the contention problem mentioned previously. However, in this model, a reader would have difficulty in searching and discerning the relevant articles from the irrelevant ones.

A majority of crowdsourcing websites employ the owner-centric model to solve the article accuracy problem experienced by Wikipedia. If a user creates an article, she is the owner of that document and also responsible for its credibility and future modifications. Since a user's reputation is highly dependent on the quality of articles created by the user, the user would pay more attention to the accuracy of her articles; hence, the quality of articles can be significantly improved. In the owner-centric model, various access control schemes were proposed to preserve the integrity of contents. In the next section, we address major requirements for access control schemes in integrity management.

4.2.2 Requirements

As mentioned earlier most crowdsourcing systems employ the owner-centric model which uses access control schemes for preserving the integrity of contents in a crowdsourcing site. Here, we discuss the requirements of all access control schemes used in the owner-centric model [56]:

1. **Full control:** An access control scheme should provide the owner with full control on how she modifies access to other users.

2. **Flexibility:** A crowdsourcing website can have various data and editing requirements. Therefore, the access control scheme should be flexible and capable of operating at different data granularities. This flexibility requirement has various aspects including: providing different levels of editing capabilities for different users, changing editing conditions of an existing content (e.g., increasing its integrity level), and changing editing capabilities of a user (e.g., decreasing the trustworthiness of a user and consequently limiting her access to modify contents). While flexibility is essential in access control schemes, it can add significant overhead in terms of user effort

required to setup and maintain an ownership model.

3. **Collaborative environment:** The user effort required by the access control scheme is certainly a major factor in its eventual acceptance in the user community. One way of retaining the flexibility and reducing the user effort required by the access control scheme is to enable collaborative decision making for each user. This can be done for a certain user (Alice) using one or a combination of the following methods: learning from Alice's past activities, learning from past activities of the community within a social neighborhood of Alice, and learning from the past activities of a set of users who are similar to Alice within her social neighborhood.

4. **Prediction of accessibility:** Another important requirement for an access control model is the ability to predict the accessibility of created contents. For instance, Alice would like to know the list of users who are able to modify the new article she is creating with a particular integrity setting. Such prediction of accessibility can be used to interactively shape the integrity settings for important data.

Some of these requirements have already been implemented in Stack Exchange sites which we will discuss them in detail in Chapter 5.

4.3 Wikipedia's Integrity

In this section, we consider the integrity of Wikipedia as one of the most widely-used collaborative (crowdsourcing) systems. We first challenge its integrity by performing some experiments. Afterwards, we analyze the dump datasets from its website to find the reasons behind high integrity for few articles and low integrity for majority of them.

4.3.1 Challenging Wikipedia's integrity

To meet its integrity objectives, Wikipedia encourages contributors by facilitating easy article creation and update. The integrity of a user's contributions are checked and flagged by subsequent readers. For highly trafficked articles, this model of integrity enforcement works very well. However, when integrity emerges out of the crowd activity, it can be less effective on sections of the encyclopedia that do not gain wide exposure. Incorrect or intentional bias can be introduced into the Wikipedia articles and remain there until subsequent readers flag the problems.

In Wikipedia, there are two different modes for user contribution: anonymous and pseudonymous. In the anonymous mode, writers will not have full privileges and Wikipedia keeps their IP addresses in the history of the page. With pseudo identities, users accumulate reputation and the privileges associated with the users are directly dependent on the corresponding reputations.

To examine how Wikipedia preserves the integrity of its pages, we conducted some experiments by creating new pages containing false information and by modifying existing pages by adding non-related sentences and URL links.

Creating a new page

We tried to create a new page with invalid content. Every new page proposal by an anonymous or new user goes through a validation by one of the Wikipedia's editors. The editor will check whether the page already exists, has commercial or offensive content or includes invalid information. After validation, the proposed page will be added to Wikipedia. The page we created could not pass the validation phase as we were anonymous or new user and had invalid content.

Modifying an existing page

Next, we attempted to modify an existing page by adding random sentences. New and anonymous users' modifications to existing pages go through an automatic validation phase. Wikipedia uses an anti-vandal detection mechanism called ClueBot [86] which utilizes machine learning techniques to detect users behavior and their vandalism. Our modifications were flagged by this software as invalid and never applied to Wikipedia pages. To mislead the ClueBot validation, we tried to add sentences with incorrect information containing words related to the page's topic. In this case, the ClueBot failed to detect our modifications.

Adding random URL links

In this experiment, we tried to add random, non-related URL links to popular and unpopular pages. We realized that all added links from new or anonymous users go through validation phase similar to the previous experiment. Wikipedia uses a software called XLinkBot [87] to deal with domains frequently misused by new and anonymous users. The random URL links, introduced by us, were detected and removed by XLinkBot. To mislead the XLinkBot validation, we attempted to modify references in some pages by copying existing URL links from other Wikipedia pages. Here, we had two different scenarios where XLinkBot becomes confused. In the first scenario, the topic of the two pages are completely different and non-related. Since the copied link has no relation to the page's topic, the XLinkBot is still able to detect this modification and remove the link. In the second scenario, we considered two pages which are related to each other, i.e., one page has linked to the other one. In this case, XLinkBot failed to detect our modifications.

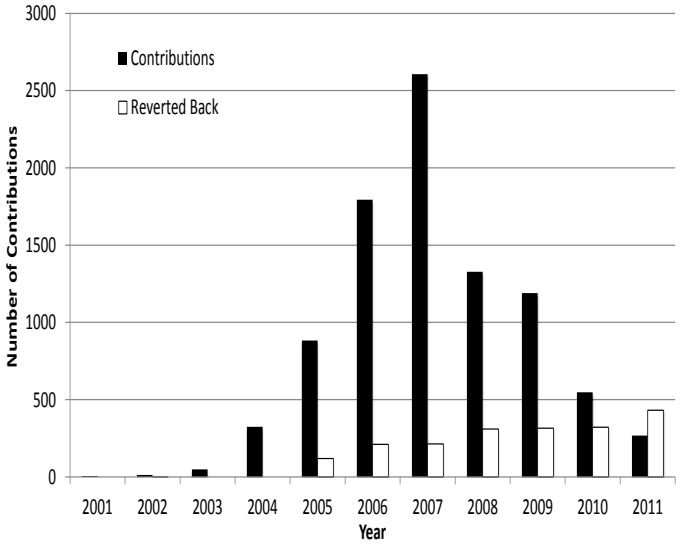
After running these three experiments, we can conclude that the integrity of the page

depends on the readers who are responsible to find unrelated references and remove them by rolling back the page. However, when Wikipedia readers have less interest in some articles, their integrity may not be heavily scrutinized as pages with heavy readership.

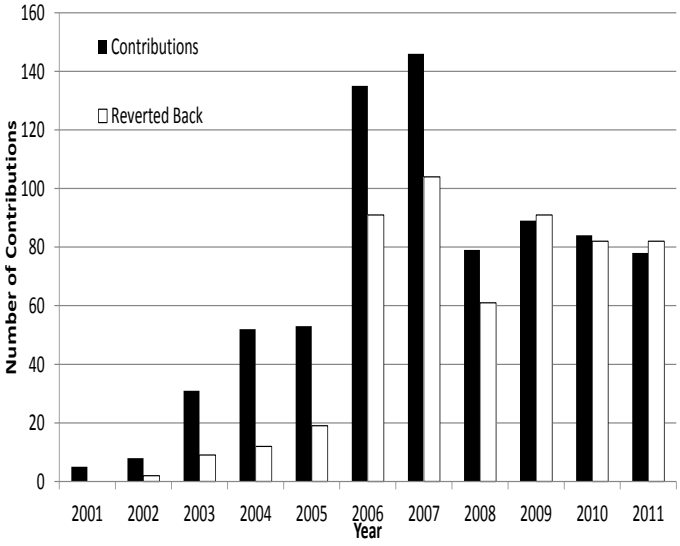
4.3.2 Analyzing Wikipedia's integrity

To solve the integrity problem, Wikipedia has developed various approaches for evaluating its articles. Wikipedia provides a user-driven approach where users can vote for articles to be marked as "*Featured Articles*," "*Good Articles*," or "*Non-Featured Articles*." Here, we aim at providing a better understanding of how an article becomes a featured article while others remain at low quality.

In this integrity analysis, we used the Wikipedia dump dataset for a period of ten years from 2001 to 2011. The dataset includes XML files containing the source texts of all pages with their complete edit history. The edit history contains the usernames or the IP addresses of the editors and the modification times. Because the size of the dataset is very large, Wikipedia divided the dataset into many files; each one contains information for around one thousand pages. Here, we only considered 100 good and featured articles as the sample for high quality articles and 100 non-featured articles as the test-case for low quality articles. This is a reasonable amount of data given that there are only $3,783$ featured articles in Wikipedia which is around one of every $1,100$ articles. Also, note that extracting the information about these types of articles is an extremely time consuming procedure. In addition, there are many low quality articles with very few contributions. Here, we only consider an article as part of our test-case if the article has more than 100 modifications in its edit history.



(a) Featured and good quality articles



(b) Low quality articles

Figure 4.1: Number of contributions for low and high quality articles

Contributions and reverted back modifications

The number of contributions and the number of reverted back modifications for both low and high quality articles are shown in Figure 4.1. For low quality articles, the number of reverted back modifications is similar or larger than the number of accepted contributions. It seems that Wikipedia community for low quality articles could not agree on the content; therefore majority of contributions were reverted back. On the contrary, the evolution of high quality articles shows a significantly different pattern. In general, the number of contributions for high quality articles is larger compared to the number of contributions for low quality articles. It appears that in a particular period of time (2006 and 2007) the high quality articles became the focus of the Wikipedia community; thus the number of contributions rose with increasing maturity. Afterward, the articles became good or featured and the number of contributions decreased.

Generally, the number of reverted back modifications for high quality articles is smaller compared to the number of accepted contributions. However, after the articles became good or featured, the number of reverted back modifications exceeded the number of accepted contributions. This trend suggests that with increasing maturity, Wikipedia community tends to accept less number of new contributions. Therefore, a lot of contributions are reverted back when the articles are of high quality.

Major and minor contributions

Wikipedia categorizes modifications as minor or major. A minor edit is one where the modification requires no review and could never be the subject of a dispute. An edit of this kind is marked in its page's revision history with a lower case, bold "m" character (m). However, a major edit is one that should be reviewed for its acceptability to all concerned

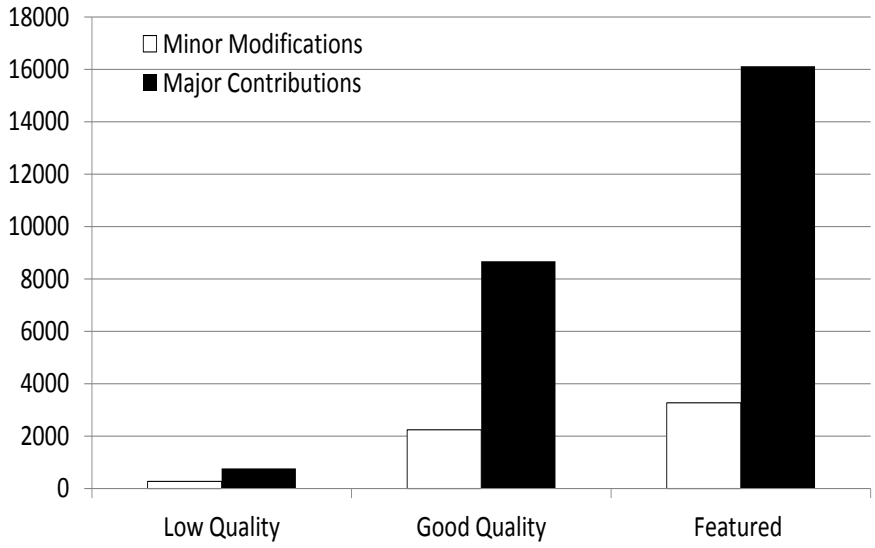


Figure 4.2: Average number of major and minor contributions

editors. We could observe that most of the contributions for high quality articles are major (i.e., changing a big portion of an article). For high quality articles, around 85 percent of contributions were tagged as major compared to only 52 percent of the modifications being major for low quality articles as shown in Figure 4.2.

Contributors

We found the number of contributors for different types of documents as shown in Figure 4.3. In this figure, we index the pages from 1 to 200 and sort them in terms of the number of contributors (horizontal label). The average number of contributors for low quality articles is 356 compared to $1,621$ for good articles and $3,332$ for featured ones. This shows that high quality articles gain wide exposure 10 times as much as low quality articles. In addition, we noticed that there is a highly active group of contributors involved from the creation of high quality articles until present. However, the majority of editors for low quality articles never contribute after their first contributions.

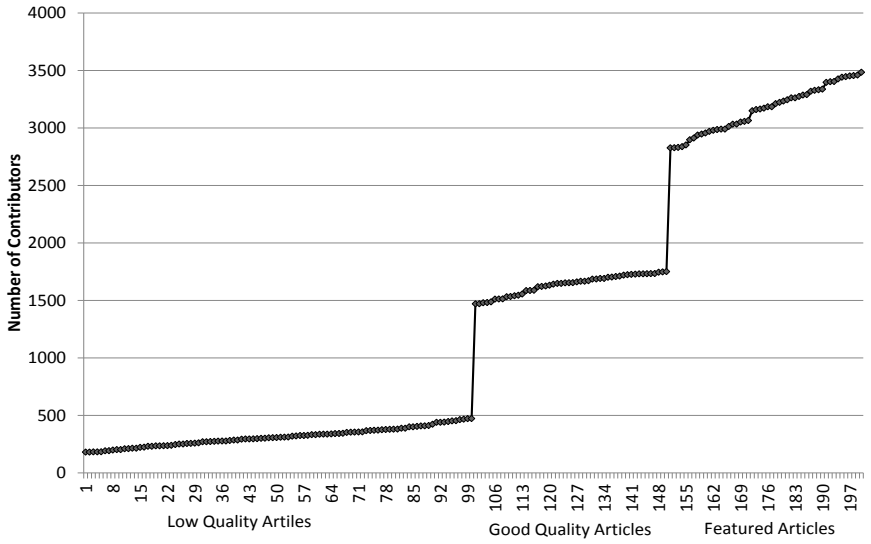


Figure 4.3: Number of contributors for low quality, good quality, and featured articles

We consider an editor as a top contributor for an article if the number of times her modifications got reverted back is much smaller than the total number of her contributions. In other words, the set of top contributors (E) for an article will be:

$$E = \{\forall u_i \in U | R_{u_i}/C_{u_i} < \epsilon, C_{u_i} > \Delta\}, \tag{4.1}$$

where U is the set of all editors, R_{u_i} is the number of revert backs done on contributor u_i's modifications, C_{u_i} is the total number of contributions for editor u_i, and Δ and ϵ are the thresholds. We need the threshold in order not to consider editors who only contributed few times and all their modifications got accepted. We found out that for high quality articles the average number of top contributors is 32 for the threshold $\Delta = 50$ and $\epsilon = 0.1$. In other words, a very small group of contributors are responsible for the majority of activities around a high quality article. Henceforth, we focus on this small group of top contributors and analyze their activities and impacts.

Characterizing top contributors

To analyze the activities of top contributors in more detail, we measured the number and quality of their contributions (minor or major) as well as the number of revert backs. We noticed that they are responsible for more than 62% of accepted contributions and around 85% of revert back modifications for high quality articles. Figure 4.4 shows the average number of revert backs done by top contributors. This figure shows an increase in the average number of revert backs through the years. In addition, we measured the quality of top contributors' modification by finding whether they were minor or major. We figured that more than 90 percent of their contributions were tagged as major compared to other contributors whose modifications were tagged minor more than 92 percent. It seems that for high quality articles, top contributors formed an informal group which is responsible for the page. Although Wikipedia is a democratic publishing platform which provides contributing capability for everyone, the top contributors tend to impose their opinions by contributing more and reverting back other editors' modifications.

As another characteristic of top contributors, we used the notion of resemblance to measure the similarity between two sets of top contributors in two consecutive years. We calculated the set of top contributors for each year separately in the interval 2001 to 2011. We define resemblance as the portion of top editors who remain in the set over two years. Let R_t denote the resemblance of two editing sets at year t. R_t can be defined as:

$$R_t = \left| \frac{E(t) \cap E(t+1)}{E(t)} \right|, \tag{4.2}$$

where $E(t)$ is the set of top contributors for a high quality article at year t. The value of R_t varies between 0 and 1 where $R_t = 1$ shows the entire set of top contributors stayed for the

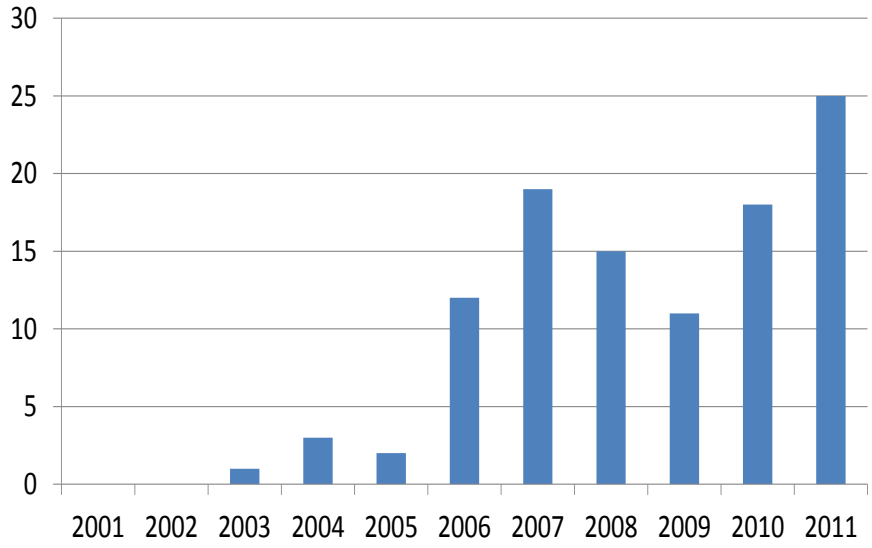


Figure 4.4: Average number of reverting back done by top editors of high quality articles

next year, and $R_t = 0$ represents that none of the top contributors at year t remained in the subsequent year $t + 1$. Figure 4.5 represents the average resemblance of top contributors for 100 high quality pages. It appears that the majority of top contributors were present from the creation of high quality articles until the present time. Also, after the documents reached their maturity and became featured or good articles in Wikipedia, the resemblance of top contributors in subsequent years is more than 90 percent. In other words, the top contributors have become the owners of high quality articles and their engagement has increased.

Accordingly, we attempt to find the similarity of top contributors in terms of their interests. For this purpose, we measured the overlap between the topics which have been mostly edited by top contributors. Table 4.2 presents the similarity of top contributors for high and low quality articles. Top contributors of high quality articles are more like-minded than the top contributors of low quality articles. We noticed that the similarity between top 10, 20, and 30 contributors of high quality articles is more than 80 percent. This value suddenly

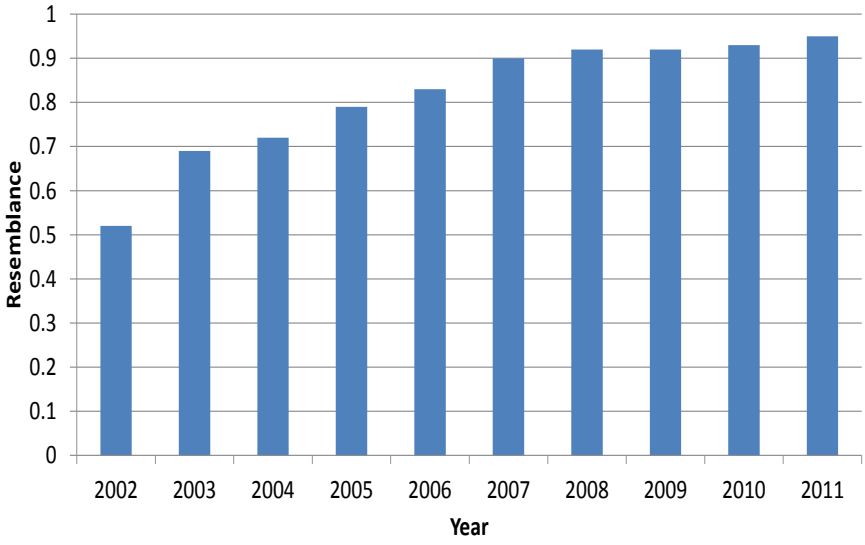


Figure 4.5: Resemblance between top contributors

Table 4.2: Similarity of top contributors

Contributors	Top 10	Top 20	Top 30	Top 40	Top 50
High quality articles	89.60%	85.20%	82.85%	63.12%	61.23%
Low quality articles	65.02%	63.34%	42.24%	29.86%	25.01%

drops to 60 percent when the number of top contributors increases to 40. This is due to the fact that the top-30 contributors is close to the typical number of top contributors which amounts to 32 based on Equation 4.1. Accordingly, Table 4.3 shows the similarity between top-10 and bottom-N (N = 10, 20, ..., 50) contributors for different types of articles. Here, bottom contributors are the ones who have the least number of contributions. We found out that the similarity between top and bottom contributors is very small. In addition, top editors for high quality articles are more focused on specific topics compared to the top editors for low quality articles. Therefore, the similarity between top editors and bottom editors for high quality articles is smaller compared to low quality articles.

Table 4.3: Similarity of top 10 contributors with bottom contributors

Editors	Bottom 10	Bottom 20	Bottom 30	Bottom 40	Bottom 50
High quality articles	42.51%	40.89%	33.23%	28.23%	33.44%
Low quality articles	53.33%	51.98%	48.39%	44.78%	38.01%

4.3.3 Discussion

In this section, we analyzed the Wikipedia dataset for different types of articles based on the role of contributors. We found out that the main difference between low quality and featured articles is the number of contributions. High quality Wikipedia articles have more contributors than the lower quality ones. We noticed that for low quality articles, the number of contributions is very close to the number of revert back modifications. Therefore, articles in this group suffer from instability and low integrity.

For high quality articles, we observed that most contributions are performed by a small number of contributors who have similar interests. They form a small group with size around 32 and edit similar topics in Wikipedia. In addition, this small group of editors controls the high quality documents and is responsible for reverting back the edits made by others. In other words, we can claim that a small set of users have taken ownership of the featured articles because of their contributions and tend to revert back other's contributions. This results in higher quality and integrity in a small portion of articles in Wikipedia. We can observe that to have higher integrity in crowdsourcing systems, we need to have a permanent set of editors who are responsible for maintaining their contributed articles. For systems with open access such as Wikipedia, this can be a huge burden for the permanent editors. Therefore, we need to bring ownership and co-ownership in order to prevent low quality modifications. In the next section, we will introduce a new scheme which takes these observations into consideration.

4.4 Social Integrity Management Scheme

In this section, we present the Social Integrity Management (SIM) scheme for preventing unauthorized modifications in online crowdsourcing systems. First, we introduce the different concepts in the design of SIM and then discuss the rationale that supports the choices made in the design process.

In this scheme, users can view, vote, comment, edit and create (own) an article. We incorporate a number of social factors and ownership in our design to preserve the integrity of documents. It means that SIM utilizes user activities and the topological structure of the social graph to establish trust between the users. In an editing situation, an owner of the article is more willing to accept modifications from an editor who is considered trustworthy. We apply these factors in SIM to find the trust between the users and categorize users around the owner of the article.

Our approach for developing SIM is built on the following assumptions.

1. All users are part of a centrally maintained social network (e.g., Facebook).

2. Friendships among users on the social network are context independent (e.g., direct friends of Alice could include family members and university colleagues).

3. The social network follows the best security practices in resisting whitewashing [99] and Sybil attacks [100, 101].

4.4.1 SIM scheme details

In this scheme, we use an integrated namespace to facilitate discovery of articles for users. Users can search articles about a specific topic without even logging into the system. If

a user wants to vote, comment, edit or publish an article, she should be logged into the system with her unique username. Having unique identifications helps to prevent multiple votes on one article from a specific user. More votes on one article indicate that it has higher level of acceptance among the users. Therefore, there are no anonymous comments, editing or writers in the system.

Trust level

The design of SIM leverages the structural properties of the social network and the activities of its users. In social networks, the structure of the relationships can be used to infer the degree of trust between its users [102, 103]. In such cases, trust is represented using the social distance between the users (hop-distance in terms of user relationships) in the social network. For instance, direct friends (1 hop friends) are considered more trusted than friend-of-friends (2 hop friends).

To quantify the trust between two users, we propose a *trust level* measure. Every user should classify her friends in different categories based on the level of trust between them. Friends can have three different level of trust: highly trusted, trusted and untrusted. Let us denote the trust level between two users x and y as $L_{trust}(x, y)$. We assume that trust level is measured as a real number in $[0, 1]$. Therefore, the trusted level for *highly trusted friends* can be represented as 1, for *trusted friends* as 0.5 and for *untrusted friends* as 0.

SIM scheme supports transitive trust between users. The trust level between users x and y will be the transitive trust for the smallest hop distance between them. Assume that the hop distance between x and y is 2 with user z in the middle, then the trust level between x any y will be:

$$L_{trust}(x, y) = L_{trust}(x, z) \times L_{trust}(z, y) \tag{4.3}$$

The trust level between friends is automatically updated based on their activities in the system. These activities represent the history of the different editing activities that take place between the users. We will explain this process in detail in the voting section.

Set of trustable friends

Based on the trust level, SIM categorizes friends around each user into different groups. Each user will have a set of trustable friends as follows:

$$T_x = \{y \in U | L_{trust}(x, y) \geq \beta\} \tag{4.4}$$

where, U is the set of users in the system, T_x is the set of trustable friends for user x, and β is the threshold. It means that if the level of trust between users x and y is greater than some threshold, then y will be considered as the trusted friend of x. In SIM, the trust level between two users (e.g., x and y) is asymmetric. User x can trust y to modify its articles (e.g., $L_{trust}(x, y) \geq \beta$); however, user y may have not enough trust on x in to give editing permissions to x.

Co-ownerships

In SIM scheme, any user who accepts to modify an article is considered as the co-owner of that article with the same privileges. Therefore, the set of co-owners for a particular article includes the creator and all other users who have edited the article so far. The creator of an article is considered the originator and all co-owners are responsible for the credibility

and future modifications of the article. We incorporate co-ownerships in our model to prevent the article accuracy problem in existing systems such as Wikipedia. Since there is no anonymous modification, an article represents the opinion of the authors who put their reputation on the line. The users pay more attention to the accuracy of their articles; hence, the quality of articles can be significantly improved.

In addition, the co-ownership limits the editing capabilities to only trustable users. In open systems such as Wikipedia, users have to watch their articles and revert back inaccurate and unreliable modifications. This results in large efforts for users to keep the integrity of their articles. Having co-ownership provides incentive for other users to participate in the modification and evolution of an article. SIM scheme supports three different mechanisms for finding the set of trustable editors for a particular article, limited, semi-limited and open access.

- *Limited*: In this case, the editing access will be limited to users who are considered trustable by all co-owners of the article. In other words, the set of trustable editors will be the intersection of all sets of trustable friends for all co-owners.

- *Semi-Limited*: a user will be considered trustable if at least q percent of the co-owners consider her trustable. It means that the user has to be in the q percent of co-owners' sets of trustable friends.

- *Open Access*: a user will have editing privileges if at least one of the co-owner considers her as a trustable editor. That is the set of trustable editors will be the union of all sets of trustable friends for all co-owners.

Limited number of versions

Co-ownership makes the creator of an article to lose control over her documents, since no agreement is necessary for co-owners to modify the article. In addition, depending on who accepts to edit an article, we can have different sets of co-owners for one particular article. This results in having different versions for specific topic with different integrity levels. Number of versions for a particular topic can be either constant k (e.g., k = 3, 4, ...) or p percent of the total number of documents.

Voting

In SIM, there is a voting process to select the most popular articles for each specific topic. To obtain the highest number of votes, owners of an article push their article to their trustable friends to get their feedback and also involve them in the evolution of their article. When a trustable friend accepts to modify an article, she is considered as one of the co-owners. Therefore, the new co-owner as well as previous co-owners try to improve the article to get the highest number of votes. The more trustable friends get involved, the higher the chance that the article becomes one of the most popular ones.

After an article gets sufficient amount of contributions, any of the co-owners can nominate the article by listing it as a candidate for the most popular article in a particular topic. After the nomination of an article, it is flagged with a special tag called nominee and the community decides whether or not the article belongs to the most popular ones via a voting process. The voting process includes a voting period which the co-owners have time to advertise their article to get large number of votes. After a successful election, the SIM scheme presents the most popular articles based on the number of votes with the list of all co-owners.

Accordingly, the trust level between friends is automatically updated based on their activities in the system. These activities represent the history of the different editing activities that take place between the users. When a trustable friend, user x, accepts to edit a particular article automatically the trust level between her and all the co-owners will be updated based on the result of the voting process. If the article becomes one of the most popular ones, the trust levels between user x and all the co-owners will be increased by a constant value α :

$$L_{trust}(x, y) = L_{trust}(x, y) + \alpha, for \quad \forall y \in CO(A) \tag{4.5}$$

where A is an article and $CO(A)$ is the set of co-owners for A. This means that the initial trust level values that the co-owners assigned to her friend at the beginning of the friendship were accurate and now they can even have higher trust on her trustable friend. On the other hand, if the article cannot pass the voting process, the SIM scheme automatically decreases the trust level between the co-owners and the user x by a constant value γ:

$$L_{trust}(x, y) = L_{trust}(x, y) - \gamma, for \quad \forall y \in CO(A) \tag{4.6}$$

4.5 Experimental Results

We carried out a simulation study to evaluate the characteristics of SIM under different system configurations. In this section, we first describe the general setup of our model followed by a discussion of the details related to the evaluation of SIM. To setup the simulation, we used topologies extracted from Facebook [97]. The Facebook dataset is a collection from the New Orleans regional network with around $60,290$ users, $1,545,686$ friendships, and

$876,993$ interactions between the users. The interactions can be any wall post such as posting videos, photos, and comments.

Using the friendship traces from our dataset, we constructed a synthetic social network as the main social graph for our simulation. In this graph, if user u is friend with user v, user v is also friend with user u. However, the trust level between two users is asymmetric ($L_{trust}(u,v) \neq L_{trust}(v,u)$). The trust level between every two users is assigned randomly at the beginning of the simulation. The trust level is a real number between 0 and 1 to show the level of trust that user u has on user v as follows:

$$L_{trust}(u,v) = \begin{cases} < 0.4 & & & untrusted \\ > 0.4 & \&\& & < 0.8 & trustable \\ > 0.8 & & & highly \quad trustable \end{cases} \tag{4.7}$$

We selected 100 users randomly from the whole graph to generate the articles. We call these special users as creators (C) in our scheme. The creators are scattered in the graph to cover as many users as possible. Each iteration of our simulation has different steps. First, the 100 selected creators create new articles on random topics. Since the topics are chosen randomly, there is a possibility that two creators would generate two documents on the same topic.

The newly generated and existing articles are introduced to the highly trusted friends by creators or any of the co-owners. To find the set of trusted editors, we tried all the three different mechanisms mentioned in previous section (limited, semi-limited and open access). The limited and semi-limited techniques resulted in highly restricted participation with very small set of editors for each documents with an average size of 3. It is necessary for any crowdsourcing systems to allow crowd participation at a large scale. Therefore, the

results shown in this chapter are based on the open access mechanism.

After a user is selected to receive an article, she can accept or reject the editing request based on her workload. In our simulation, we assign a capacity for each user which represents the maximum number of documents that the user can edit. When a user's workload reaches the maximum capacity, she is considered to be busy and she starts rejecting any subsequent requests for editing. However, if the editing request is for one of the documents for which she is a co-owner, then the user will accept the request with some probability even after she has reached her full capacity. The busier a user gets the smaller the chances of her accepting an editing request.

Different versions of an article go through a voting process which selects the most popular ones. To simulate the voting process, we assign a global reputation for each user randomly. The reputation of a user represents the average trust assessed by a community of users. Therefore, the higher the reputation of the editor, the larger the chance that the article will be selected as one the most popular one. The k versions with the highest number of votes are selected as the most popular ones. In our simulation, we start with the case with only one version (k = 1) representing Wikipedia and continue up to 10 versions for each articles (k = 10). After the most popular versions are selected, other versions will be removed from the system.

Finally, we adjust the trust level between users based on the result of the voting process. If an article passes the voting process successfully, the trust level between the co-owners increases based on Equation 4.5. On the other hand, when an article is eliminated through the voting process, the trust level between the co-owners is adjusted based on Equation 4.6.

To evaluate our scheme, we perform different groups of studies. We first focus on the

Table 4.4: Ratio of writers to readers for different trust level

Trust level	1.0	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	0.0
writers/readers	0.90	2.12	5.32	10.18	18.96	30.32	39.59	55.94	67.12	83.45	100.00

characteristic of our scheme. We evaluate the accuracy of trust generated by our simulation. In addition, we calculate the availability of each user to find how users become busy or overwhelmed. Next, we find the average number of co-owners for each article and their resemblance in order to assess the integrity of the documents. Finally, we study the evolution of an article over time. We find out how an article would change throughout time and reach its maturity. We run our simulation 1000 times with different number of iterations ($NI = 1, 5, 10, ..., 100000$). The average results observed in 1000 times of simulation are presented in the next sections.

4.5.1 Characteristics of the SIM scheme

We first start by analyzing the characteristics of our scheme. In crowdsourcing systems, we face two important issues: the amount of contributions and the integrity of existing information. Here, we measure the contribution factor and in the next section, we measure the integrity of articles in our scheme. We measure the amount of contributions by computing the ratio of writers to readers. The writers here are the contributors and the readers are the consumers. For open systems such as Wikipedia, every reader has the capability to be a contributor and modify an existing article or create a new one. Therefore, writers to readers ratio for open systems such as Wikipedia will be equal to 1. On the other hand, for traditional publishing (e.g., book writing), the ratio will be close to zero, because the number of writers is very limited comparing to the number of readers.

In our scheme, each user is responsible for assigning a trust level for all her friends.

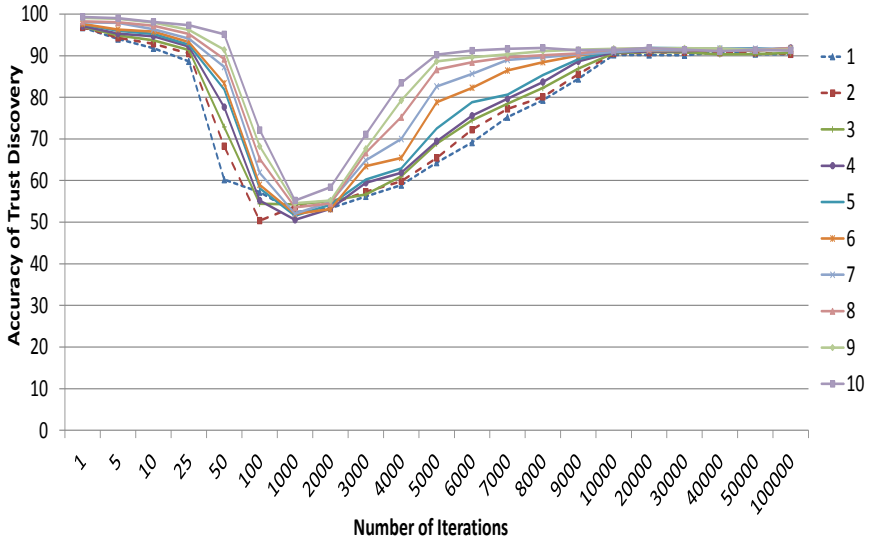


Figure 4.6: Accuracy of Trust Discovery

Some friends are considered highly trusted based on the trust level. We find that the number of users who are considered as highly trusted friends based on different values for trust level as shown in Table 4.4. By setting the trust level to 1, one can ensure that only highly trusted friends have access to an article with editing permissions. In this case, the ratio of writers to editors will be 0.9 percent. In other words, only 0.9 percent of the users may have the editing capabilities in the system. On the other hand, setting the trust level to 0 results in an open system such as Wikipedia in which every user can be a potential editor. In our simulation, users with trust level more than 0.7 are considered as highly trusted friends. Therefore the ratio of writer to reader our simulation will be about 10 percent.

Second, we look at the accuracy of trust assignment at the start of the simulation (this is also relevant in an actual SIM development, where the trust values need to be discovered). In SIM, the trust level between the users is adjusted based on the results of their editing activities. If the contribution of an editor increases the popularity of the article, then the trust level between the owner and the editor increases. On the other hand, the trust level can

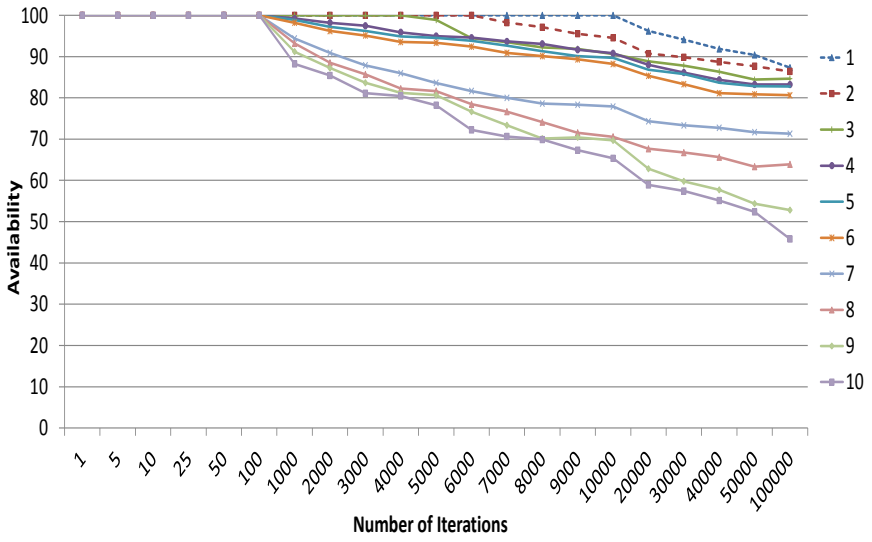


Figure 4.7: Availability

be decreased when the edited documents fail to pass the voting process. Figure 4.6 shows the variation of the trust level between users with the number of iterations. We observe that the trust level between users change up to 50 percent after their initial values. The accuracy of the trust values converges to within 90 percent after 9000 iterations. The accuracy of trust discovery never reaches 100 percent because in an editing situation co-owners of the article might select a user considered sufficiently trustworthy. Highly trusted users might reach their capacities for editing much quicker than others. Therefore, the co-owners have to consider editors with less trust levels. This leads to constant changes in the accuracy of trust in our scheme.

Finally, we find the availability of the highly trusted users in our system. Each article has the set of trustable editors. Here, we try to find the availability of trustable editors upon receiving an editing request. Up to 1000 iterations, the trustable editors are always available as shown in Figure 4.7. Afterwards, trustable editors might reach their capacity and their availability decreases

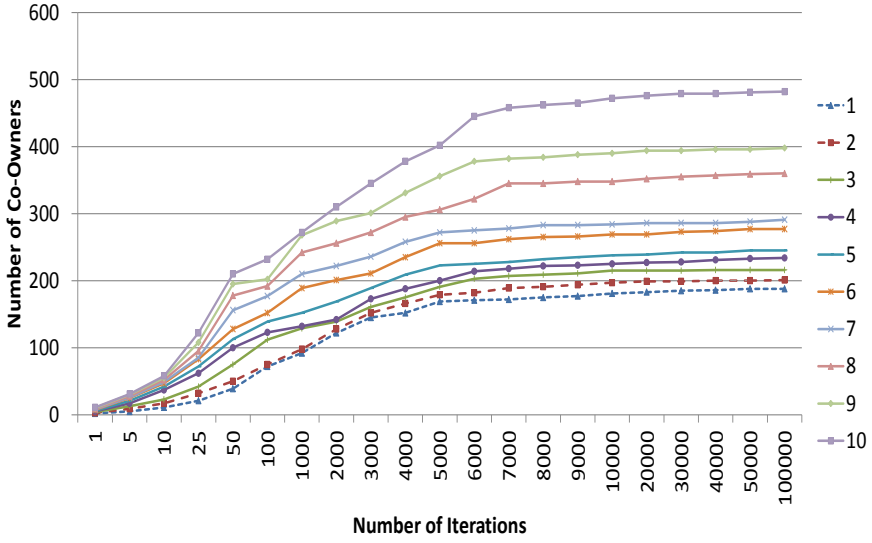


Figure 4.8: Variation of the number of co-owners with the number of versions

4.5.2 Analysis of co-ownership

In this section, we analyze the co-ownership in our scheme. Any user who contributes to an article is considered as a co-owner. The number of versions varied from one to 10 (k= 1, ..., 10). Figure 4.8 represents the average number of co-owners for each article in different iterations. The number of co-ownership increases at the beginning of the simulation and steadies after 8000 iterations. This means that most articles reach maturity between iterations 8000 and 9000. In the next sections, we will discuss article maturity in more details. Additionally, we notice that having more versions for each article result in larger number of co-owners. The number of co-owners with 10 versions is around 500 which is more than double compared to 192 for one version. With more versions, the articles can reach a larger group of contributors, therefore, the number of co-owners increases faster.

To estimate how stable co-ownership is in our scheme, we calculate the resemblance between sets of co-owners at different iterations of our simulation using Equation 4.2. The reason behind this analysis is that, in our Wikipedia experiments, we noticed that an article

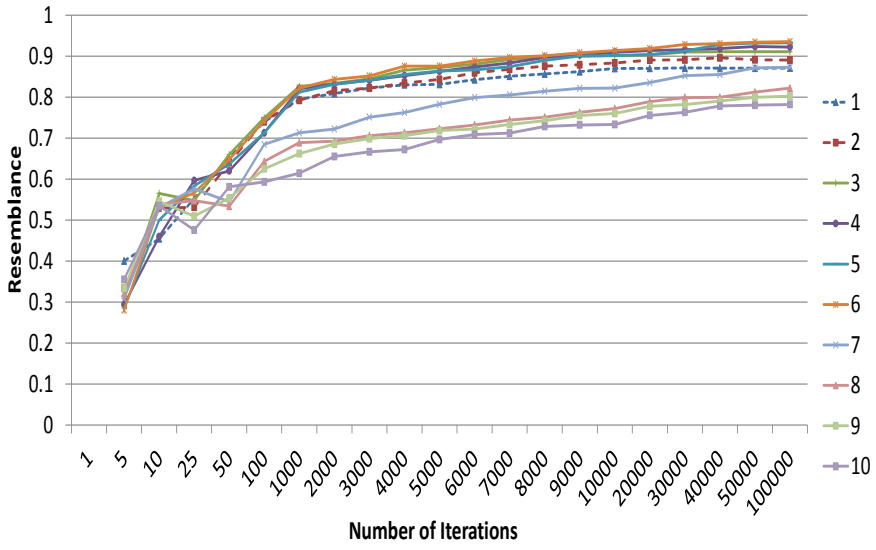


Figure 4.9: Resemblance between set of editors (comparing with previous iteration)

with a stable set of top contributors has higher quality (integrity) compared with an article without a set of top contributors. Therefore, we consider a stable set of co-owners as an indication of higher integrity.

Here, we calculate the resemblance in two different ways. First, we compute the resemblance between two consecutive points in our simulations. For instance, we find the resemblance between the set of co-owners in the 5000th iteration and the 6000th iteration. As shown in Figure 4.9, the resemblance between successive iterations increases with increasing number of iterations. The increasing trend in this graph shows that in each iteration of the simulation our scheme selects the best set of contributors; therefore the resemblance between co-owners becomes larger. However, after the 9000th iteration, the resemblance increases slightly. It means that there is a stable set of co-owners for most of articles generated at the beginning of the simulation. In addition, we notice that having six different versions of each articles results in higher resemblance between co-owners.

Second, we find the resemblance between the set of co-owners in the 1000th iteration

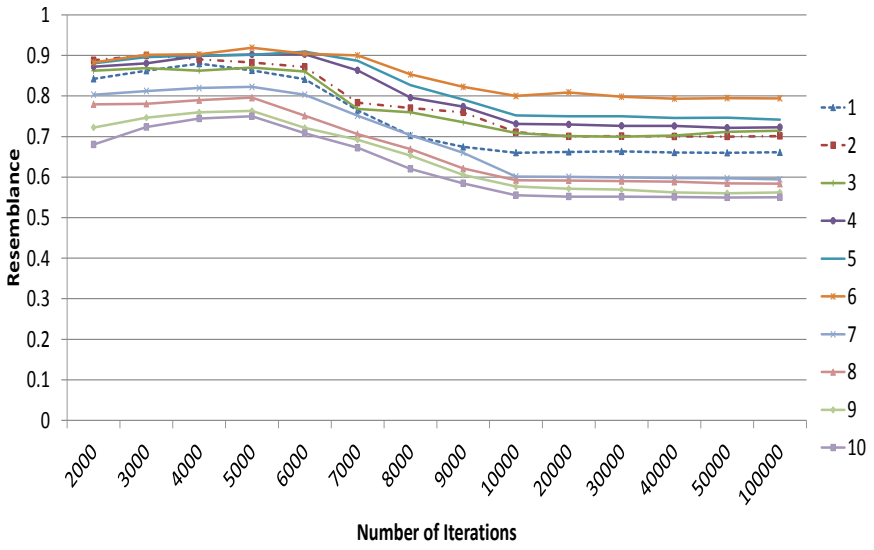


Figure 4.10: Resemblance between set of editors (comparing with 1000th iteration)

and the iterations after that. In other words, we try to find how the set of co-owners changes after the 1000th iteration. As shown in Figure 4.10, the resemblance starts high with a decreasing trend. Again, we notice that having six versions of each article gives higher resemblance. The resemblance with six versions is 0.8 compared that to 0.55 given by ten versions. However, after the 9000th iteration, we notice that the resemblance values have stabilized for all different number of versions.

4.5.3 Evolution of articles

In this section, we study how articles evolve over time and reach their maturity. An article is considered to have reached maturity when its co-owner set stops changing. Leveraging the observations we made in Wikipedia analysis, we consider the system to have reached convergence when the resemblance of the co-owner sets are high and they are relativity stable. As shown in the Table 4.5, the average resemblance is higher than 0.7 after the 10000th iteration for number of versions equals to 2, 3, 4, 5, and 6. Also, the resemblance

Table 4.5: Average resemblance at 10000th iteration

After 10000 iterations	
Versions	Average resemblance
1	0.66
2	0.70
3	0.71
4	0.73
5	0.75
6	0.79
7	0.60
8	0.59
9	0.57
10	0.55

values for number of versions 4, 5, and 6 is higher than the values for other number of versions.

Next, we measure the number iterations required for an article to reach its maturity. Here the resemblance is calculated as the similarity between two successive co-owners sets based on Equation 4.2. On average, it takes around 9000 iterations for an article to reach its maturity with resemblance more than 0.9. The resemblance value of more than 0.9 only occurs when the number of versions is 3, 4, 5 or 6 as shown in Figure 4.11. Figure 4.11 is similar to Figure 4.10 examined differently to show the maximum resemblance achieved for different number of versions. Here, we can see that reaching high resemblance, representing the convergence of the editorial sets, happens only when the number of versions are not too high or too low.

Each iteration in the simulation represents one contribution from an editor. From the Wikipedia analysis, we found out that for featured and good articles the resemblance is around 0.9 with an average of 10254 contributions which is very similar to our simulation results. In addition, we notice that the later the article is produced, the faster it becomes

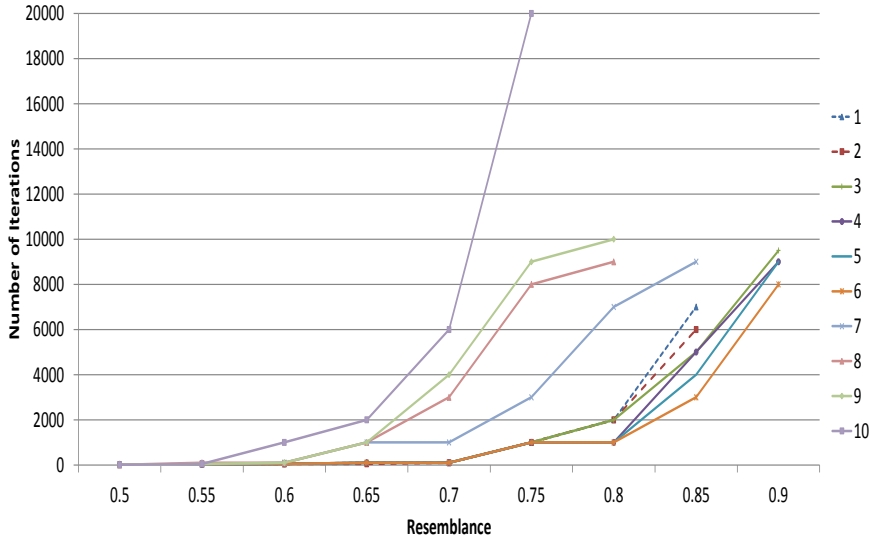


Figure 4.11: Number of iterations for an article to become mature

mature. For instance, an article generated at the 10000th iteration reaches maturity after only 5000 iterations. This is because after 10000 iterations the best editors are known and creators can try to get the top editors involved in the evolution of their documents.

Next, we find the average number of co-owners for articles when they reach their maturity. On average, mature articles generated at the start of the simulation have around 242 co-owners. However, articles generated later (e.g. the 10000th iteration) have fewer co-owners (e.g., 151 co-owners). Since the best editors are known, fewer users participate in the evolution of the articles. We also calculate the fraction of mature articles in our simulation. Figure 4.12 presents the total number of existing articles, the fraction of active and mature articles when the number of versions is set to 6. Here, the active articles are the ones that need more contributions to reach their maturity. We notice that after the 10000th iteration the fraction of mature articles becomes more than 0.4. These articles are removed from the simulation to avoid overwhelming the users.

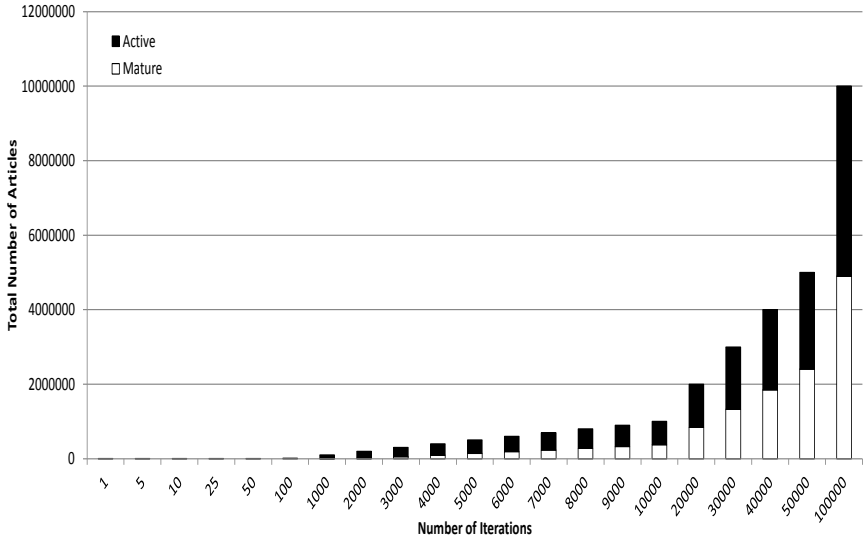


Figure 4.12: Variation of article maturity with iteration number

We notice that in all of our analysis having 6 versions per article leads to more promising results. To find the reason behind this phenomenon, we tried to change the density of our social graph by changing the average node degree. The average node degree for the Facebook dataset is 25 [97]. We kept the other properties of our social graph as much as possible while scaling the node degree. For instance, we make connection between a user and only her friends-of-friends to double the average node degree. In this case the node degree of every user becomes double and thus the total average node degree is double as well. In addition, users still are connected to their neighborhood. However, some characteristics of the social graph such as shortest path will change.

Figure 4.6 shows the number of versions for highest resemblance as the node degree is varied. The larger the average node degree, the smaller the number of versions per article we need to reach the best results. By reducing the average node degree to half (i.e., average node degree equals 12), we see the best results with 5, 6 and 7 versions per documents. On the other hand, we have the highest resemblance with three versions per

Table 4.6: Variation of the best number of versions with node degree changes

	Versions									
Ave node degree	1	2	3	4	5	6	7	8	9	10
12	0	0	0	0	9	80	11	0	0	0
25	0	0	0	0	23	81	3	0	0	0
50	0	0	0	0	25	74	1	0	0	0
100	0	0	0	0	37	63	0	0	0	0
150	0	0	0	0	39	61	0	0	0	0
200	0	0	0	4	40	56	0	0	0	0
250	0	0	0	4	47	49	0	0	0	0
300	0	0	0	5	54	41	0	0	0	0
400	0	0	0	6	58	36	0	0	0	0
500	0	0	1	9	62	28	0	0	0	0
600	0	0	4	14	59	23	0	0	0	0
700	0	0	5	18	57	20	0	0	0	0
800	0	0	5	27	50	18	0	0	0	0
900	0	0	6	31	48	15	0	0	0	0
1000	0	0	6	34	45	15	0	0	0	0
2000	0	0	6	37	42	15	0	0	0	0
3000	0	0	6	37	42	15	0	0	0	0

article when we have the average node degree of 3000. We notice that fewer versions are required to reach the best editors in the social neighborhood, when the density of the social graph increases (i.e., the larger average node degree). It means that with more connections between users (higher density in the social graph), it becomes easier to reach the best editors in the social graph. Therefore, we need fewer versions per article. On the other hand, when the social graph is parse, we need more versions to increase the chance of reaching the best editors. In addition, having more versions results in overwhelming the best editors and their availability decreases as shown in Figure 4.7. Therefore, reaching best editors becomes harder and as a result higher number of versions never leads to high resemblance.

4.6 Summary

In this chapter, we considered the problem of information integrity in online crowdsourcing systems. We presented challenges and requirements of any access control mechanism for preserving integrity by examining 12 different crowdsourcing systems. We observed that except for Wikipedia all other crowdsourcing systems include ownership in their design.

We analyzed Wikipedia in detail. We probed Wikipedia's integrity by modifying existing pages or adding new URLs. We noticed that the page integrity completely depends on the readers. Users in Wikipedia are responsible for finding unrelated references and removing them by reverting back the page. Therefore, incomplete and inaccurate information can remain in an article when the Wikipedia community has no interest in the article. Afterwards, we analyzed Wikipedia based on the role of contributors. We found that high quality articles have higher number of contributions performed by a small number of contributors who have similar interests. The top contributors are the key factor to have high quality articles. They control high quality documents and are responsible for reverting back the modifications made by other users.

Based on our findings from Wikipedia analysis, we introduced the new scheme called Social Integrity Management (SIM) for preserving integrity of articles in online crowdsourcing systems. The design of SIM uses co-ownerships as a key factor to control integrity of documents. The ownership bottleneck is relaxed by including co-ownerships and having multiple versions. In addition, SIM uses user activities to assign proper trust level between users.

Finally, we evaluated our new scheme by using extensive simulations on actual dataset from Facebook and comparing them with the results from Wikipedia analysis. In our Wikipedia experiments, we noticed that an article with a stable set of top contributors

has higher quality (integrity) compared with an article without a set of top contributors. Therefore, we considered a stable set of co-owners as an indication of higher integrity. We estimated the stability of co-owner sets by calculating the resemblance between sets of co-owners at different iterations of our simulation. In our simulation, each iteration represented one contribution from an editor. We found out that for high quality articles the resemblance of editors is around 0.9 with an average of 10000 contributions which is very similar to the results from our Wikipedia analysis. In addition, we studied the effects of social network topological properties on the number of versions per articles. We noticed that the number of versions per documents depends on the structural properties of social graph. The larger the average node degree, the smaller the number of versions per article we need to reach article' maturity with less number of iterations. We observed that fewer versions are required to reach the best editors in a social neighbourhood when the density of the social graph increases (i.e., a larger average node degree).

5

Characterizing User Behavior in Crowdsourcing Question Answering Systems

5.1 Overview

The Internet provides access to a vast information source. However, to gain access to this source users need to pose appropriate queries. Internet search engines such as Google work efficiently when discriminating keywords are included in the queries. Computational search engines such as Wolfram Alpha can decipher queries posed in natural language sentences. However, complex queries that use even simple paragraphs to explain a problem are beyond the capabilities of the best automated search engines. Such information access scenarios are still handled by human powered search systems (referred to as question answering services). There are many crowdsourcing question answering services on the Internet such as Yahoo! Answers, ChaCha, Ask.com, and Answerbag. While the above sites attempt to cover all possible topics, Stack Exchange sites are very specialized. They are a collection of sites where each site is specialized in a single topic area (e.g., TeX publishing). In this study, we chose to focus on Stack Exchange sites because anecdotal evidence suggests that they are highly effective in providing solutions to actual problems people encounter in

topics ranging from programming to cooking.

Although Stack Exchange sites do not have explicit social links that interconnect the people asking the questions or answering them, each user has a profile page. Profile pages of one-timers can be quite empty while others can have significant identifying information that can accurately point to the actual person. Therefore, Stack Exchange sites create some form of implicit social network where the interactions create the inter-personal links. There has been a lot of research on the measurement and analysis of Question Answering sites [25, 26, 27, 28, 29].

In this chapter, we used the dump datasets provided by Stack Exchange over a period of two years, August 2009 till July 2011 to study various user behavior related to question answering. We believe that more investigation into understanding the characteristics of such sites is necessary to evaluate the effectiveness of current systems and to design future systems. To the best of our knowledge, this is the first analysis of user behavior in crowdsourcing question answering systems by considering the effect of tagging, user reputation and user feedbacks. We conducted a survey among users of Stack Exchange sites and their feedback supports several of our findings. Of the 94 sites operated by Stack Exchange we selected 10 of the most popular sites in this study.

Section 5.2 explains the Stack Exchange sites and provides the details of our survey. Section 5.3 focuses on the data set and the key characteristics of the Stack Exchange sites. Section 5.4 and Section 5.5 present our analysis on user reputation and tags which are the features of the Stack Exchanges sites. Construction of the collaborative network among the co-answerers and its structural properties are discussed in Section 5.6. Finally, we summarize major findings of our analysis in Section 5.7.

5.2 Survey

Stack Exchange sites provide free services to their users, where the answerers or editors volunteer their time by contributing answers or modifications for other users' posts. In return, the answerers or editors gain reputation as they provide acceptable answers or modifications. By gaining reputation, users increase the level of capability they have in Stack Exchange sites from voting (e.g. 15 points) to full capability with reputation values equal or greater than 2000.

To participate fully in the Stack Exchange sites, users must register in them under a pseudo or real identity. A person may create multiple identities, and may even post questions or answers with different identities. We consider each of these identities as separate users. Users may volunteer information about themselves (e.g., name or age), which is added to the users' profiles.

To better understand user behavior in the Stack Exchange sites, we conducted a survey among actual users of these sites. We gathered e-mail addresses of more than 1000 users with different reputation scores (from very low to very high) from their profiles on the Stack Exchange sites. We sent our survey directly to the users and more than 100 users accepted to participate in our survey. In Figure 5.1, we show the distribution of the reputation scores of the participants. Users with reputation higher than $10,000$ points were the largest group of participants (33%). These users answered our survey very fast; however, low active users with reputation less than 200 were the smallest group of participant, only 15%. This is due to the fact that most of low active users have no information in their profile pages; therefore it was very hard to find their e-mail addresses.

The first question in the survey asked the users how they choose a question to answer in the Stack Exchange sites. The responses are shown in Figure 5.2. The majority of users

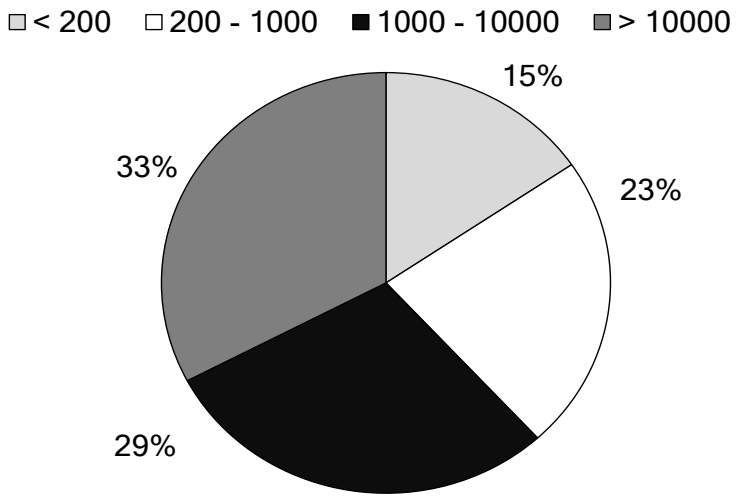


Figure 5.1: What is your reputation?

chose their questions based on the topic (e.g., Latex, programming, cooking). In addition, tags, quality of the question, and whether the question is on the first page or not are main factors for users while picking a question to answer. On the other hand, factors such as who asked the question, who have already answered the question, and the popularity of the question (number of votes), have relatively low impact on the way users choose questions to answer. The survey also found that users tend to answer questions without a lot of answers instead of questions with a lot of replies. There can be various reasons for this strategy. The questions with many answers are likely to have an accepted answer. Therefore, the likelihood of the user's answer gathering high number of votes is low.

Our survey had five more questions. We explain the question and the responses with the analysis of the relevant Stack Exchange data analysis in following sections.

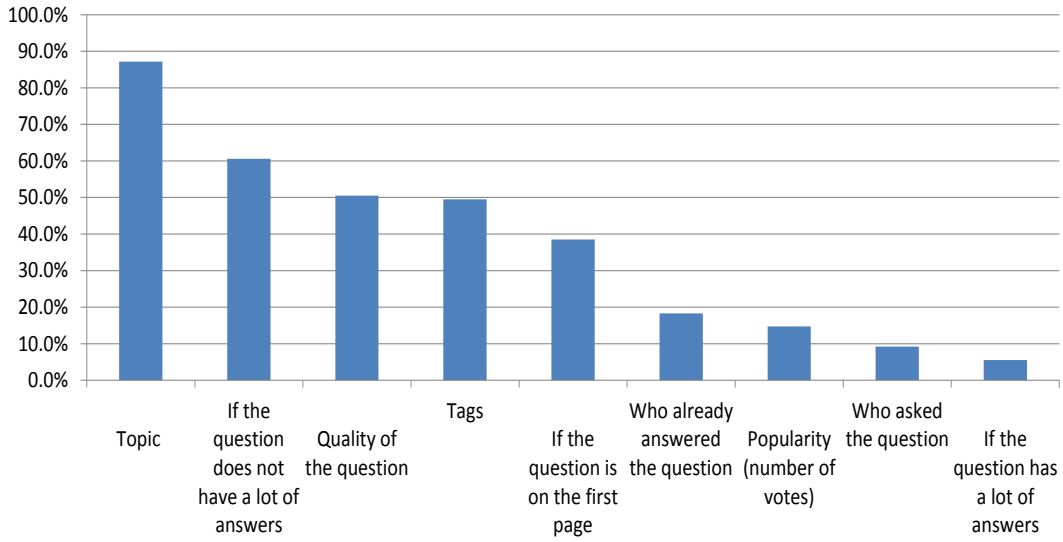


Figure 5.2: How does a user choose a question to answer?

5.3 Key Characteristics of the Stack Exchange Sites

5.3.1 Data Set

The datasets from the Stack Exchange sites have information for more than a million users, 2.2 million questions and 5 million answers for over a period of two years, August 2009 till July 2011. Because the data collection is a dump dataset, it contains all information about the users (e.g. name, reputation, age) and their activities (e.g. voting, making comments, editing questions or answers, posting questions or answers) and associated details. In our analysis, we only consider traces from the 10 most popular sites: LaTeX, Programmers, Cooking, Server Fault, English Language and Usage, Super User, Photography, Gaming, Ask Ubuntu, and Mathematics.

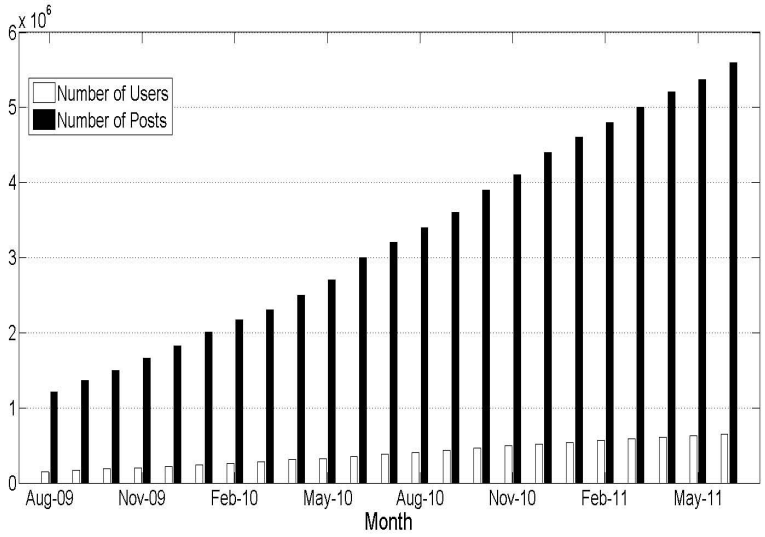


Figure 5.3: Evolution of the site over time

5.3.2 High-level characteristics

In this section, we present a high level characterization of the Stack Exchange sites. First, we use the creation date in the dataset to examine the growth of the number of users and posts over time. Figure 5.3 shows the evolution of the site from the beginning, August 2009, up to the June 2011. We observe a steady increase in both numbers of users and posts; however, the growth of numbers of posts is faster than the growth of the numbers of users.

Second, we determined the minimum, maximum, and average response times for the 10 sites. The minimum average response times are displayed in the format of "days: hours: minutes: seconds" as shown in Table 5.1 (when leading values are zeros, they are omitted for brevity). This illustrates the popularity of these types of sites because the minimum value of the average response times is very low. This means a question can receive answers as soon as it is posted on these sites. The Photography site has the lowest average response time of $00 : 02 : 50 : 53$ (2 hours, 50 minutes and 53 seconds) compared to the Super

Name	Minimum	Average	Maximum
LaTeX	23	00 : 17 : 27 : 02	094 : 21 : 57 : 19
Cooking	11	00 : 04 : 33 : 39	031 : 22 : 59 : 54
Server Fault	02	04 : 17 : 52 : 13	271 : 07 : 24 : 17
Programmers	18	00 : 11 : 30 : 57	275 : 02 : 47 : 48
Gaming	14	03 : 01 : 05 : 37	237 : 22 : 43 : 21
Ask Ubuntu	10	02 : 09 : 29 : 47	113 : 23 : 51 : 59
Photography	20	00 : 02 : 50 : 53	065 : 13 : 47 : 00
Mathematics	19	00 : 22 : 36 : 05	166 : 21 : 57 : 45
English Language & Usage	17	00 : 11 : 02 : 55	237 : 05 : 00 : 47
Super User	01	06 : 05 : 20 : 31	294 : 09 : 00 : 10

Table 5.1: Summary of the response times in Stack Exchange sites (days:hours :minutes:seconds)

User with the largest average response time of $06 : 05 : 20 : 31$. This phenomenon can be explained by the very nature of Photography and Super User sites. Because photography is widely accessible, most people have some experience and advice to share; therefore, they can quickly answer any question on the Photography site. However, in the Super User site, topics are about technical subjects and only experts are able to answer them. The Super User site contains more than 75000 questions and is therefore the largest dataset. The Photography site with 2000 questions is the smallest dataset. Another reason for the sluggish response times is the question discovery mechanism. All Stack Exchange sites follow the same question listing mechanism that lists the questions in pages, which can make it harder for an answerer to reach the most appropriate question that he/she could answer. Therefore, a more efficient question-to-answerer matching service might keep the response time low as the system scales up.

Third, we compared low active users who posted no more than one question making up for 54% of the users with highly active users who posted more than 10 questions or answers making up for 23% of the users. We were interested in understanding the reasons for the

Response time	Average	Minimum	Maximum	Average no. of Answers	Votes
Highly active users	06 : 58 : 24	0 : 0 : 23	62 : 1 : 45 : 45	3.06	6.16
Low active users	23 : 48 : 42	6 : 10 : 3	76 : 18 : 35 : 23	1.20	2.89

Table 5.2: Comparison between highly active users and low active users

low level of activity from a large portion of users. We noticed that the low active users post their first question over a month after joining the system whereas the high active users start their activities immediately after they become a member. In addition, the minimum response time for low active users is in order of hours whereas for highly active users, it is in order of seconds as shown in Table 5.2. Further, the average number of answers and votes for a question from highly active users is three times larger than a question from low active users.

In one of the questions in our survey, we asked "how often do you get a valid answer for your questions?" More than 66% of the participants answered that they "Always" or "Usually" obtain solutions to their issues and only 13% of users "Never" find proper answers as shown in Figure 5.4. This shows the effectiveness of the Stack Exchange sites in providing solutions to user issues. After filtering the answers, we realized that only users with high reputation answered "Always" or "Usually" and users with low reputation answered "Sometimes" or "Never."

These observations indicate that low active users got not so enthusiastic responses for their questions from the Stack Exchange community, which might have contributed to their lack of continued engagement with the site. Because user identity is not a significant factor in Stack Exchange sites, we can only surmise that questions posed by these did not gain much interest from the answerers.

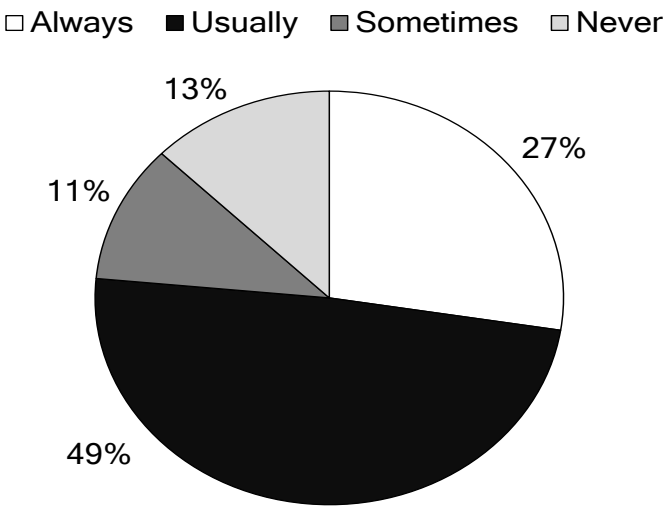


Figure 5.4: When you ask a question, how often do you get a proper answer?

5.4 Analysis of User Reputation

In this section, we analyze the correlation between the user reputation and their activities such as posting or editing questions and answers in the system. First, we investigate the overall distribution of the answerers by calculating the cumulative distribution function (CDF) of the number of answers for each user. Less than 30% of users answer all questions in the 10 sites as shown in Figure 5.5. Some users ($< 1\%$) posted more than 1000 answers.

5.4.1 Correlation between user reputation and number of posts

To analyze this phenomenon in more detail, we found the correlation between the user reputation and the number of posts (questions or answers) users have for all of the 10 sites. To do this computation, we only consider the users who answer the questions. Figure 5.6 shows the reputation of answerers versus the number of questions they answered. The higher the reputation, the more questions a user answer. There are a few number of users in the Programmers site who answered around 1000 questions. In all Stack Exchange

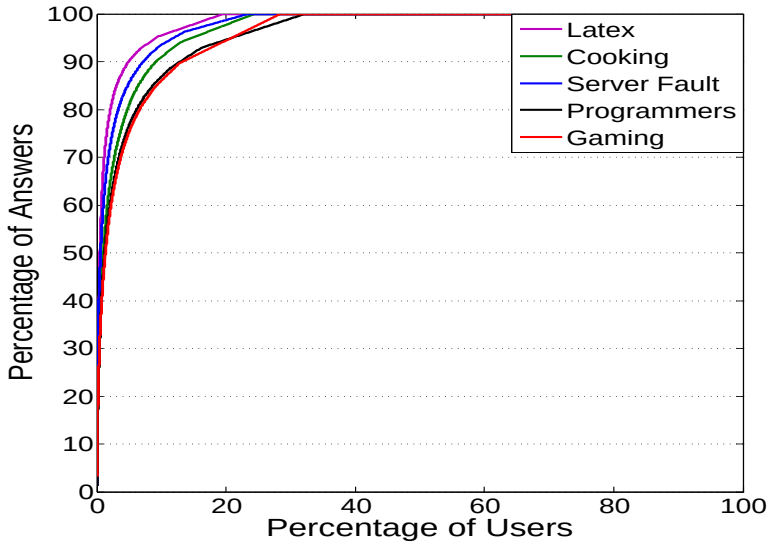
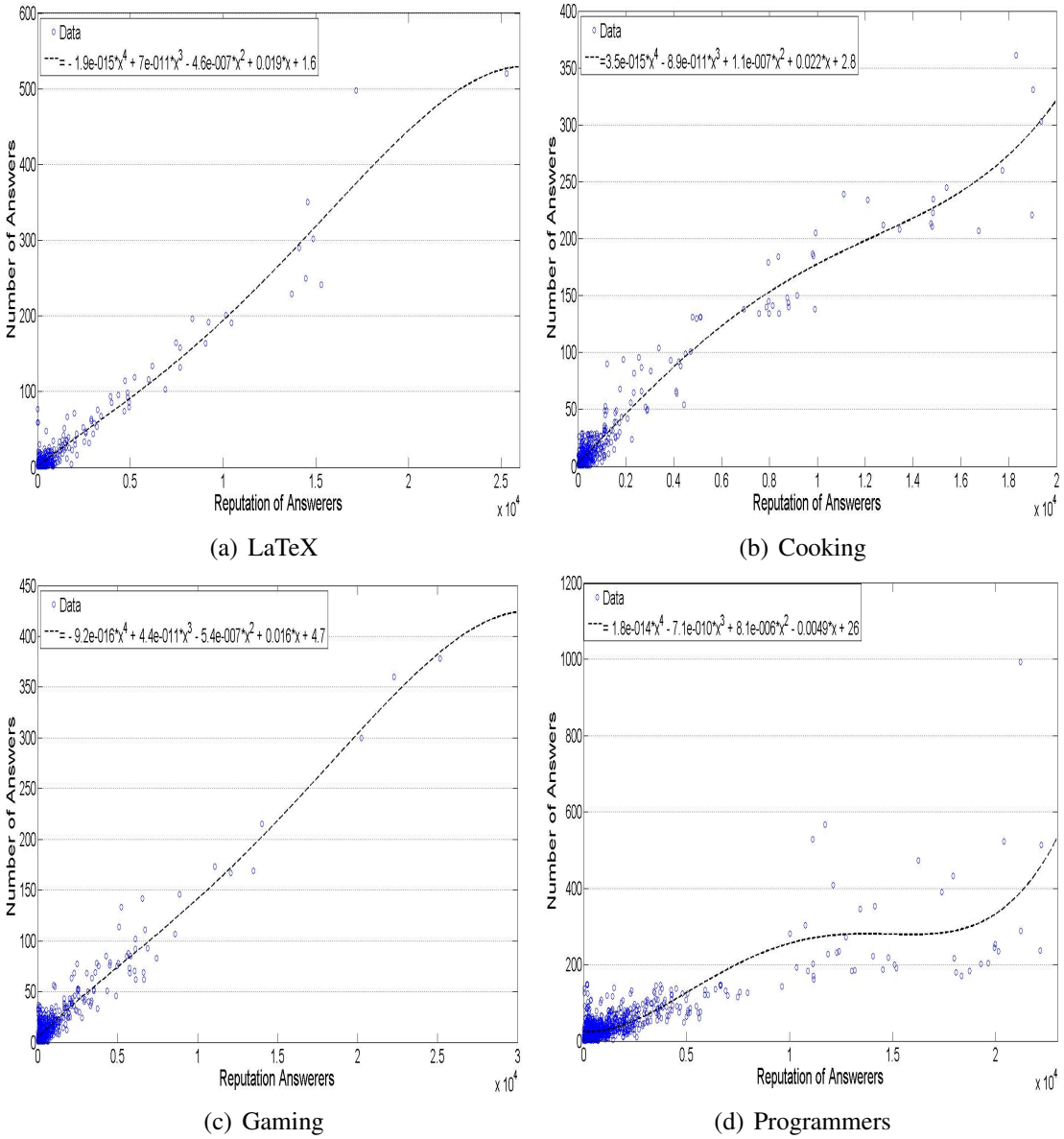


Figure 5.5: Users' activities cumulative distribution functions for number of answers

sites, users have to answer questions in order to gain more reputation. A user with higher reputation would have answered more questions than a user with lesser reputation. In addition, we observe that the growth rate in the Programmers site is small compared to other sites. This is because in the Programmers site there are wide variety of answers for a given question. For example, for each question on average there are more than 6 answers in the Programmers site while in other sites there are only about 2 answers per question.

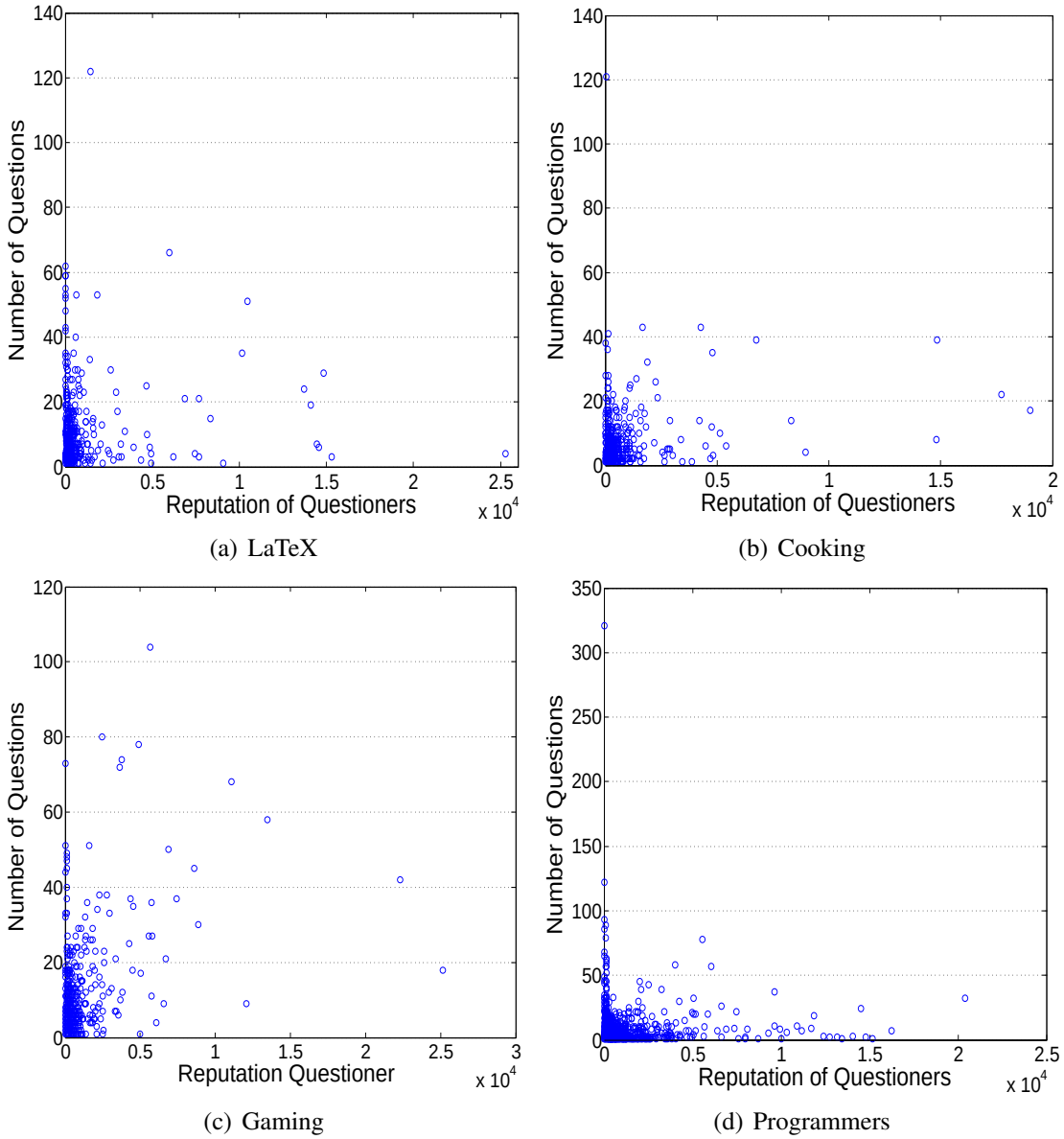
Similarly, when we examined the users who post questions in the system, as shown in Figure 5.7 we saw that most of the questions were posed by low reputed users. Figure 5.8 presents the distribution of questioners and answerers in LaTeX, Programmers, Cooking and Gaming sites. Because the corresponding graphs for the other sites look similar, we omit them. More than 90% of the questions were asked by relatively low reputed users as shown in Figure 5.8(a). In addition, Figure 5.8(b) shows that the majority of the answers came from a small portion of users in the system. Therefore, we can conclude that the information flows from highly reputed users to low reputed ones.



(a) LaTeX (b) Cooking

(c) Gaming (d) Programmers

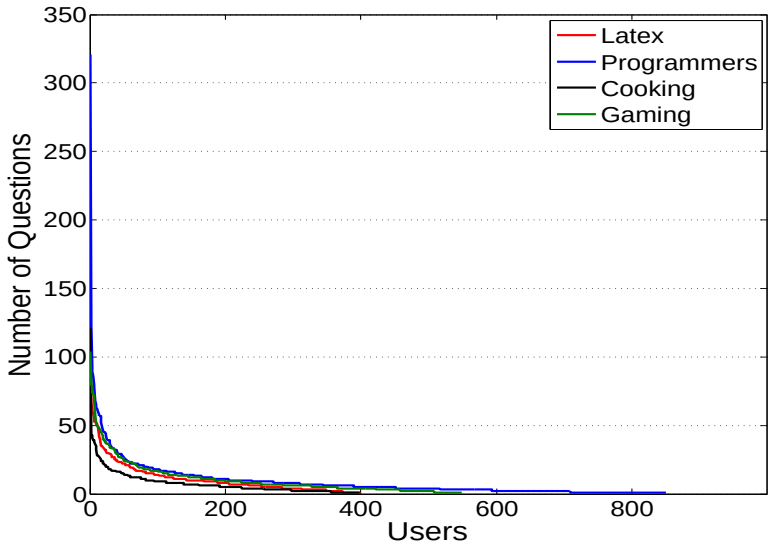
Figure 5.6: Reputation of answerers vs. number of answers



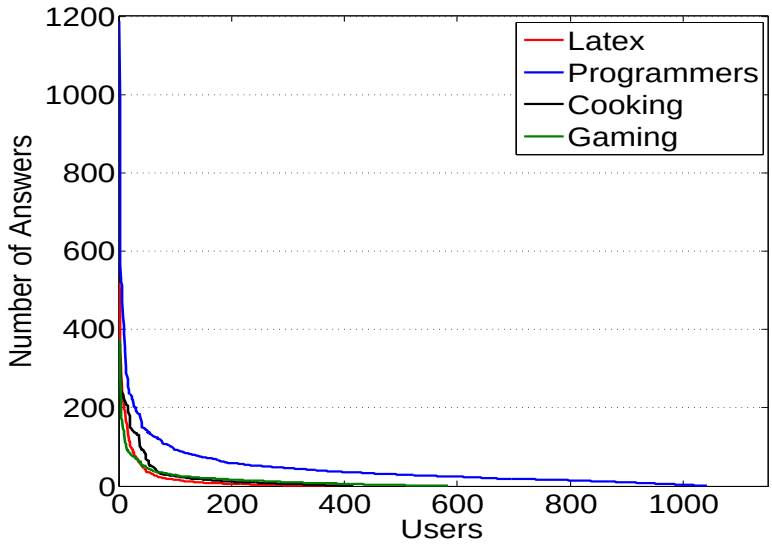
(a) LaTeX (b) Cooking

(c) Gaming (d) Programmers

Figure 5.7: Reputation of questioner vs. number of questions



(a) Distribution of questioners



(b) Distribution of answerer

Figure 5.8: Distribution of number of questioners and answerers

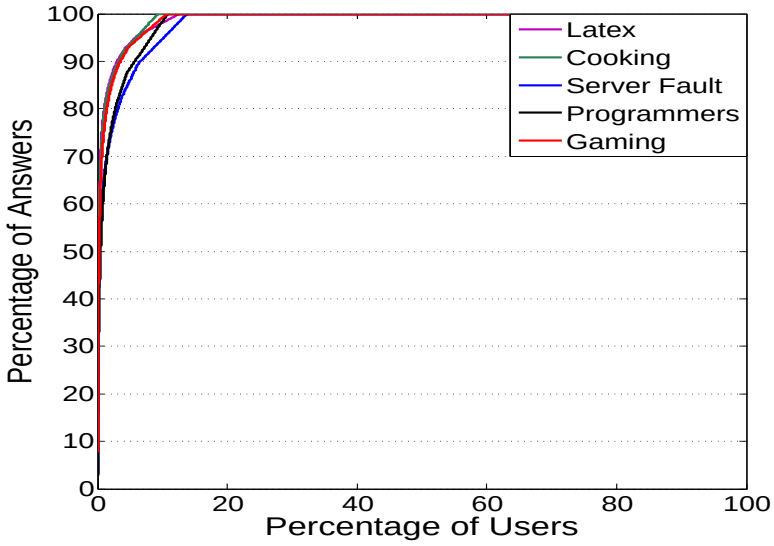
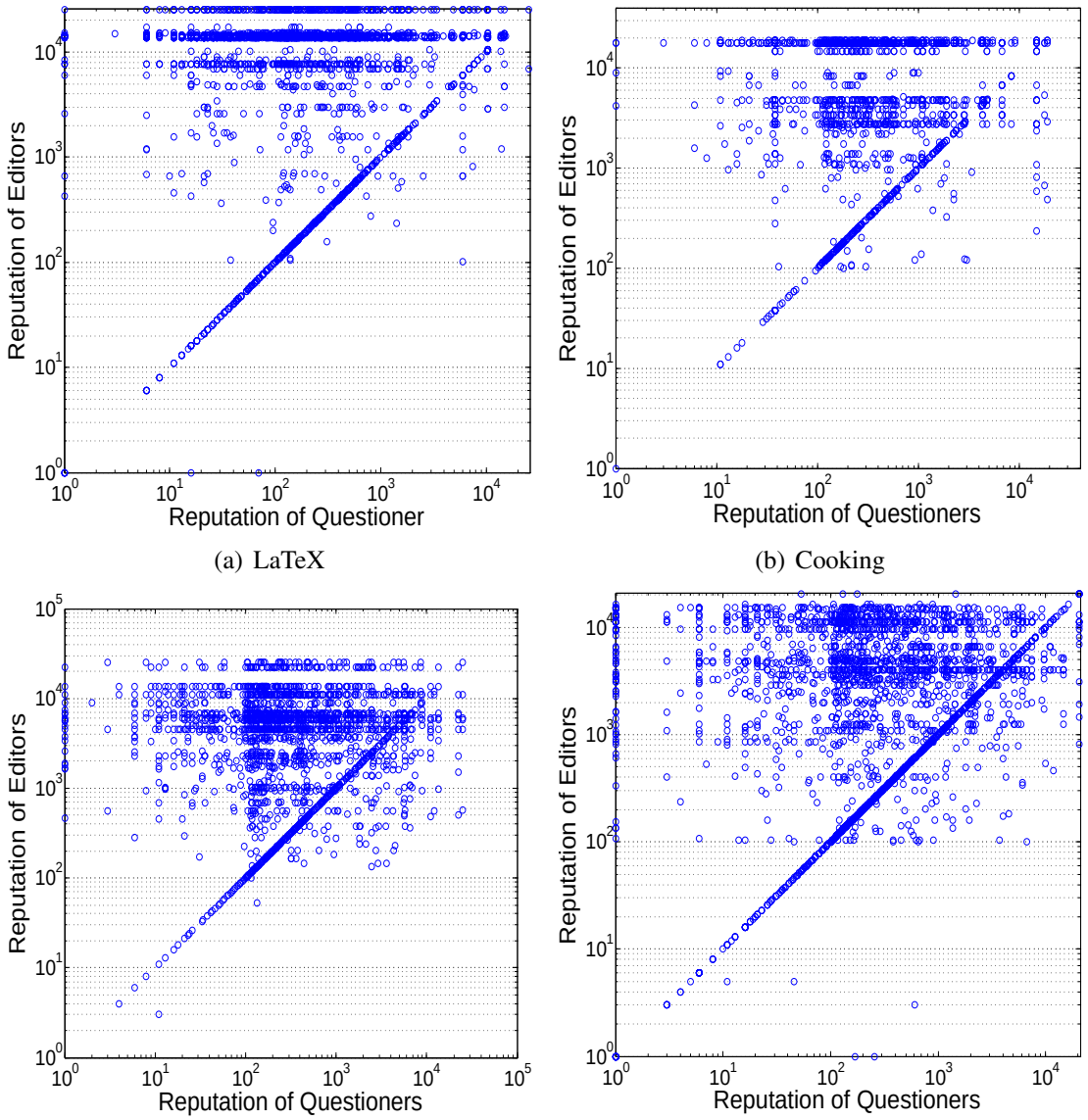


Figure 5.9: Users' activities cumulative distribution functions for number of edits

5.4.2 Correlation between user reputation and editing activities

Here, we focus on the correlation between user reputation and their editing activities. Around 12% of users participate in editing other users' posts as shown in Figure 5.9. Highly active users edit other users' posts in order to gain a better reputation. To analyze the users' editing activities in detail, we consider the relation between reputation of editors and questioners. The reputation of editors is most likely to be equal or larger to the reputation of questioners as shown in Figure 5.10. It means that the updates occur such that posters with low reputation are corrected by members with higher reputation. Similar results apply to all 10 sites.

Finally, we observe that all of the sites present activities according to power-law networks. Power-law networks are networks where the probability that a node has degree k is proportional to $k^{-\alpha}$, for large k and $\alpha > 1$ [104]. The parameter α is referred to as the power-law coefficient. Many real-world networks are shown to be power-law networks such as Internet topologies [105], the Web [106, 107], and social networks [108].



(a) LaTeX (b) Cooking

(c) Gaming (d) Programmers

Figure 5.10: Log-log: reputation of questioners vs. reputation of editors

Name	α	D	Name	α	D
LaTeX	1.63	0.0555	Ask Ubuntu	1.60	0.1022
Cooking	1.59	0.0585	English Language & Usage	1.74	0.0721
Server Fault	1.57	0.0823	Mathematics	1.68	0.0599
Programmers	1.59	0.0621	Photography	1.55	0.0623
Gaming	1.72	0.0732	Super User	1.50	0.1230

Table 5.3: Power-law coefficient estimates (α) and corresponding Kolmogorov-Smirnov goodness-of-fit metrics (D).

Our examination of the datasets indicate that the majority of the users obtain information from the system without making any contribution and that a few users provide most of the information in the system. We calculate the best power law fit using the maximum likelihood method [109] in order to test how well the users' activities distributions are modeled by a power-law [104]. Table 5.3 presents the estimated power-law coefficients for all the sites with the Kolmogorov-Smirnov goodness-of-fit metric [109]. This shows that the power law coefficient approximates the users' activities distributions perfectly without any significant deviations.

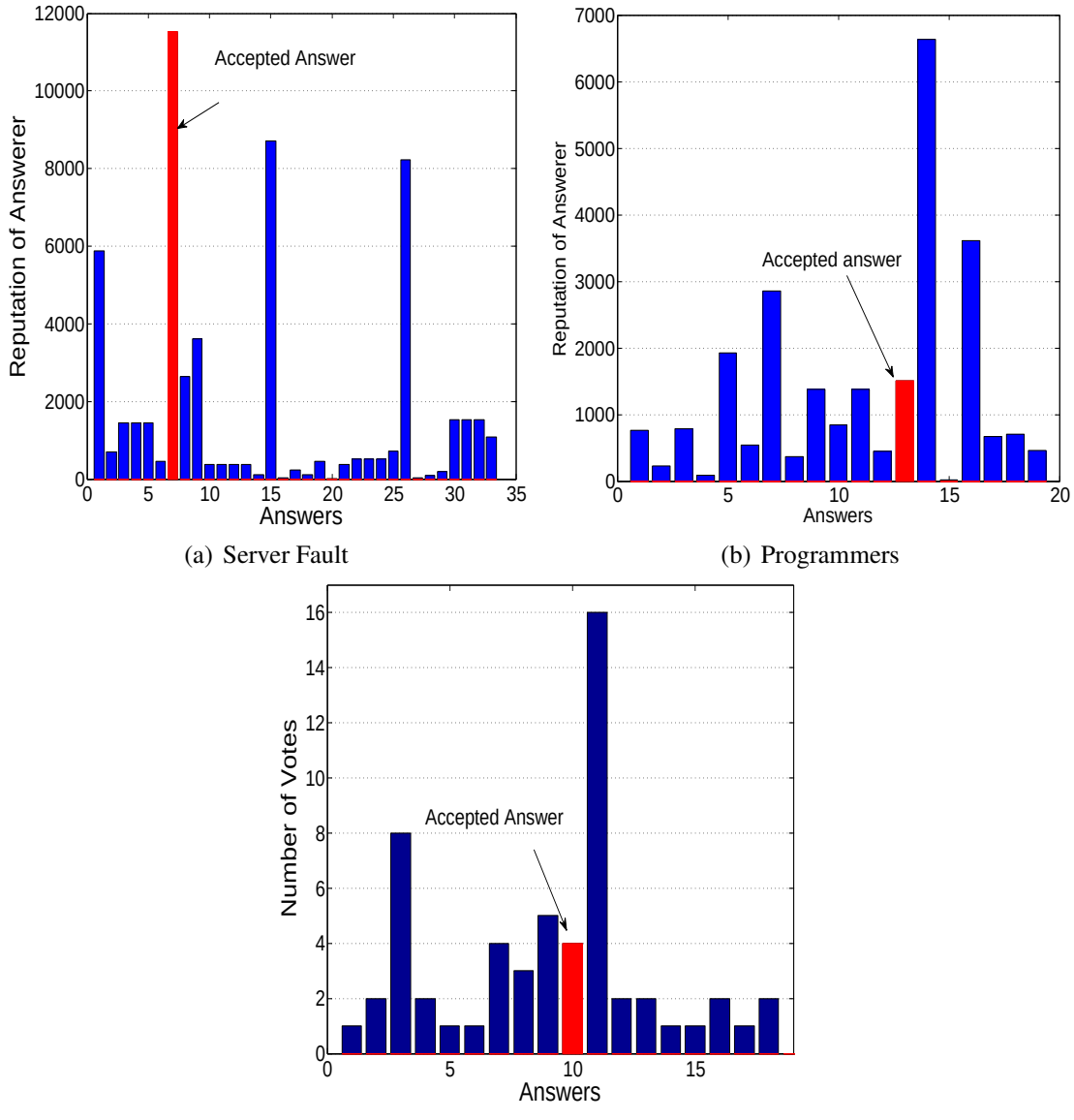
5.4.3 Correlation between user reputation and acceptance of answers

In this section, we examine the correlation between the user reputation and the probability that their answers will be the accepted ones. In Stack Exchange sites, users with various reputations can answer any question in the system. In addition, users can vote for their favorite answers which are presented based on their popularity. The questioner has the ability to accept any answer for her post: including the most popular one, the answer from a highly reputed user or the desired one. Figure 5.11 shows a particular distribution of answers to three different questions in Server Fault, Programmer and LaTeX sites. As shown in Figure 5.11(a), the answer number 7 from highly reputed users has more chance

to be the accepted one. However, the questioner has the ability to accept other answers with low popularity or from low reputed users. As shown in Figure 5.11(b), the questioner accepted the answer 13 from a user with a reputation of 1500 points while there is the answer 14 from a user with the reputation of 6800 points. Similarly, the answer with 4 votes was accepted while a popular answer exists with more than 16 votes as shown in Figure 5.11(c).

Figures 5.12 and 5.13 present the correlation between the user reputation and the probability that their answers will become the most popular or the accepted answer. To compute the probabilities, we divide the number of times a user's answers were selected as the most popular or the accepted answer by the total number of the user's answers. The greater the user's reputation, the larger these probabilities were. More than 65% of the answers from highly reputed users obtain the highest votes among all answers. Similarly, the probability that an answer from a highly reputed user becomes the accepted answer is more than 60%. It means that more than half of the times, answers from users with high reputation become the accepted one.

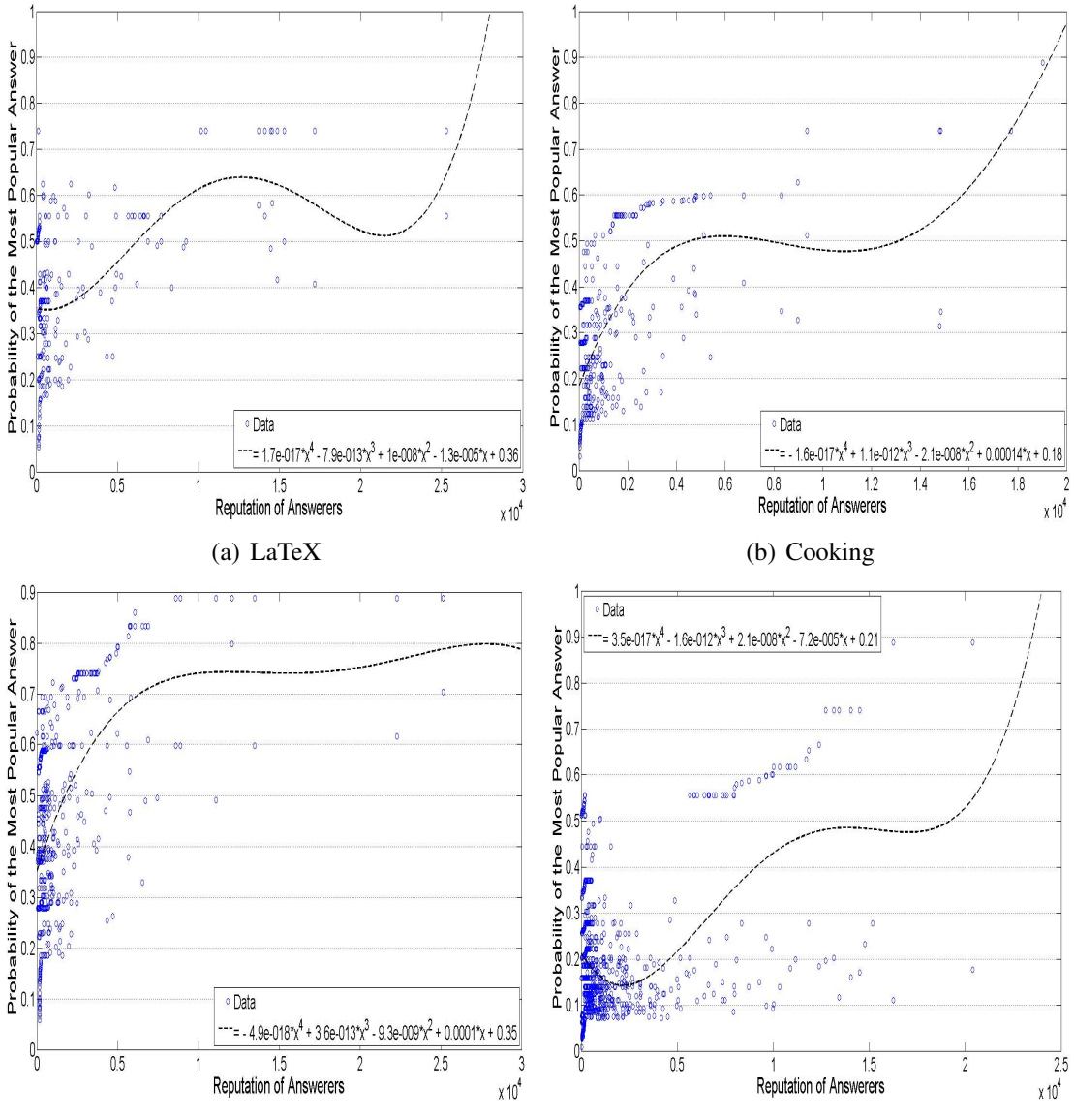
To better understand these findings, we gather additional statistics from the 10 sites. Table 5.4 presents the number of questions, answers, accepted answers, and accepted answers which are not the most popular ones. On average, 55% of the questions have accepted answers. The rest are still open. Among those accepted answers, only 13.7% of them are not the most popular ones in average. It means that the probability that the most popular answer will become the accepted one is more than 85%. In addition, we observe that Programmers site has the largest number of accepted answers that are not the most popular at 23.84% and Gaming has the smallest number of accepted answers that are not the most popular at 7.70%. This is because in the Programmers site, for each question, there are more than 6



(a) Server Fault (b) Programmers

(c) LaTeX

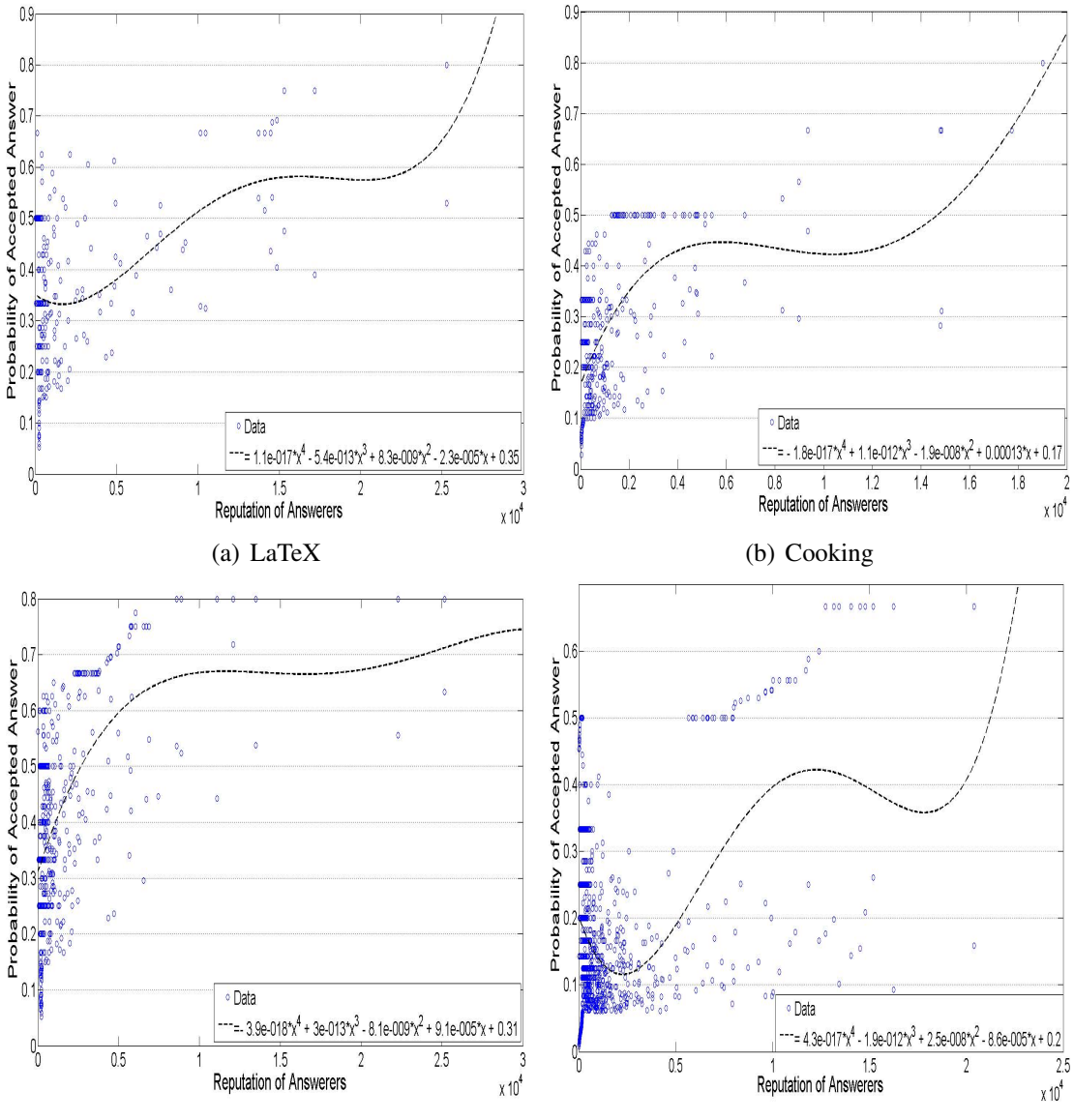
Figure 5.11: Sample question and the distribution of its answers



(a) LaTeX (b) Cooking

(c) Gaming (d) Programmers

Figure 5.12: Reputation of answerers vs. probability of most popular answers



(a) LaTeX (b) Cooking

(c) Gaming (d) Programmers

Figure 5.13: Reputation of answerers vs. probability of accepted answer

Name	Questions	Answers	Accepted Answers	Accepted Answers, not the most popular
LaTeX	4573	9229	3350	384(11.47%)
Cooking	3065	9820	2158	345(15.98%)
Server Fault	71962	162401	36726	4291(11.68%)
Programmers	8124	50790	3867	922(23.84%)
Gaming	5989	11702	4256	328(7.70%)
Ask Ubuntu	10113	20620	5538	599(10.81%)
English Language & Usage	4756	13182	3531	545(15.43%)
Mathematics	9580	20488	5978	626(10.47%)
Photography	2180	7823	1501	254(16.92%)
Super User	75562	183422	40221	5062(12.58%)

Table 5.4: Statistics from the 10 Stack Exchange sites

answers whereas in the Gaming site, for each question there are around 1.9 answers. This shows the variety of answers in the Programmers site.

5.5 Analysis of Tags

In this section, we investigate the use of tags in answering and asking questions in Stack Exchange sites. Figure 5.14 shows the distribution of tags based on the number of times they were used by users. Few tags were used significantly more times and the majority of tags were used only once. It means that this distribution has the power law property with the power-law coefficient of 1.59 and a deviation of 0.0272. The median number of times that tags were used is 5 and around 80% of tags were used not more than 10 times. However, some tags ($< 1\%$) were used more than 500 times. In fact, the most popular tag was used 702 times to tags different questions.

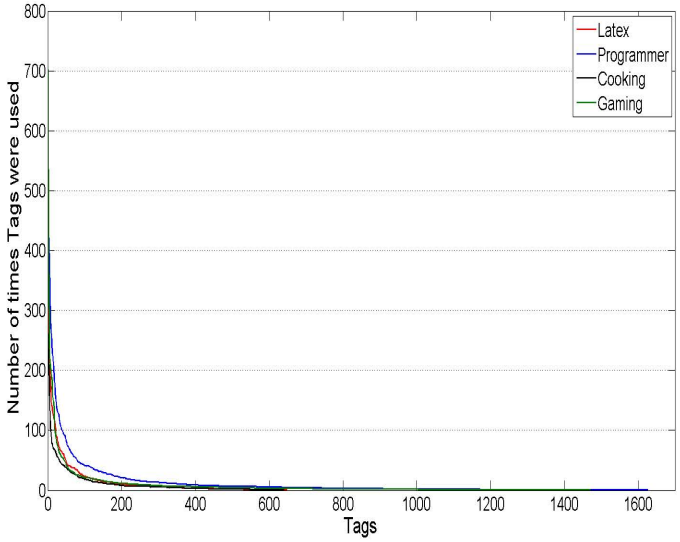


Figure 5.14: Distribution of tags

5.5.1 Tags popularity

To analyze the popularity of tags in more details, we classified them in ten different groups based on the number of times tags were used by users to tag their questions. For example, there are the one time tags (rare tags) which were used only once in the system to tag a question. The other groups consist of the two times tags, two to four times tags, ..., 126 to 256 times-tags and the larger than 256 times-tags (very popular tags) which were used to tag more than 256 questions. In the system, the answers to a question automatically inherit the tags of the question.

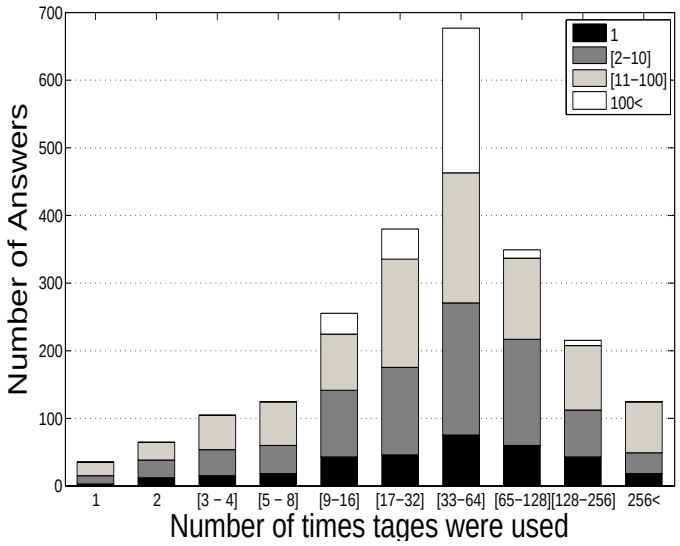
We found the number of answers that were given for the questions in each group and normalized it as shown Figure 5.15. For example, there are around 700 answers for the group of 9 to 16 times tags for the cooking site. We categorized the answers by regrouping the users according to the number of answers they provided. Each category is identified with a different color. For example, 87 out of 700 answers were given by users who answered only one question [represented in black]. A large portion of the answers came from

users who answered between 11 to 100 questions [represented in light grey], and only a few users answered more than 100 questions [represented in white].

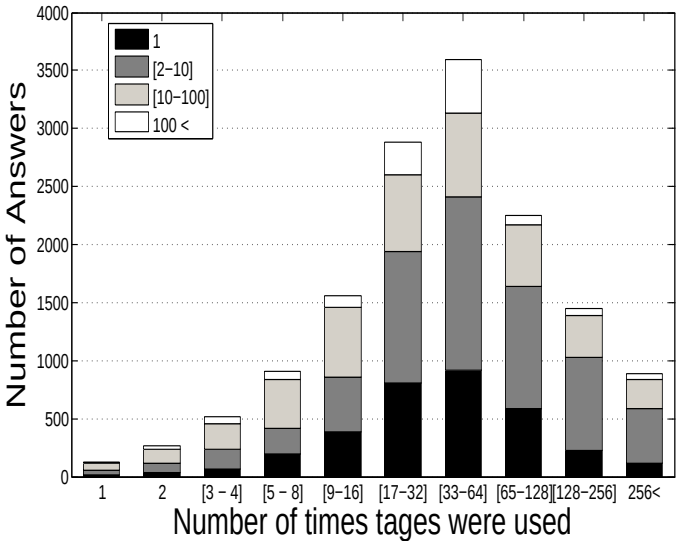
As shown in Figure 5.15, 33 to 64 times-tags are the most popular ones. We observe that these tags are very popular amongst the users who want to obtain as much reputation as possible. More than 84% of the users with mainly high and low reputations are interested in answering questions in popular topics (i.e., questions tagged with popular keywords) in order to gain more reputation. On the other hand, questions tagged by not so popular tags are mostly answered by medium reputed users.

5.6 Collaborative Network

In this section, we study the patterns of collaboration between pairs of users while answering questions. Because there is no notion of friendship in Stack Exchange sites, our goal is to understand whether users have collaborations in answering questions or they simply follow tags and form crowds. In general, it seems that users utilize tags as their primary signaling mechanism in order to notify others about specific questions. Here, we try to understand how these collaborations between pairs of individuals work and evolve over time. Towards this goal, one of the question in our survey asked whether users follow any specific user(s) in Stack Exchanges sites. More than 25% of users accepted that they follow other users in these sites. It means that if the specific user(s) asks or answers a question, these users will also try to answer that question. Therefore, we construct a collaborative network by using answers for each question as an undirected graph, where a link exists between a pair of users if they answer at least an α (threshold) number of same questions. In our experiment, if two users answer at least the same 10 questions, they would be adjacent in the collaborative network. To analyze the collaborative network in details, we conduct two



(a) LaTeX



(b) Programmers

Figure 5.15: Popularity distribution of tags

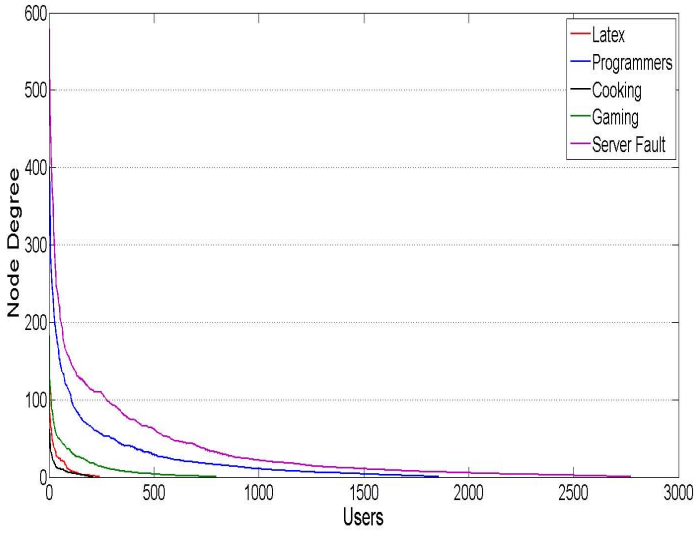


Figure 5.16: Distribution of node degree in the collaborative network

sets of experiments. First, we investigate the characteristics of the collaborative network. Second, we study the impact of the dynamics of the user activity on the global structure of the collaborative network.

5.6.1 Analysis of the collaborative network structure

We first examine the structure of the collaborative network by considering the node degree distribution. A large portion of the users (up to 84%) tend to co-answer questions with other users, thus the node degree in the collaborative network is not significantly lower than in the social network. We observe that the largest node degree is around 600 in the Server Fault website. As shown in Figure 5.16, the node degree distribution in the collaborative network conforms to power-laws with the coefficient of 1.59 and with the Kolmogorov-Smirnov goodness-of-fit metric of 0.0286. The majority of users have small degrees, and few users have significantly high degrees.

Next, we explore which users tend to co-answer questions with each which other users.

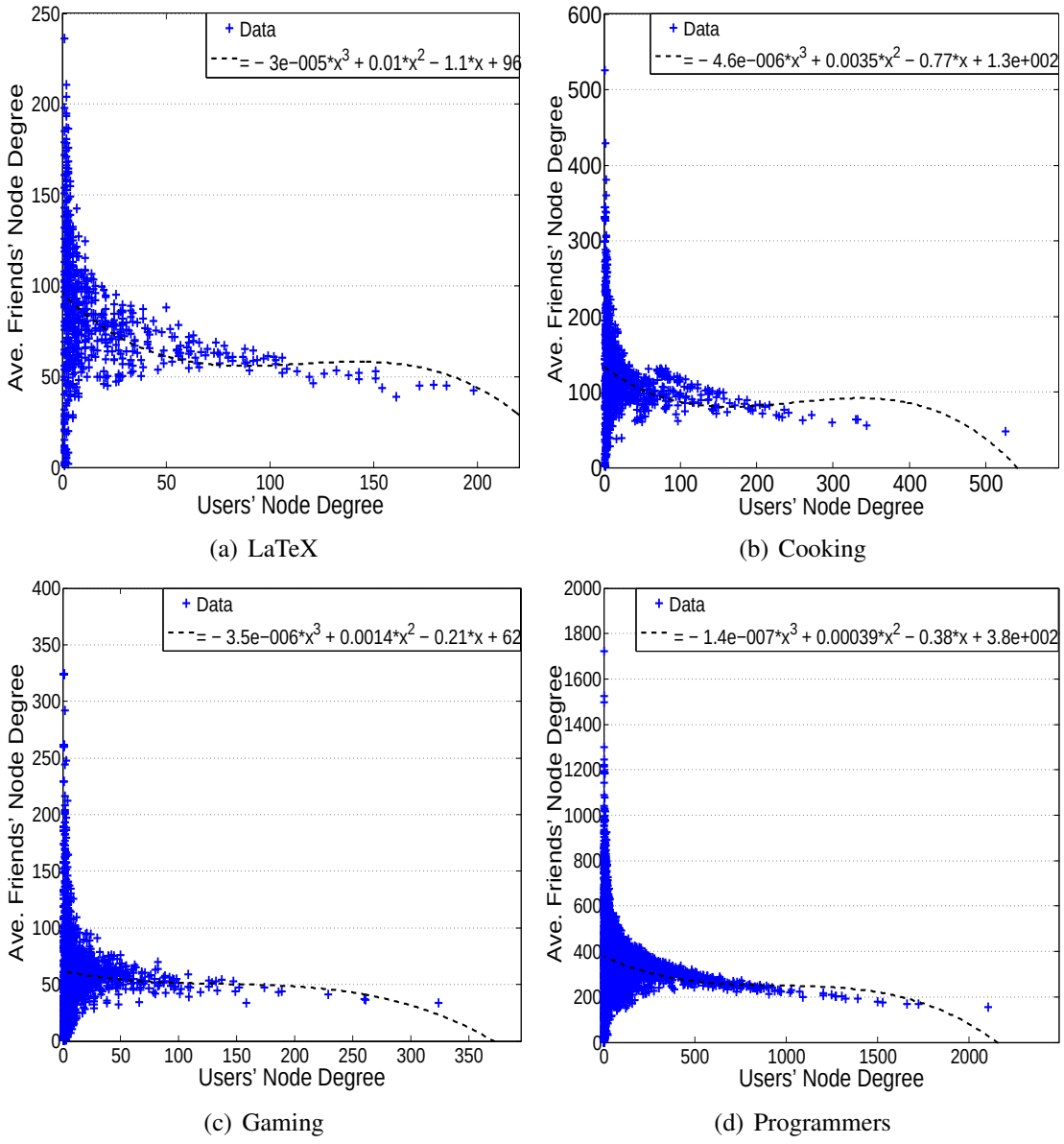
We try to find the joint degree distribution (JDD) (2K-distribution [110]) of the network. JDD presents how often nodes with different degrees connect with each other. For instance, in most social networks, high-degree nodes tend to connect to other high-degree nodes [104].

Here, the joint degree distribution is approximated by the degree correlation function between pairs of connected nodes. This function is a mapping between the node degree of users and the average node degree of all their friends. The larger value for JDD presents the tendency of high-degree users to connect to other high-degree users and a low value for JDD shows that high-degree users tend to connect to low-degree users. Figure 5.17 presents JDD in collaborative networks derived from 4 sites. We observe the same patterns for all of the 10 websites as high-degree users tend to connect to high-degree users. In the collaborative network, we observe that the high-degree users tend to connect to other high-degree nodes and form a "*core*" which is very similar to social networks [104]. It means that high-degree users have collaborations in answering questions and they follow tags to form crowds in Stack Exchange sites.

5.6.2 Evolution of the network over the time

In this part, we study the evolution of the collaborative network over time. We try to figure how collaborations between users start and finish, and also to what extent the collaborative network is stable and keeps its overall properties. To simulate the evolution of the network, we use a total of 12 time slots from July 2010 to June 2011. Each time slot contains information on users who have activities during the period given by the slot.

First, we examine to what extent the activity network changes overtime. We calculate the average node degree for each of the 12 time slots. Figure 5.18 presents how the average



(a) LaTeX (b) Cooking

(c) Gaming (d) Programmers

Figure 5.17: Joint degree distribution for all users in the collaborative network

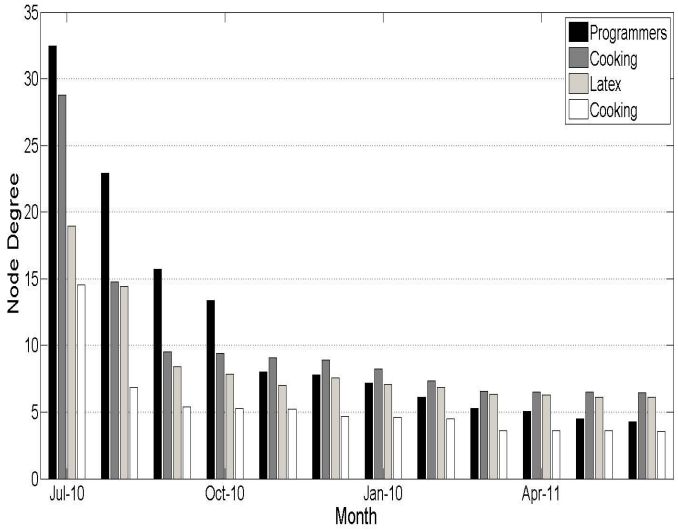


Figure 5.18: Evolution of node degree over time in the collaborative network

node degree evolves over the two consecutive time slots. In all sites, the average node degree starts with high value, decreases rapidly, and then remains stable overtime. This is because when these sites started to work, they contained small numbers of questions. Therefore, users could go through all of the questions easily and answer as many questions as they could. As the site became popular, the number of questions have increased to the order of thousands making it impossible for users to browse let alone answer all of the questions they are capable of answering.

Second, we explore what fraction of the collaborations between users persists from one time slot to the next time slot. We use the notion of resemblance [111] to measure the similarity between networks in two consecutive time slots. We define resemblance as the proportion of the network connections that remain unchanged over two time slots. Let denote R_t resemblance of the collaborative network at time t [111]. R_t can be defined as:

$$R_t = |\frac{C_t \cap C_{t+1}}{C_t}|$$

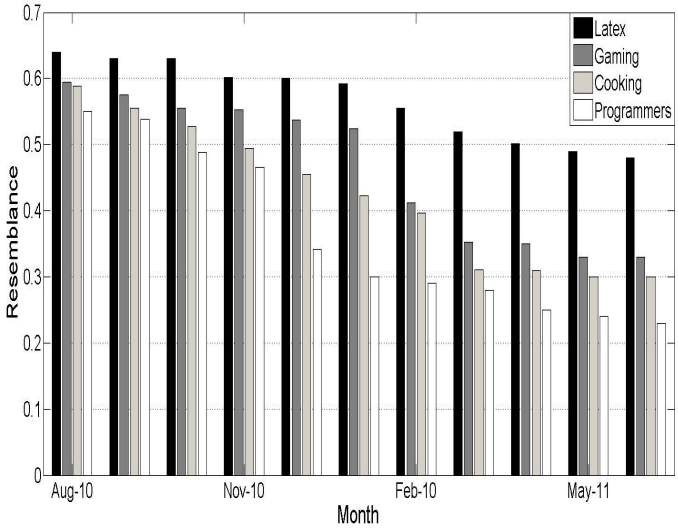


Figure 5.19: Resemblance of the collaborative network through time

where C_t is a collaborative network at time t. The value of R_t varies between 0 and 1 where $R_t = 1$ shows that the entire users pairs continued to collaborate at the next time step, and $R_t = 0$ represents none of the users who co-answered questions at time t collaborated in time $t + 1$.

Figure 5.19 shows the resemblance values of 4 different sites over time. We observe that the resemblance has a higher value at the beginning and decreases gradually. For instance, the resemblance for the LaTeX site starts at 64% and ends at 49%. This indicates that the collaborative network becomes more dynamic, since the growth of Stack Exchange sites is rapid as shown in Figure 5.3 and that more users participate in answering questions.

In addition, the average resemblance across all the time slots and sites is around 49% as shown in Table 5.5. This indicates that 49% of the users continue their collaborations and co-answering over the entire time period. Also, the resemblance is sensitive to the size of the time slot. Here, we chose a time slot of one month. If we use larger time slots (i.e. a season, three months), the average resemblance increases to 65%. This means that around

	LaTeX	Cooking	Gaming	Programmers	Server Fault
Average node degree	8.59	8.58	5.45	12.29	9.92
Average resemblance	0.58	0.43	0.48	0.41	0.54

Table 5.5: Average node degree and resemblance for different sites

35% of the collaborations disappear even over a large time scale. Conversely, when we use a smaller time slot (i.e. two weeks), the resemblance decreases from 49% to 35%.

5.7 Summary

In this chapter, we analyzed the characteristics of crowdsourcing question answering systems using datasets from 10 popular Stack Exchange sites. We chose Stack Exchange collections because of their popularity and their effectiveness in providing solutions to user issues.

We used the dump datasets to study various user behavior related to question answering. From our investigations, we made the following important observations. First, users with high reputation values are interested in answering questions tagged with highly popular keywords. Similarly, users with low reputation values also seek questions tagged with highly popular keywords. Questions with not so popular keywords are mostly answered by users with medium reputation values. Therefore, we consider some of the answerers as reputation seekers. Their primary motive is to gain high reputation as quickly as possible by looking for popular keywords where their expertise lies or are likely to get more votes. However, the medium reputation users are willing to genuinely help other users in Stack Exchange sites by sharing their knowledge.

Second, although Stack Exchange sites have no notion of social links, we observe crowd behavior in question answering. That is answerers tend to co-answer questions with each

other. We notice that users can be gyrating towards few tags and this effectively results in constructing a crowd. Therefore, tags become the signaling mechanism in order to gather a crowd together and form networks we refer to as collaborative networks. Further, we observed that the co-answering patterns change over time (i.e., over the course of one month about 51% of users co-answering in a collaboration setting disappear). However, the graph theoretic measures (e.g., average node degree) of the collaborative network formed by co-answerers remain stable over the course of one year.

Finally, our survey showed that question topic (e.g. programming, cooking, photography) is the most important factor for users in choosing a question to answer. Tags, quality of the question, and if the question is on the first page are other major factors. In addition, we notice that users prefer to answer questions without many replies since the chance that their answers will be the accepted or become the most popular one is high for such questions.

6

Related Work

In this chapter, we discuss previous work related to the research problems we have addressed in this thesis. First, in Section 6.1, we focus on other work related to the problem of confidentiality of information sharing in online social networks. We mainly review different access control system proposals and compare them to the α-myCommunity scheme. Second, Section 6.2 discusses the work related to the integrity problem for crowdsourcing systems and compares our solutions with their proposed schemes. The third part of this chapter (Section 6.3) reviews literature dealing with the analysis of online social systems. We highlight major similarities and differences between our approach for analyzing a crowdsourcing system and the discussed work.

6.1 Confidentiality Control

The goal of the α-myCommunity scheme presented in Chapter 3 is to provide the users control over information sharing activities in large scale social networks. Our scheme determines the potential spread of a shared data object and informs the user of the risk of information leakage associated with different sharing decisions she can make in a social

network. In this section, we review previous work related to the key ideas used in α-myCommunity scheme.

Recently, Relationship-Based Access Control (ReBAC) was proposed in [6, 7] to express the access control policies in terms of interpersonal relationships between users. ReBAC captures the contextual nature of relationships in OSNs. PriMa [112] is another recently proposed privacy protection mechanisms for OSNs. The policy construction algorithm considers factors such as average privacy preference of similar and related users, popularity of the owner, and etc. These factors are then combined to generate access control rules for profile items. While PriMa is a scheme by which access control policies are automatically constructed for users, our proposed methods try to find the best sharing sets based on users access control policies [113, 114, 115]. Similarly, [116] presented an adaptive modularity-based method for identifying and tracing community structure of dynamic OSNs. Their algorithm can quickly and efficiently update the network structure, and trace the evolving of its communities over time.

Authors of [8] studied access control policies of data co-owned by multiple parties in OSNs setting, in a way that each co-owner may separately specify her own privacy preference for the shared data. A voting algorithm, using game theory, was adopted to enable the collective enforcement of shared data. A complete survey of several privacy preserving techniques and access control model for OSNs is provided by [117, 118]. Authors of [119] addressed the problem of inferences of private user attributes from public profile attributes, links and group memberships in OSNs, whereas the effect of social relations on sensitive attribute inference was investigated in [120].

Group-centric models have been proposed for secure information sharing in [121], and

[122]. The authors focused on authorizations involving the temporal aspect of group membership. In addition, they proposed super distribution (SD) and micro distribution (MD) as solutions for secure information sharing. In SD algorithm, a single key is shared amongst all group users, while in MD algorithm the objects are individually encrypted for each group user. The limitations of SD and MD are addressed in [123] where a new hybrid approach is also proposed as a solution.

The usage control based security framework (UCON) for collaborative applications is proposed in [92]. Authors tried to develop a unified framework to encompass traditional access control, trust management, and digital rights management. In their framework, policies can be specified as attributes for subjects and objects, system attributes (e.g. conditional constraints), and user actions (e.g. obligations). They defined not only mutable usage attributes of subjects and objects, but also persistent attributes (e.g. roles and group memberships) as general attributes. In addition, UCON uses conditions to support context-based authorizations in improvised collaborations [93]. The proposed architecture in [93] uses a hybrid approach of attributes acquisitions and event-based updates in order to support attribute mutability and decision continuity.

Effectively, Dynamic Coalition Problem (DCP) has been proposed as a new context to be considered in information sharing challenges [124, 125, 126]. This scheme is about forming a coalition in order to solve problems through quick information sharing. Utilizing trust metrics for imposing access restrictions is similar to the multi-level security that is proposed in [3] in order to preserve the trustworthiness of the users' data in OSNs. Furthermore, [127, 4, 128, 129] introduced a new discretionary access control model and a related enforcement mechanism for the controlled sharing of information in online social networks. The new scheme adopts a rule-based approach for specifying access policies

on information owned by network users. In their scheme, authorized users are denoted in terms of the type, depth, and trust level of the relationships existing between nodes in the network. The authors of [130, 131, 132, 133] attempted to advance beyond the access control mechanisms found in commercial social network systems. They developed a decentralized social network system with relationship types, trust metrics and degree-of-separation policies. Furthermore, Carminati et al. [17] tried to find the upper-bound and lower-bound for the risk of unauthorized propagation of information in OSN. They used a probability-based approach to modeling the likelihood that information propagates from one social network user to users who are not authorized to access it. However, in our approach, we closely approximated the risk of information leakage by using randomized algorithms. We also provided algorithms in order to change the risk of information leakage to the desired value. In [134], authors presented a new OSN called Persona where users state who may have access to their information. This OSN uses attribute-based encryption to hide users' data and allows users to apply their own policies over who may view their data.

The metadata calculus for used secure information sharing is presented in [135]. This scheme models the metadata for security as a vector to support different operations. It is shown that, without incurring exponential metadata expansion, it is impossible to achieve strong homomorphism. In addition, Authors of [136, 137] presented new techniques for controlling the information flow in decentralized systems. In [137], authors introduced a new model for controlling information flow in systems with mutual distrust and decentralized authority . Their model allows users to share information with distrusted code and to control how that code propagates the shared information to other users. [136] also presented a new trust management paradigm for securing both intra- and inter-organizational

information flows against the threat of information disclosure. They proposed an approach for assessing the risks in terms of trustworthiness and improving risk estimations by involving estimates of trust. Their approach also provides a mechanism for handling risk transfer across organizations and forcing rational entities to be honest. Furthermore, Lockr is proposed in [2] as an access control system in order to improve the privacy of OSNs users. Lockr separates the management of social information from all other functionality of OSNs by letting users exchange digitally signed attestations. This feature facilitates the integration of Lockr's access control with various centralized or decentralized online applications. Finally, the access control paradigm behind the privacy preservation mechanism of Facebook is discussed in [5]. The authors show how their model can be represented to specify access control policies through different social factors. In addition, the authors proposed a privacy-enhanced visualization tool, which approximates the extended neighborhood of a user in such a way that policy assessment can still be conducted in a meaningful manner, while the privacy of other users is preserved [138, 16, 14, 139].

6.2 Integrity Management

The incredible success of crowdsourcing systems has attracted a lot of researchers. So it is not surprising that numerous publications about these types of systems have appeared in the last few years. There is a wide and interdisciplinary array of issues being discussed, such as visualization tools [140, 141, 142], motivations for participation [143], the effects of coordination and collaboration [144], vandalism analysis and detection [145, 146, 147, 148], reputation systems [149, 89, 150], quality assurance and automatic quality measurement [149, 90, 89, 151, 150, 152, 153]. Relating to integrity management, there are two divisions of research. The first group investigates the trustworthiness of article whereas the

second one is involved in the assessment of the integrity of the article as a whole.

The first group of study focuses on the computing the trustworthiness of text within the articles. The methods in this category offer a means for predicting the accuracy of some facts of an article. Cross [152] introduced an approach that calculates the trustworthiness throughout the life span of the text in the article and marks this by using different colors. Adler and de Alfaro calculated the reputation of the authors by using the survival time of their edits as the first step [149]. Then they analyzed exactly which text of an article was inserted by precisely which author. Finally, based on the reputation score of the respective authors, Adler and de Alfaro are able to compute the trustworthiness of each word [154]. Analogous to Cross they illustrate the trustworthiness by using color-coding.

Our work is similar to the above in the sense that we found the highly trustable authors as well. We differ with those techniques in certain aspects; for instance, we specify the trustable users by looking at the result of their contributions through a voting process. The voting process evaluates the contributions based on the opinion of other users whereas this is not the case in Adler and Alfaro mechanisms.

The second group of study focuses on assessing the integrity of an article as a whole. A first work in this category was published in [88] introduced a correlation of the integrity of an article with the number of editors as well as the number of article revisions. [89] defines three models for ranking articles according to their integrity level. The models are based on the length of the article, the total number of revisions and the reputation of the authors, which is measured by the total number of their previous edits. In [150], the authors proposed to compute the integrity of a particular article version with a Bayesian network from the reputation of its authors, the number of words the authors have changed and the quality score of the previous version. Furthermore, on the basis of a statistical

comparison of a sample of Featured and Non-Featured Articles in the English Wikipedia, authors of [151] constructed seven complex metrics and used a combination of them for integrity measurement. However, their work is completely based on the Wikipedia model as one of the widely used crowdsourcing systems.

In [153], the authors derived ten metrics from research related to collaboration to predict integrity of a particular article. Similarly, [90] investigated over 100 partial simple metrics, for instance the number of words, characters, sentences, internal and external links in order to measure the integrity of an article in crowdsourcing systems. In this work, authors evaluated the metrics by using them for classifications between Featured and Non-Featured Articles in Wikipedia. Also, they demonstrated, with an accuracy of classification of 97%, that the number of words is the best current metric for distinguishing between Featured and Non-Featured Articles in open access crowdsourcing systems such as Wikipedia.

6.3 Analysis of Online Social Systems

Online social systems have been widely studied under many different topics. In this section, we discuss the previous works on the two main categories of online social systems: crowdsourcing and social networks. First, we focus on characterizing the online question answering sites as one of the widely used crowdsourcing systems in Section 6.3.1. Second, we study the previous works on the large scale measurement of online social networks in Section 6.3.2.

6.3.1 Analysis of crowdsourcing systems

Many researchers have focused on large scale measurement studies of Question Answering (Q/A) systems [26, 155, 156]. The most important factor in the success of Q/A systems is providing the best answer for a given question. There are various approaches to better achieve this goal. The existing content in Q/A systems can be reused to provide appropriate answers for a given question based on effective retrieval of relevant questions and answers [157, 158, 159, 160]. In [25], the authors attempted to reduce the rate of unanswered questions in Yahoo! Answers by reusing the answers of past resolved questions. By leveraging concepts and methods from query performance prediction and natural language processing, their method approximated the probability that new questions could be satisfactorily answered by one of the best answers from the past.

Similarly, authors of [161, 162, 163] tried to tackle the same problem by pushing open questions to potential answerers. To find experts on the topic of each question, [164] formulated a graph structure of Q/A systems in order to find authoritative users in topical categories. Authors of [165] focused on automatically differentiating between authoritative and non-authoritative users by modeling the user authority scores for each topic. In general, [166, 167] proposed efficient ways to find experts in online forums which can be used in the context of Q/A systems. [168] developed a probabilistic model that incorporates both the topic-level and term level information in order to recommend new questions to relevant answerers. Authors of [27, 169, 170] tried to model user interest with different approaches in order to reduce the response time to new questions in Q/A systems. Additionally, [171] used machine learning techniques to automatically classify questions as conversational or informational in order to archive and access the informational questions for future use.

Another set of effort proposes different ways to find the best answer among a collection of answers for a given question. [172] utilized translation models to find the most relevant answers and evaluate their approaches on a small set of questions with known answers ahead of time. Authors of [173] combined translation and similarity features in order to rank answers based on their relevance and then choose the best answer. Similarly, [174] used a supervised learning-to-rank algorithm to support relevant answers to the given question based on its properties. This is another type of approach proposed in order to find past questions which are related to the given question. In [175], authors combined a translation model for question similarity and a language model for answer similarity as part of their retrieval model for similar questions. Retrieving questions with similar topics and focusing on those that pertain to the target question was proposed by [176]. [177] proposed to identify similar questions by assessing the similarity between their syntactic parse trees.

Q/A systems can be made more efficient by encouraging users to participate more. Many researchers have focused on better understanding of user behavior in Q/A systems. The content properties and the user interaction patterns across different Yahoo! Answers categories has been studied in [28]. They found that lower entropy correlates with receiving higher answer ratings, for categories where factual expertise is a sought factor. In addition, they combined both user attributes and answer characteristics to predict whether a particular answer will be chosen as the best by the questioner for each category. Similarly, authors of [178] analyzed user activity levels, interests and reputation in Yahoo! Answers.

In [179], authors focused on finding the tradeoff between maximizing the information accuracy while minimizing the waiting time for users in Q/A systems. In a popular Korean Q/A system named Naver, the motivation of top answerers has been investigated by [180].

[181] explored the patterns of user contributions in online knowledge sharing social networks. In [29], the authors suggested to consider whether the potential answerer is likely to accept and answer the recommended questions in a timely manner. They studied answerer behavior in Yahoo! Answer. More specifically, they analyzed how answerers tend to choose the questions and when users tend to answer questions in a Q/A system.

6.3.2 Analysis of online social networks

There have been a lot of works focused on large scale measurement studies of online social networks. Of these studies, few have focused on characterizing user behavior in online social networks such as [182]. This paper focused on understanding how users behave when they connect to social networks. The authors classified the users' social interactions into two different groups: publicly visible activities (e.g. comments) and silent activities (e.g. viewing photos of a friend and browsing a profile page). In contrast, our study mainly focuses on the user behavior in crowdsourcing question answering systems. In addition, [183] presents an analysis on the online social network formed by users on Twitter. Authors analyzed the user behavior and studied the geographical spread of the Twitter usage. Further, [184] conducted studies on the user motivations for any contribution on the Facebook social network. Their results indicated that new users share more contents if their friends have more contributions.

Many studies such [104, 185, 186, 187] have analyzed the structure and evolution of online social networks. These studies have confirmed the power-law, small world and scale free properties of online social networks. While these studies investigated the topological structure of online social networks, another direction of research focused on user activity in these networks [188] examining the activities from the guest book logs of the Cyworld

online social network. In their study, the activity network was constructed based on the comments posted by users in their guest books. Authors discovered that users tended to have mutual interactions similar to the ones in social networks.

Similarly, [189] studied activity networks from Facebook. Their results showed that the structural properties of activity networks differed from social networks. The evolution of the activity network of Facebook over time is presented by [111]. Their results showed that the strength of the ties exhibits a decreasing trend of activity as the social network link ages. In addition, the graph-theoretic properties of the activity network (e.g., average node degree, average clustering coefficient, and average path length) remain stable, while the links of the activity network change rapidly over time. The analysis of an activity network based on the user interaction in a large instant messaging network is presented in [190]. The authors examined the structure of the activity network and its dependency on the user demographics. They found out that similar users interact more often.

The geographical location of LiveJournal users is analyzed by [191]. Their results showed that there is a strong correlation between friendship and geographic proximity. In addition, the measurement study of the Flickr network is presented in [192]. Their results indicated that the majority of user interactions are done by a small fraction of users. Similarly, the measurement analysis of Twitter users is done by [193]. They observed that Twitter users have a small number of friends compared to the number of followers they declare.

A number of papers have been published on various analysis of Facebook social network datasets. Analysis of the temporal and social access patterns in Facebook is presented in [116]. For their analysis, the authors focused on the messages exchanged by users in Facebook. They examined the message header and found out the periodic patterns in terms

of messages exchanged on the network. In addition, the authors of [194] have studied application usage workloads in Facebook and the popularity of applications. They discovered that although the total number of application installations increases with time among Facebook users, the average user activity decreases and Facebook users with more applications installed are more likely to install new applications. Similarly, the application characteristics analysis of Facebook network is done in [195]. The authors developed and launched their own applications with over 8 million subscribers. They figured out that user response times for Facebook applications are independent of the source-destination user locality. Finally, authors of [196] conducted survey interviews to analyze the web browsing patterns of various users from 4 different nationalities. They examined the ethnographical differences in the usage of online social networks.

In summary, compared to the above studies that have extensively analyzed the structural properties of online social networks and the user behavior in them, we focused on characterizing the user behavior in a crowdsourcing question answering service. For the first time, we considered the effect of tagging, user reputation, and their feedbacks on users' behavior and their response time.

7

Conclusion

7.1 Summary of Contributions

This thesis studied ways of leveraging social factors characterized by relations in online systems to develop novel solutions for important information confidentiality and integrity problems on the Internet. Controlling the confidentiality of information within online social networks is the first problem studied in the thesis. The second problem dealt with is the integrity of articles in large-scale crowdsourcing systems and finally characterizing the user behavior in crowdsourcing question answering systems is the third problem focused in the thesis.

Chapter 3 was based on the premise that information sharing in an OSN cannot be completely controlled by a single user. For instance, Alice can determine the set of friends with whom a piece of data should be shared in the network. However, Alice does not have complete control over her friends' actions. When sharing, she trusts that her friends would adhere to the accepted norms with regard to information usage. Some of her friends may adhere to the informally accepted information usage policies while others may not adhere to them.

We proposed a Monte Carlo based algorithm to compute the sharing subgraph which

shows the dispersion of the information on the network. We refer to such subgraphs as α-myCommunity, where α specifies the certainty that all members of the subgraph will know about the information. We used datasets from Flickr and Facebook to compute α-myCommunities for various α values and blocking lists for different adversaries for each individual users. From the experiments, we noticed that certain α-myCommunities are more robust with regard to information leakage. In addition, we observed that considering the history of interactions between users results in better estimation of α-myCommunity and blocking lists.

Further, Chapter 3 was concerned with developing the notion of α-myCommunity and applying it to datasets extracted from actual OSN activities. The major focus of this study has been to validate the notion of α-myCommunity by illustrating the *community-centric* information sharing patterns that actually take place within OSNs. Also, we developed schemes that will allow a user to shape the α-myCommunity to fit her intention. For instance, if Alice wants to minimize the possibility of information leaking to her nemesis, with our algorithm, she can find out how she should shape her sharing decisions.

In summary, the main contributions of Chapter 3 are the following:

- The larger the value of α the smaller the size of α-myCommunity. This indicates that interactions in OSNs take place in tightly knit groups. These groups, however, may not be cleanly defined by notions such as friends, or friends-of-friends.

- The best α value for Facebook users is about 0.7. This means by choosing 0.7-myCommunities Facebook users can form communities that are robust in terms of information sharing. The information discharged into such communities are more likely to be contained within them. Similarly, for the Flickr network, we observed an α value of 0.8. Instead of asking the user for an α value the algorithm selects the

value yielding a robust configuration.

- Blocking an adversary directly may not prevent flow of information to him in social networks. This indicates that users can receive information not only directly from the owner, but also indirectly from common friends. However, the indirect ways may not be cleanly defined by existing notions in OSNs.

- Because interactions are time dependent, we evaluated different ways of computing α-myCommunities. By including historical information in the computation process, we were able to compute relatively stable α-myCommunities.

In Chapter 4, we focused on the problem of information integrity in online crowdsourcing systems. We started the study by examining the challenges and requirements for preserving information integrity in crowdsourcing systems. We observed that integrity remains a hard problem primarily due to the openness feature of crowdsourcing. By examing twelve major crowdsourcing systems, we noted that all crowdsourcing systems except Wikipedia include content ownership in their design.

In Wikipedia, we observed that the main difference between low and featured articles is the number of contributions. High quality articles have larger number of contributions compared to low quality ones. We noticed that most of the contributions for featured articles come from a small number of editors who we refer to as the top editors. On average, the number of top editors for a featured article is around 32. The top editors have similar interests in terms of their editing activities. We observed that the top editors control the evolution of high quality articles by their contributions and actively revert back other's contributions. It means that although Wikipedia is an open platform top editors own the featured articles.

In this chapter, we presented the design for Social Integrity Management (SIM), a novel scheme for preserving integrity of articles in online crowdsourcing systems. The design of SIM is motivated by our findings from Wikipedia's analysis and our observations of user behaviour in another crowdsourcing system in Chapter 5. The design of SIM uses ownership as a key factor to control integrity of documents. We addressed the ownership bottleneck by including co-ownership and allowing limited number of versions. In addition, SIM uses user activities to assign proper trust levels between users. We evaluated the new scheme using extensive simulations on actual datasets from Facebook.

Next, we studied the effects of social network topological properties on the number of versions per articles. We considered the influence of structural properties such as node degree on the required number of versions per article. The larger the average node degree, the smaller the number of versions per article we need to reach article' maturity with less number of iterations. We observed that fewer versions are required to reach the best editors in a social neighbourhood when the density of the social graph increases (i.e., the larger average node degree).

The major contributions of Chapter 4 are as follows:

- Analyzed integrity management in Wikipedia and examined the role of contributors in an article becoming a featured one.
- Proposed a social integrity management scheme for preserving integrity in online crowdsourcing systems based on incorporating co-ownership and multiple versions.
- Analyzed the effects of social network structure on the features of the proposed scheme.

In Chapter 5, we analyzed the characteristics of crowdsourcing question answering

systems using datasets from 10 popular Stack Exchange sites. We chose Stack Exchange collections because of their popularity and their effectiveness in providing solutions to user issues. One of our focus was analyzing user behavior in the Stack Exchange sites. We conducted a survey among users of Stack Exchange sites to gather actual feedbacks. Our survey showed that question topic is the most important factor for users in choosing a question to answer. Tags, quality of the question, and if the question is on the first page are other major factors. In addition, we observed that users prefer to answer questions without many replies to obtain more reputation points by having higher chance of their answer becomes the accepted or most popular one.

Accordingly, we focused on examining user activity in the Stack Exchange sites. Our analysis in this regard demonstrates that a minority of users with relatively high reputation provide more than 85% of the answers. We also found that in these sites, information flows from highly-reputed users to users with low reputation values. We observed that more than 90% of the edits were done by small portions of users (around 12%), and that the updates occur in such a way that posters with low reputation are corrected by members with higher reputation. We also noticed that while routine questions tagged by the popular tags were answered by the majority of the users (high or low reputed users), the questions on not so popular topics are mostly answered by users with medium reputation.

Additionally, we noted that the Programmers site has a larger diversity of answers than other sites included in this study. That is, each question on average received more than 6 answers compared to 2 answers for other sites. Also, in more than 85% of the cases, the most popular answer was the accepted one. However, in sites with relatively diverse answers like Programmers, the most popular answer was accepted only 75% of the time.

Another focus of our study was to characterize the patterns of activities for high and

low active or reputed users. We observed that low active users typically take more than a month to initiate their activities by asking their first and only question. In contrast, highly active users start their activities immediately after joining the system. In addition, posts from highly-active users gain three times more responses compared to the posts made by low-active users, and highly-active users get proper solutions in those responses. This finding was confirmed by our survey where majority of low active users provided "Never" for getting proper answers to their questions. Further, we noticed that the chance for an answer to become the most popular or the accepted one increases with the reputation of the answerer. The probability that an answer from a highly reputed user becomes the most popular one is more than 0.55.

Yet another focus of our study was to observe the crowd behavior within the answerers. That is, we wanted to examine whether the answerers prefer to follow each other while answering questions. From our analysis of the datasets, we observed such crowd behavior although Stack Exchange does not have any explicit mechanism such as friendships or follows to organize the signalling mechanisms to initiate such activities. In our analysis, we formed the collaborative network based on the number of times a pair of users answered the same questions together. We noticed that users collaborate up to 84% in answering various questions together. While the co-answering patterns that form the collaborative network changed rapidly overtime, the average network properties remained relatively stable. In addition, we showed that the distribution of the questioners, answerers and node degrees fit into a power-law network with small deviations.

From our analysis we were able to make the following key observations:

- Users with high and low reputations are interested in answering questions in popular topics (i.e., questions tagged with popular keywords) whereas questions on not so

popular topics are mostly answered by medium reputed users.

- Answerers form "crowds" while answering the questions. In other words, users tend to co-answer questions despite the lack of an explicit friendship facility in the Stack Exchange sites.

- Although the co-answering patterns change over time some graph theoretic measures of the collaborative networks were observed to be stable. In addition, we conducted a survey among actual users of the Stack Exchange sites.

7.2 Future Extensions

In this section, we discuss some future research directions for the research presented in this thesis.

In Chapter 3, we considered a single context. That is all users were concerned about a single topic. Future studies need to consider multiple contexts. With a diverse array of topics, a conversation between two users could be about many topics and modeling how much a particular conversation contributes towards the propagation of an information object becomes a harder problem.

In Chapter 4, we analyzed Wikipedia as one of the widely used owner-free crowdsourcing systems. We examined on the evolution of articles in an owner-free environment where every reader is a potential contributor.

One future work is to extend our analysis to compare owner-free systems such as Wikipedia with owner-centric wiki-style pages. Stack Exchange sites have community wikis which are built by pseudonymous participants. The user reputation gain or loss is associated with pseudonym handle they use within the sites. The capabilities a user has is

directly determined by her reputation. It is interesting to compare the evolution of community wiki pages in Stack Exchange sites with the evolution of Wikipedia pages. Such comparison will shed light on the importance of accountability in crowdsourcing systems.

In Chapter 5, we discovered that the efficiency of Stack Exchange sites decreases when the system scales up. The average response time becomes larger and the number of answers for each question decreases because there is an increase in the number of questions in these sites. The delay in receiving an answer in any type of question answering service impacts the quality of service enjoyed by the users in a significant manner. The average response time in Stack Exchange sites is in the order of hours. Therefore, further enhancements should be made to better route the questions to the eventual answerers. In particular, how the response times can be contained as the network grows is a major issue. One observation we made in our research is the importance of tags in question routing. Questions using tags with medium popularity tend to get higher number of answers while questions with popular or rare tags gain less attention from answerers. Therefore, there is a need for further study on how tags, their quality and the way questioners use them, can be utilized to have more efficient question-to-answerer matching service.

Bibliography

[1] G. Bruns, P. W. L. Fong, I. Siahaa, and M. Huth, "Relationship-based access control: Its expression and enforcement through hybrid logic," in *In Proceedings of the 2nd ACM Conference on Data and Application Security and Privacy (CODASPY'2012)*, San Antonio, TX, USA, February 7-9, 2012, pp. 51–60.

[2] A. Tootoonchian, S. Saroiu, A. Wolman, and Y. Ganjali, "Lockr: Better privacy for social networks," in *In Proceedings of the 5th CONEXT*, 2009.

[3] B. Ali, W. Villegas, and M. Maheswaran, "A trust based approach for protecting user data in social networks," in *Proceedings of the conference of the center for advanced studies on Collaborative research*, Richmond Hill, Ontario, Canada, 2007.

[4] B. Carminati, E. Ferrari, and A. Perego, "Enforcing access control in web-based social networks," *ACM Transactions on Information and System Security (TISSEC)*, vol. 13, pp. 6:1–6:38, November 2009.

[5] P. W. L. Fong, M. Anwar, and Z. Zhao, "A privacy preservation model for facebook-style social network systems," in *Proceedings of the 14th European conference on Research in computer security*, Saint-Malo, France, 2009.

[6] P. W. L. Fong, "Relationship-based access control: Protection model and policy language," in *Proceedings of the First ACM Conference on Data and Application Security and Privacy (CODASPY)*, San Antonio, Taxas, USA, February 2011, pp. 191–202.

[7] P. W. L. Fong and I. Siahaa, "Relationship-based access control policies and their policy languages," in *Proceedings of the 16th ACM Symposium on Access Control Models and Technologies (SACMAT'11)*, Innsbruck, Austria, June 2011, pp. 51–60.

[8] A. C. Squicciarini, M. Shehab, and J. Wede, "Privacy policies for shared content in social network sites," *The VLDB Journal*, vol. 19, pp. 777–796, December 2010.

[9] B. Carminati, E. Ferrari, and J. Girardi, "Trust and share: Trusted information sharing in online social networks," in *Demo paper at 28th International Conference on Data Engineering (ICDE 2012)*, April 2012.

[10] C. Akcora, B. Carminati, and E. Ferrari, "Risks of friendships on social networks," in *Proceedings of the IEEE International Conference on Data Mining (ICDM 2012)*, April 2012.

[11] J. Domingo-Ferrer, A. Viejo, F. Sebé, and ı. González-Nicolás, "Privacy homomorphisms for social networks with private relationships," *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 52, pp. 3007–3016, October 2008.

[12] N. Elahi, M. M. R. Chowdhury, and J. Noll, "Semantic access control in web based communities," in *Proceedings of the Third International Multi-Conference on Computing in the Global Information Technology (ICCGI)*. Washington, DC, USA:

IEEE Computer Society, 2008, pp. 131–136.

[13] S. R. Kruk, S. Grzonkowski, A. Gzella, T. Woroniecki, and H. Choi, "D-foaf: Distributed identity management with access rights delegation," in *Proceesing of the Asian Semantic Web Conference*, ser. Lecture Notes in Computer Science, vol. 4185. Springer, 2006, pp. 140–154.

[14] M. Anwar and P. W. L. Fong., "A visualization tool for evaluating access control policies in facebook-style social network systems," in *Proceedings of the 27th ACM Symposium on Applied Computing (SAC'12), Security Track*, Riva del Garda, Trento, Italy, March 2012.

[15] C. Akcora, B. Carminati, and E. Ferrari, "Privacy in social networks: How risky is your social graph?" in *Proceedings of 28th International Conference on Data Engineering (ICDE 2012)*, April 2012.

[16] M. Anwar, P. W. L. Fong, X.-D. Yang, and H. Hamilton, "Visualizing privacy implications of access control policies in social network systems," in *Proceedings of the 4th International Workshop on Data Privacy Management (DPM'09)*, Saint Malo, France, September 2009, pp. 106–120.

[17] B. Carminati, E. Ferrari, S. Morasca, and D. Taibi, "A probability-based approach to modeling the risk of unauthorized propagation of information in on-line social networks," in *Proceedings of the first ACM conference on Data and application security and privacy*, ser. CODASPY '11. San Antonio, TX, USA: ACM, 2011, pp. 51–62.

[18] A. Steffen, "The linux integrity measurement architecture and tpm-based network endpoint assessment," in *technical report, HSR University of Applied Sciences Rapperswil*, 2012.

[19] Y. Suzuki and M. Yoshikawa, "Mutual evaluation of editors and texts for assessing quality of wikipedia articles," in *Proceedings of the 8th International Symposium on Wikis and Open Collaboration*, August 2012.

[20] Citizendium, http://en.citizendium.org/.

[21] Scholarpedia, http://www.scholarpedia.org/.

[22] A. West and I. Lee, "Towards content-driven reputation for collaborative code repositories," in *Proceedings of the 8th International Symposium on Wikis and Open Collaboration*, August 2012.

[23] Reliability of Wikipedia, http://en.wikipedia.org/wiki/ Reliability_of_Wikipedia.

[24] D. Zhang, K. Prior, and M. Levene, "How long do wikipedia editors keep active?" in *Proceedings of the 8th International Symposium on Wikis and Open Collaboration*, August 2012.

[25] A. Shtok, G. Dror, Y. Maarek, and I. Szpektor, "Idan. learning from the past: answering new questions with past answers," in *In Proceedings of the 21st international conference on World Wide Web (WWW '12)*, Lyon, France, 2012.

[26] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne, "Finding high-quality content in social media," in *In Proceedings of the international conference on Web search and web data mining (WSDM '08)*, Palo Alto, California, USA, 2008.

[27] M. Liu, Y. Liu, and Q. Yang, "Predicting best answerers for new questions in community question answering," in *In Proceedings of the 11th international conference on Web-age information management (WAIM'10)*, Jiuzhaigou, China, 2010.

[28] L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman, "Knowledge sharing and yahoo answers: everyone knows something," in *In Proceedings of the 17th international conference on World Wide Web (WWW '08)*, Beijing, China, 2008.

[29] Q. Liu and E. Agichtein, "Modeling answerer behavior in collaborative question answering systems," in *In Proceedings of the 33rd European conference on Advances in information retrieval (ECIR'11)*, Dublin, Ireland, 2011.

[30] B. Wellman and S. D. Berkowitz, *Social structures: a network approach*. New York, NY: Cambridge University Press, 1988.

[31] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994.

[32] M. Morris, "Epidemiology and social networks: Modeling structured diffusion," *Sociological Methods Research*, vol. 22, no. 1, pp. 99–126, August 1993.

[33] L. F. Berkman and S. L. Syme, "Social networks, host resistance, and mortality: a nine-year follow-up study of alameda county residents," *American Journal of Epidemiology*, vol. 109, no. 2, p. 186204, February 1979.

[34] M. S. Granovetter, "The strength of weak ties," *American Journal of Sociology*, vol. 78, no. 6, p. 13601380, May 1973.

[35] J. Travers and S. Milgram, "An experimental study of the small world problem," *Journal of Sociometry*, vol. 32, p. 425443, 1969.

[36] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, jun 1998.

[37] J. D. Montgomery, "Social networks and labor-market outcomes: Toward an economic analysis," *The American Economic Review*, vol. 81, no. 5, p. 14081418, December 1991.

[38] D. M. Boyd and N. B. Ellison, "Social network sites: Definition, history, and scholarship," *Journal of Computer-Mediated Communication*, vol. 13, no. 1, p. 210230, 2008.

[39] Online social networks - research report, http://www.communities.gov.uk/publications/communities/onlinesocialnetworks, October 2008.

[40] After 10 Years of Blogs, the Future's Brighter Than Ever, http://www.wired.com/entertainment/theweb/news/2007/12/blog-anniversary.

[41] Alexa Top 500 Global Sites, http://www.alexa.com/topsites.

[42] For Social Networks, There's Still Room to Play, http://www.nielsen-online.com/blog/2008/10/22/for-social-networks-there's-still-room-to-play/.

[43] R. Gross and A. Acquisti, "Information revelation and privacy in online social networks," in *Proceedings of the 2005 ACM workshop on Privacy in the electronic society*. Alexandria, VA, USA: ACM, 2005, pp. 71–80.

[44] G. Hobgen, "Security issues and recommendations for online social networks," EINSA position paper No.1, October 2007.

[45] M. Bishop, *Computer Security: Art and Science*. Addison-Wesley, 2002.

[46] B. W. Lampson, "Protection," in *Proceedings 5th Princeton Symp. Information Science and Systems*, March 1971, pp. 437–443. Reprinted in Operating Systems Rev. 8, 1 (Jan. 1974), 18–24.

[47] D. Bell and L. Lapadula, "Secure computer systems: Mathematical foundations," *The MITRE Corporation, Bedford, Massachusetts, Technical Report, MTR-2547, 1973*.

[48] ——, "Secure computer system: Unified exposition and multics interpretation," *Deputy for Command and Management Systems, United State Air Force, March 1976*.

[49] D. Ferraiolo and R. Kuhn, "Role-based access controls," *In Proceedings of the 15th NIST-NCSC National Computer Security Conference, pp. 554-563, Oct. 1992*.

[50] M. Benantar, *Access Control Systems: Security, Identity Management and Trust Models*. New York, NY: Springer, 2006.

[51] R. S. Sandhu and P. Samarati, "Access control: Principles and practice," *IEEE Communications Magazine*, vol. 32, no. 9, pp. 40–48, 1994.

[52] R. S. Sandhu, "Lattice-based access control models," *IEEE Computer*, vol. 26, no. 11, pp. 9–19, Nov. 1993.

[53] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-based access control models," *IEEE Computer*, vol. 29, no. 2, pp. 38–47, 1996.

[54] A. Tootoonchian, K. K. Gollu, S. Saroiu, Y. Ganjali, and A. Wolman, "Lockr: social access control for web 2.0," in *Proceedings of the first workshop on Online social networks*. Seattle, WA, USA: ACM, 2008, pp. 43–48.

[55] B. Carminati, E. Ferrari, and A. Perego, "Rule-based access control for social networks," in *OTM Workshops (2)*, 2006, pp. 1734–1744.

[56] W. Villegas, B. Ali, and M. Maheswaran, "An access control scheme for protecting personal data," in *PST '08: Proceedings of the 2008 Sixth Annual Conference on Privacy, Security and Trust*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 24–35.

[57] M. Hart, R. Johnson, and A. Stent, "More content – less control: Access control in the web 2.0," in *Proceedings of the Web 2.0 Security and Privacy Workshop*, 2007.

[58] R. Steinmetz and K. Wehrle, "Peer-to-peer systems and applications," *Springer, Lecture Notes in Computer Science*, no. 3485, 2005.

[59] A. Doan, R. Ramakrishnan, and A. Y. Halevy, "Crowdsourcing systems on the world-wide web," *Communications of the ACM*, vol. 54, no. 4, pp. 86–96, April 2011.

[60] Z. G. Ives, N. Khandelwal, A. Kapur, and M. Cakir, "Orchestra: Rapid, collaborative sharing of dynamic data," *In Proceedings of Conference on Innovative Data Systems Research (CIDR), Asilomar, CA*, 2005.

[61] P. DeRose, X. Chai, B. J. Gao, W. Shen, A. D. amd P. Bohannon, and X. Zhu, "Building community wikipedias: A machine-human partnership approach," in *Proceedings of the 24th IEEE International Conference on Data Engineering (ICDE)*, Cancn, Mxico, April 2008.

[62] D. S. Weld, F. Wu, E. Adar, S. Amershi, J. Fogarty, R. Hoffmann, K. Patel, and M. Skinner, "Intelligence in wikipedia," in *In Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (AAAI)*, Chicago, Illinois, USA, July 2008.

[63] D. Tapscott and A. D. Williams, *Wikinomics*. Portfolio, 2006.

[64] J. Surowiecki, *The Wisdom of Crowds*. Anchor Books, 2005.

[65] H. Rheingold, *Smart Mobs*. Perseus Publishing, 2003.

[66] M. Richardson and P. Domingos, "Building large knowledge bases by mass collaboration," in *In Proceedings of the International Conference on Knowledge Capture (K-CAP)*, 2003.

[67] L. von Ahn and L. Dabbish, "Labeling images with a computer game," in *In Proceedings of the International Conference on Human Factors in Computing Systems (CHI)*, Vienna, Austria, 2004.

[68] M. Olson, "The amateur search," *SIGMOD Record*, vol. 37, no. 2, p. 2124, 2008.

[69] Http://en.wikipedia.org/wiki/Special:Statistics.

[70] The sheep market. http://www.thesheepmarket.com/.

[71] A. M. Koblin, "The sheep market," in *In Proceeding of the seventh ACM conference on Creativity and cognition*, New York, NY, USA, 2009, p. 451452.

[72] D. C. Brabham, "Crowdsourcing as a model for problem solving: An introduction and cases," *Convergence: The International Journal of Research into New Media Technologies*, vol. 14, no. 1, p. 7590, 2008.

[73] J. Leimeister, M. Huber, U. Bretschneider, and H. Krcmar, "Leveraging crowdsourcing: Activation-supporting components for it-based ideas competition," *Journal of Management Information Systems*, vol. 26, p. 197224, July 2009.

[74] L. von Ahn, "Games with a purpose," *IEEE Computer*, vol. 39, no. 6, p. 9294, June 2006.

[75] L. von Ahn, R. Liu, and M. Blum, "Peekaboom: A game for locating objects in images," *In ACM SIGCHI Conference on Human Factors in Computing Systems*, 2006.

[76] E. Law and L. von Ahn, "Input-agreement: A new mechanism for data collection using human computation games," in *In Proceedings of the International Conference on Human Factors in Computing Systems (CHI)*, 2009.

[77] M. Mandel and D. Ellis, "A web-based game for collecting music metadata," in *In 8th International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, 2007.

[78] D. Turnbull, R. Liu, L. Barrington, and G. Lanckriet, "A gamebased approach for collecting semantic annotations of music," in *In 8th International Conference on Music Information Retrieval (ISMIR)*, Vienna, Austria, 2007.

[79] Amazon mechanical turk. https://www.mturk.com/.

[80] R. McCann, A. Doan, V. Varadarajan, and A. Kramnik, "Building data integration systems: A mass collaboration approach," in *In Proceedings of the International Workshop on Web and Databases (WebDB)*, San Diego, California, June 2003.

[81] R. McCann, W. Shen, and A. Doan, "Matching schemas in online communities: A web 2.0 approach," in *In Proceedings of the 24th IEEE International Conference on Data Engineering (ICDE)*, Cancn, Mxico, April 2008.

[82] L. von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum, "Recaptcha: Human-based character recognition via web security measures," *Science*, vol. 321, no. 5895, 2008.

[83] G. Kasneci, M. Ramanath, M. Suchanek, , and G. Weiku, "The yago-naga approach to knowledge discovery," *SIGMOD Record*, vol. 37, no. 4, p. 4147, 2008.

[84] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *In Proceedings of the Tenth International World Wide Web Conference (WWW 10)*, Hong Kong, China, May 2001.

[85] Http://en.wikipedia.org/wiki/Wikipedia:Featured_articles.

[86] Http://en.wikipedia.org/wiki/User:ClueBot_NG.

[87] Http://en.wikipedia.org/wiki/User:XLinkBot.

[88] A. Lih, "Wikipedia as participatory journalism:reliable sources? metrics for evaluating collaborative media as a news resource," in *In Proceedings of the 5th InternationalSymposium on Online Journalism*, Austin, USA, April 2004.

[89] E. P. Lim, B. Q. Vuong, H. W. Lauw, and A. Sun, "Measuring qualities of articles contributed by onlinecommunities," in *In Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, Hong Kong, December 2006, pp. 81–87.

[90] J. E. Blumenstock, "Size matters: Word count as a measure of quality on wikipedia," in *In Proceedings of the 17th international conference on World Wide Web*, Beijing, China, April 2008, pp. 1095–1096.

[91] D. Easley and J. Kleinberg, *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, 2010.

[92] J. Park and R. Sandhu, "The $ucon_{ABC}$ usage control model," *ACM Transactions on Information and System Security (TISSEC)*, vol. 7, pp. 128–174, February 2004.

[93] X. Zhang, M. Nakae, M. J. Covington, and R. Sandhu, "Toward a usage-based security framework for collaborative computing systems," *ACM Transactions on Information and System Security (TISSEC)*, vol. 11, pp. 3:1–3:36, February 2008.

[94] J. Ong, *UK tribunal upholds Apple's firing of retail employee for critical Facebook post*. Apple Insider, http://www.appleinsider.com/articles/11/11/01/uk_tribunal_upholds_apples_firing_of_retail_employee_for_critical_facebook_post.html, 2011.

[95] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1990.

[96] I. B. Gertsbakh and Y. Shpungin, *Models of Network Reliability: Analysis, Combinatorics, and Monte Carlo*. CRC Press, 2010.

[97] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi, "On the evolution of user interaction in facebook," in *Proceedings of the 2nd ACM SIGCOMM Workshop on Social Networks (WOSN)*, 2009.

[98] M. Cha, A. Mislove, and K. P. Gummadi, "A Measurement-driven Analysis of Information Propagation in the Flickr Social Network," in *Proceedings of the 18th International World Wide Web Conference (WWW)*, 2009.

[99] A. P. De, M. Schorlemmer, I. Csic, and S. Cranefield, "A Social-Network Defence against Whitewashing," in *aamas 2010: Proceedings of the ninth international conference on autonomous agents and multiagent systems*, Toronto, Canada, 2010, pp. 1563–1564.

[100] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman, "Sybilguard: defending against sybil attacks via social networks," in *SIGCOMM '06: Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*. New York, NY, USA: ACM, 2006, pp. 267–278.

[101] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao, "Sybillimit: A near-optimal social network defense against sybil attacks," in *SP '08: Proceedings of the 2008 IEEE Symposium on Security and Privacy*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 3–17.

[102] J. Golbeck and J. Hendler, "Inferring binary trust relationships in web-based social networks," *ACM Transactions on Internet Technology (TOIT)*, vol. 6, no. 4, p. 529, 2006.

[103] C. Binzel and D. Fehr, "Social Relationships and Trust," *Discussion Papers of DIW Berlin*, 2010.

[104] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *In Proceedings of the 5th*

ACM/USENIX Internet Measurement Conference (IMC'07), San Diego, CA, October 2007.

[105] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," in *In Proceedings of the Annual Conference of the ACM Special Interest Group on Data Communication (SIGCOMM'99)*, Cambridge, MA, August 1999.

[106] A.-L. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, pp. 509–512, 1999.

[107] R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, "Trawling the web for emerging cyber-communities," *Computer Networks*, vol. 31, pp. 1481–1493, 1999.

[108] L. A. Adamic, O. Buyukkokten, and E. Adar, "A social network caught in the web," *First Monday*, vol. 8, no. 6, 2003.

[109] A. Clauset, C. R. Shalizi, and M. E. J. Newman, "Power-law distributions in empirical data," *Journal of Society for Industrial and Applied Mathematics*, 2009.

[110] P. Mahadevan, D. Krioukov, K. Fall, , and A. Vahdat, "Systematic topology analysis and generation using degree correlations," in *In Proceedings of the Annual Conference of the ACM Special Interest Group on Data Communication (SIGCOMM'06)*, Pisa, Italy,, August 2006.

[111] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi, "On the evolution of user interaction in facebook," in *In Proceedings of the 2nd ACM SIGCOMM Workshop on Social Networks (WOSN'09)*, Barcelona, Spain, August 2009.

[112] A. Squicciarini, F. Paci, and S. Sundareswaran, "Prima: an effective privacy protection mechanism for social networks," in *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security (ASIACCS)*, Beijing, China, 2010, pp. 320–323.

[113] A. Ranjbar and M. Maheswaran, "A case for community-centric controls for information sharing on online social networks," in *Proceedings of IEEE GLOBECOM Workshop on Complex and Communication Networks (CCNet)*, Miami, Florida, USA, 2010.

[114] ——, "Blocking in community-centric information management approaches for the social web," in *Proceedings of IEEE Global Communications Conference (GLOBECOM)*, Texas, USA, 2011.

[115] ——, "Community-centric approaches for confidentiality management in online systems," in *Proceedings of 20th IEEE International Conference Computer Communication Networks (ICCCN 2011)*, Hawaii, USA, 2011.

[116] N. P. Nguyen, T. N. Dinh, Y. Xuan, and M. T. Thai, "Adaptive algorithms for detecting community structure in dynamic social networks," in *Proceedings of the 30th IEEE International Conference on Computer Communications (INFOCOM)*, Shanghai, China, 2011.

[117] B. Carminati, E. Ferrari, M. Kantarcioglu, and B. Thuraisingham., *Privacy-Aware Knowledge Discovery: Novel Applications and New Techniques, chapter Privacy protection of personal data in social networks.* Chapman and Hall/CRC Data Mining and Knowledge Discovery Series, 2010.

[118] B. Carminati, E. Ferrari, and M. Viviani, "A multi-dimensional and event-based model for trust computation in the social web," in *Proceedings of the 4th International Conference on Social Informatics (SocInfo 2012)*, December 2012.

[119] E. Zheleva and L. Getoor, "To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles," in *Proceedings of the 18th international conference on World wide web*, ser. WWW '09. Madrid, Spain: ACM, 2009, pp. 531–540.

[120] J. He, W. W. Chu, and Z. Liu, "Inferring privacy information from social networks," in *IEEE International Conference on Intelligence and Security Informatics*, 2006.

[121] R. Krishnan, R. Sandhu, and K. Ranganathan, "Pei models towards scalable, usable and high-assurance information sharing," in *Proceedings of the 12th ACM symposium on Access control models and technologies (SACMAT)*, 2007.

[122] R. Krishnan, R. Sandhu, J. Niu, and W. H. Winsborough, "Foundations for group-centric secure information sharing models," in *Proceedings of the 14th ACM symposium on Access control models and technologies (SACMAT)*, 2009.

[123] R. Krishnan and R. Sandhu, "A hybrid enforcement model for group-centric secure information sharing," *IEEE International Conference on Computational Science and Engineering*, pp. 189–194, 2009.

[124] C. E. Phillips, Jr., T. Ting, and S. A. Demurjian, "Information sharing and security in dynamic coalitions," in *Proceedings of the seventh ACM symposium on Access control models and technologies (SACMAT)*, 2002.

[125] V. Atluri and J. Warner, "Automatic enforcement of access control policies among dynamic coalitions," in *International Conference on Distributed Computing and Internet Technology, Bhubaneswar, India*, 2004.

[126] J. Warner, V. Atluri, R. Mukkamala, and J. Vaidya, "Using semantics for automatic enforcement of access control policies among dynamic coalitions," in *Proceedings of the 12th ACM symposium on Access control models and technologies (SACMAT), Sophia Antipolis, France*, 2007, pp. 235–244.

[127] B. Carminati and E. Ferrari, "Access control and privacy in web-based social networks," *International Journal of Web Information Systems*, vol. 4, pp. 395–415, 2008.

[128] B. Carminati, E. Ferrari, J. Cao, and K. L. Tan, "A framework to enforce access control over data streams," *ACM Transactions on Information and System Security (TISSEC)*, vol. 13, pp. 28:1–28:31, July 2010.

[129] B. Carminati, E. Ferrari, and M. Guglielmi, "Policies for composed emergencies in support of disaster management," in *Proceedings of the VLDB Workshop on Secure Data Management (SDM 2012)*, August 2012.

[130] B. Carminati, E. Ferrari, and A. Perego, "Rule-based access control for social networks," in *Proceedings of the IFIP WG 2.12 and 2.14 Semantic Web Workshop, LNCS, Springer.*, vol. 4278, Montpellier, France, November 2006, pp. 1734–1744.

[131] ——, "Private relationships in social networks," in *Proceedings of the IEEE 23rd International Conference on Data Engineering Workshop (ICDE2007)*, Istanbul, Turkey, 2007, pp. 163–171.

[132] B. Carminati and E. Ferrari, "Privacy-aware collaborative access control in web-based social networks," in *Proceeedings of the 22nd annual IFIP WG 11.3 working conference on Data and Applications Security*. London, UK: Springer-Verlag, 2008, pp. 81–96.

[133] B. Carminati, E. Ferrari, R. Heatherly, M. Kantarcioglu, and B. Thuraisingham, "A semantic web based framework for social network access control," in *Proceedings of the 14th ACM symposium on Access control models and technologies*, ser. SACMAT '09. Stresa, Italy: ACM, 2009, pp. 177–186.

[134] R. Baden, A. Bender, N. Spring, B. Bhattacharjee, and D. Starin, "Persona: an online social network with user-defined privacy," in *Proceedings of the ACM SIGCOMM conference on Data communication*, Barcelona, Spain, 2009.

[135] M. Srivatsa, D. Agrawal, and S. Reidt, "A metadata calculus for secure information sharing," in *Proceedings of the 16th ACM conference on Computer and communications security (CCS)*, Chicago, Illinois, USA, 2009, pp. 488–499.

[136] M. Srivatsa, S. Balfe, K. G. Paterson, and P. Rohatgi, "Trust management for secure information flows," in *Proceedings of the 15th ACM conference on Computer and communications security (CCS)*, Alexandria, Virginia, USA, 2008, pp. 175–188.

[137] A. C. Myers and B. Liskov, "A decentralized model for information flow control," in *ACM Symposium on Operating Systems Principles (SOSP)*, Saint Malo, France, October 1997, pp. 129–142.

[138] D. Chakrabarti, C. Faloutsos, and Y. Zhan, "Visualization of large networks with min-cut plots, a-plots and r-mat," *International Journal of Human-Computer Studies*, vol. 65, pp. 343–445, 2007.

[139] L. Freeman, "Visualizing social networks," in *Journal of Social Structure*, vol. 1, 2000.

[140] F. Vegas, M. Wattenberg, and K. Dave, "Studying cooperation and conflict between authors with history flow visualizations," in *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2004, p. 575582.

[141] F. Vegas, M. Wattenberg, J. Kriss, and F. Ham, "Talk before you type: Coordination in wikipedia," in *In Proceedings of the 40th Hawaii International Conference on System Sciences*, January, 2007, pp. 78–88.

[142] M. Sabel, "Structuring wiki revision history," in *In Proceedings of the 2007 International Symposium on Wikis*, October, 2007, pp. 125–130.

[143] B. Hoisl, W. Aigner, and S. Miksch, "Social rewarding in wiki systems motivating the community," in *In Proceedings of the second Online Communities and Social Computing*, July, 2007, pp. 362–371.

[144] D. M. Wilkinson and B. A. Huberman, "Cooperation and quality in wikipedia," in *Proceedings of the 2007 international symposium on Wikis (WikiSym '07)*, 2007, pp. 157–164.

[145] A. Kittur, B. Suh, B. A. Pendleton, and E. H. Chi, "He says, she says: conflict and coordination in wikipedia," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '07)*, 2007, pp. 453–462.

[146] R. Priedhorsky, J. Chen, S. T. K. Lam, K. Panciera, L. Terveen, and J. Riedl, "Creating, destroying, and restoring value in wikipedia," in *Proceedings of the 2007 international ACM conference on Supporting group work (GROUP '07)*, 2007, pp. 259–268.

[147] K. Smets, B. Goethals, and B. Verdonk, "Automatic vandalism detection in wikipedia: Towards a machine learning approach," in *In Proceedings of the AAAI Workshop, Wikipedia and Artificial Intelligence: An Evolving Synergy (WikiAI08)*, July, 2008.

[148] M. Potthast, B. Stein, and R. Gerling, "Automatic vandalism detection in wikipedia," in *In Proceedings of the Advances in Information Retrieval - 30th European Conference on IR Research*, March/April, 2008, pp. 663–668.

[149] B. T. Adler and L. de Alfaro, "A content-driven reputation system for the wikipedia," in *Proceedings of the 16th international conference on World Wide Web (WWW '07)*, 2007, pp. 261–270.

[150] H. Zeng, M. A. Alhossaini, L. Ding, R. Fikes, and D. L. McGuinness, "Computing trust from revision history," in *Proceedings of the 2006 International Conference on Privacy, Security and Trust: Bridge the Gap Between PST Technologies and Business Services (PST '06)*, 2006, pp. 8:1–8:1.

[151] B. Stvilia, M. B. Twidale, L. C. Smith, and L. Gasser, "Assessing information quality of a community-based encyclopedia," in *In Proceedings of the International Conference on Information Quality*, November, 2005, pp. 442–454.

[152] T. Cross, "Puppy smoothies: Improving the reliability of open, collaborative wikis," *First Monday*, vol. 11, no. 9, September 2006.

[153] P. Dondio and S. Barrett, "Computational trust in web content quality: A comparative evalutation on the wikipedia project," *In Informatica — An International Journal of Computing and Informatics*, vol. 31, no. 2, pp. 151–160, 2007.

[154] B. T. Adler, K. Chatterjee, L. de Alfaro, M. Faella, I. Pye, and V. Raman, "Assigning trust to wikipedia content," in *In Proceedings of the 2008 International Symposium on Wikis*, September, 2008.

[155] M. Richardson and R. W. White, "Supporting synchronous social q&a throughout the question lifecycle," in *In Proceedings of the 20th international conference on World wide web (WWW '11)*, Hyderabad, India, 2011.

[156] F. M. Harper, D. Raban, S. Rafaeli, and J. A. Konstan, "Predictors of answer quality in online q&a sites," in *In Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems (CHI '08)*, Florence, Italy, 2008.

[157] J. Jeon, W. B. Croft, and J. H. Lee, "Finding similar questions in large question and answer archives," in *In Proceedings of the 14th ACM international conference on Information and knowledge management (CIKM '05)*, Bremen, Germany, 2005.

[158] Y. Cao, H. Duan, C.-Y. Lin, Y. Yu, and H. W. Hon, "Recommending questions using the mdl-based tree cut model," in *In Proceedings of the 17th international conference on World Wide Web (WWW '08)*, Las Vegas, Nevada, USA, 2008.

[159] X. Cao, G. Cong, B. Cui, C. S. Jensen, and C. Zhang, "The use of categorization information in language models for question retrieval," in *In Proceedings of the 18th*

ACM conference on Information and knowledge management (CIKM '09), Hong Kong, China, 2009.

[160] M. A. Suryanto, E. P. Lim, A. Sun, and R. H. L. Chiang, "Quality-aware collaborative question answering: methods and evaluation," in *In Proceedings of the Second ACM International Conference on Web Search and Data Mining (WSDM '09)*, Barcelona, Spain, 2009.

[161] G. Dror, Y. Koren, Y. Maarek, and I. Szpektor, "I want to answer; who has a question?: Yahoo! answers recommender system," in *In Proceedings of 17th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2011)*. San Diego, CA: ACM, 2011.

[162] D. Horowitz and S. Kamvar, "The anatomy of a large-scale social search engine," in *In Proceedings of the 19th international conference on World wide web (WWW '10)*, Raleigh, North Carolina, USA, 2010.

[163] B. Li and I. King, "Routing questions to appropriate answerers in community question answering services," in *In Proceedings of the 19th ACM international conference on Information and knowledge management (CIKM '10)*, Toronto, ON, Canada, 2010.

[164] P. Jurczyk and E. Agichtein, "Discovering authorities in question answer communities by using link analysis," in *In Proceedings of the sixteenth ACM conference on Conference on information and knowledge management (CIKM '07)*, Lisbon, Portugal, 2007.

[165] M. Bouguessa, B. Dumoulin, and S. Wang, "Identifying authoritative actors in question-answering forums: the case of yahoo! answers," in *In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '08)*, Las Vegas, Nevada, USA, 2008.

[166] J. Zhang, M. S. Ackerman, and L. Adamic, "Expertise networks in online communities: structure and algorithms," in *In Proceedings of the 16th international conference on World Wide Web (WWW '07)*, Banff, Alberta, Canada, 2007.

[167] Y. Zhou, G. Cong, B. Cui, C. S. Jensen, and J. Yao, "Routing questions to the right users in online communities," in *In Proceedings of the 2009 IEEE International Conference on Data Engineering (ICDE '09)*, Shanghai, China, 2009.

[168] J. Guo, S. Xu, S. Bao, and Y. Yu, "Tapping on the potential of q&a community by recommending answer providers," in *In Proceedings of the 17th ACM conference on Information and knowledge management (CIKM '08)*, Napa Valley, California, USA, 2008.

[169] M. Liu, Y. Liu, and Q. Yang, "Predicting best answerers for new questions in community question answering," *Lecture Notes in Computer Science (LNCS)*, vol. 6184, pp. 127–138, 2010.

[170] M. Qu, G. Qiu, X. He, C. Zhang, H. Wu, J. Bu, and C. Chen, "Probabilistic question recommendation for question answering communities," in *In Proceedings of the 18th international conference on World wide web (WWW '09)*, Madrid, Spain, 2009.

[171] F. M. Harper, D. Moy, and J. A. Konstan, "Facts or friends?: distinguishing informational and conversational questions in social q&a sites," in *In Proceedings of the*

27th international conference on Human factors in computing systems (CHI '09), Boston, MA, USA, 2009.

[172] D. Bernhard and I. Gurevych, "Combining lexical semantic resources with question and answer archives for translation-based answer finding," in *In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2 (ACL '09)*, Suntec, Singapore, 2009.

[173] M. Surdeanu, M. Ciaramita, and H. Zaragoza, "Learning to rank answers on large online qa collections," in *In Proceedings of 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*, Columbus, Ohio, 2008.

[174] J. Bian, Y. Liu, E. Agichtein, and H. Zha, "Finding the right facts in the crowd: factoid question answering over social media," in *In Proceedings of the 17th international conference on World Wide Web (WWW '08)*, Beijing, China, 2008.

[175] X. Xue, J. Jeon, and W. B. Croft, "Retrieval models for question and answer archives," in *In Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '08)*, Singapore, Singapore, 2008.

[176] H. Duan, Y. Cao, C.-Y. Lin, and Y. Yu, "Searching questions by identifying question topic and question focus," in *In Proceedings of 46th Annual Meeting of the Association for Computational Linguistics: Human Language Tchnologies (ACL-HLT)*, Columbus, Ohio, 2008.

[177] K. Wang, Z. Ming, and T.-S. Chua, "A syntactic tree matching approach to finding similar questions in community-based qa services," in *In Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval (SIGIR '09)*, Boston, MA, USA, 2009.

[178] Z. Gyongyi, G. Koutrika, J. Pedersen, and H. Garcia-Molina, "Questioning yahoo! answers," in *In Proceedings of the 1st Workshop on Question Answering on the Web*, Beijing, China, 2008.

[179] C. Aperjis, B. A. Huberman, and F. G. Wu, "Harvesting collective intelligence: Temporal behavior in yahoo answers," *Computer Research Repository Journal*, 2010.

[180] K. K. Nam, M. S. Ackerman, and L. A. Adamic, "Questions in, knowledge in?: a study of naver's question answering community," in *In Proceedings of the 27th international conference on Human factors in computing systems (CHI '09)*, Boston, MA, USA, 2009.

[181] L. Guo, E. Tan, S. Chen, X. Zhang, and Y. E. Zhao, "Analyzing patterns of user content generation in online social networks," in *In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '09)*, Paris, France, 2009.

[182] F. Benevenuto, T. Rodrigues, M. Cha, and V. Almeida, "Characterizing user behavior in online social networks," in *In Proceedings of Usenix/ACM SIGCOMM Internet Measurement Conference (IMC)*, Chicago, Illinois, November 2009.

[183] B. Krishnamurthy, P. Gill, and M. Arlitt, "A few chirps about twitter," in *In Proceedings of the first workshop on Online social networks (WOSN)*, Seattle, WA, USA, August 2008.

[184] M. Burke, C. Marlow, and T. Lento, "Feed me: Motivating newcomer contribution in social network sites," in *In Proceedings of the 27th international conference on Human factors in computing systems (CHI '09)*, ACM Press, Boston, MA, 2009.

[185] Y.-Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong, "Analysis of topological characteristics of huge online social networking servicess," in *In Proceedings of the 16th international conference on World Wide Web (WWW)*, Banff, Alberta, Canada, 2007.

[186] R. Kumar, J. Novak, and A. Tomkins, "Structure and evolution of online social networks," in *In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (SIGKDD)*, Philadelphia, PA, USA, 2006.

[187] M. Marcon, B. Viswanath, M. Cha, and K. P. Gummadi, "Sharing social content from home: A measurement-driven feasibility analysis," in *In Proceedings of the 21st International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*, Vancouver, Canada, June 2011.

[188] H. Chun, Y.-H. E. H. Kwak, Y.-Y. Ahn, S. Moon, and H. Jeong, "Online social networks: Sheer volume vs social interaction," in *In Proceedings of the 6th ACM/USENIX Internet Measurement Conference (IMC'08)*, Vouliagmeni, Greece, October 2008.

[189] C. Wilson, B. Boe, A. Sala, K. P. N. Puttaswamy, and B. Y. Zhao, "User interactions in social networks and their implications," in *In Proceedings of the 4th ACM European conference on Computer systems (EuroSys '09)*, Germany, April 2009.

[190] J. Leskovec and E. Horvitz, "Planetary-scale views on a large instant-messaging network," in *In Proceedings of the 17th International World Wide Web Conference (WWW)*, Beijing, China, 2008.

[191] D. Liben-Nowell, R. K. J. Novak, P. Raghavan, and A. Tomkins, "Geographic routing in social networks," *Proceedings of the National Academy of Sciences (PNAS)*, vol. 102, no. 33, pp. 11 623–11 628, 2005.

[192] M. Valafar, R. Rejaie, and W. Willinger, "Beyond friendship graphs: a study of user interactions in flickr," in *In Proceedings of the 2nd ACM workshop on Online social networks (WOSN)*, Barcelona, Spain, August 2009.

[193] B. Huberman, D. Romero, and F. Wu, "Social networks that matter: Twitter under the microscope," *First Monday*, vol. 14, no. 1, 2009.

[194] M. Gjoka, M. Sirivianos, A. Markopoulou, and X. Yang, "Poking facebook: characterization of osn applications," in *In Proceedings of the first workshop on Online social networks (WOSN)*, Seattle, WA, USA, 2008.

[195] A. Nazir, S. Raza, and C.-N. Chuah, "Unveiling facebook: a measurement study of social network based applications," in *In Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*, Vouliagmeni, Greece, 2008.

[196] C. N. Chapman and M. Lahav, "International ethnographic observation of social networking sites," *In ACM CHI '08 extended abstracts on Human factors in computing systems*, 2008.