Graph-based Analytics for Decentralized Online Social Networks

AMIRA SOLIMAN

Doctoral Thesis in Information and Communication Technology
School of Electrical Engineering
and Computer Science
KTH Royal Institute of Technology
Stockholm, Sweden 2018

KTH
School of Electrical Engineering
and Computer Science
SE-164 40 Kista
SWEDEN

TRITA-EECS-AVL-2018:4
ISBN: 978-91-7729-666-9

Akademisk avhandling som med tillstånd av Kungl Tekniska högskolan framlägges till offentlig granskning för avläggande av doktorsexamen i informations och kommunikationsteknik fredagen den 9 mars 2018 klockan 09:00 i sal C, Electrum, Kungl Tekniska högskolan, Kistagången 16, Kista.

Tryck: Universitetsservice US AB

Abstract

Decentralized Online Social Networks (DOSNs) have been introduced as a privacy preserving alternative to the existing online social networks. DOSNs remove the dependency on a centralized provider and operate as distributed information management platforms. The main objective behind decentralization is to preserve user privacy in both shared content and communication. Current efforts of providing DOSNs are mainly focused on designing the required building blocks for managing the distributed network and supporting the social services (e.g., search for topics or people, content delivery, etc.). However, there is a lack of reliable techniques for enabling complex analytical services (e.g., spam detection, identity validation, recommendation systems, etc.) that comply with the decentralization requirements of DOSNs. In particular, there is a need for decentralized data analytic techniques and machine learning (ML) algorithms that can successfully run on top of DOSNs.

In this thesis, we empower decentralized analytics for DOSNs through a set of novel algorithms. Our algorithms allow decentralized analytics to effectively work on top of fully decentralized topology, when the data is fully distributed and nodes have access to their local knowledge only. Additionally, our algorithms follow unsupervised ML paradigm, thus removing the need for collecting labeled training data that potentially puts user privacy at risk. Furthermore, our algorithms and methods are able to extract and exploit the latent patterns in the social user interaction networks and effectively combine them with the shared content, yielding significant improvements for the complex analytic tasks. We argue that, community identification is at the core of the learning and analytical services provided for DOSNs. We show in this thesis that knowledge on community structures and information dissemination patterns, embedded in the topology of social networks has a potential to greatly enhance data analytic insights and improve results. At the heart of this thesis lies a community detection technique that successfully extracts communities in a completely decentralized manner. In particular, we show that multiple complex analytic tasks, like spam detection and identity validation, can be successfully tackled by harvesting the information from the social network structure. This is achieved by using decentralized community detection algorithm which acts as the main building block for the *community-aware learning* paradigm that we lay out in this thesis. To the best of our knowledge, this thesis represents the first attempt to bring complex analytical services, which require decentralized iterative computation over distributed data, to the domain of DOSNs. The experimental evaluation of our proposed algorithms using real-world datasets confirms the ability of our solutions to generate efficient ML models in massively parallel and highly scalable manner. Furthermore, our algorithms preserve user privacy and achieve better performance compared to the existing centralized and global approaches.

Sammanfattning

Decentralized Online Social Networks (DOSN) har införts som ett alternativ för att skydda privatlivet för de befintliga sociala nätverk på nätet. DOSN tar bort beroendet av en centraliserad leverantör och fungerar som distribuerade informationshanteringsplattformar. Huvudmålet bakom decentralisering är att bevara användarnas integritet för både delat innehåll och kommunikation. Nuvarande ansträngningar att tillhandahålla DOSN är huvudsakligen inriktade på att utforma de nödvändiga byggstenarna för att hantera det distribuerade nätverket och stödja sociala tjänster (t.ex. sökning, innehållsleverans, etc.). Det finns emellertid brist på tillförlitliga tekniker som att möjliggör komplexa analytiska tjänster (t ex spamdetektering, identitetsvalidering, rekommendationssystem etc.) som överensstämmer med decentraliseringskraven för DOSN. I synnerhet finns det behov av decentraliserade dataanalyser och maskininlärning (ML) -algoritmer som framgångsrikt kan köras ovanpå DOSN.

I denna avhandling tillhandahåller vi en decentraliserad analys för DOSN genom en uppsättning nya algoritmer. Våra algoritmer möjliggör att en decentraliserad effektivt kan fungera med en decentraliserad topologi, där data är helt distribuerade och noder endast har tillgång till local information. Dessutom följer våra algoritmer oövervakad en ML-paradigm och eliminerar därmed behovet av att samla klassifirad träningsdata som eventuellt riskerar användarens integritet. Våra algoritmer och metoder kan dessutom extrahera och utnyttja latenta mönster i sociala användargränssnittsnätverk och effektivt kombinera dem med det delade innehållet, vilket ger betydande förbättringar för våra komplexa analytiska metoder. Vi hävdar att gemenskapsidentifiering är kärnan i de lärande och analytiska tjänster som tillhandahålls för DOSN. Väsentligen visar vi i denna avhandling att kunskap om samhällsstrukturer och informationsspridningsmönster, inbäddade i topologin hos sociala nätverk, potentiellt kan förbättra insikter inom dataanalys och förbättra resultaten. Kärnan i denna avhandling ligger en communitydetekteringsteknik som framgångsrikt extraherar community på ett helt decentraliserat sätt. I synnerhet visar vi att flera komplexa analytiska metoder, till exempel skräppostdetektering och identitetsvalidering, kan hanteras framgångsrikt genom att skörda informationen från den sociala nätverksstrukturen med hjälp av en decentraliserad communitydetekteringsalgoritm som fungerar som huvudbyggstenen för det communitysmedvetna lärande paradigmet som vi presenterar i denna avhandling. Så vitt vi vet är den här avhandlingen det första försöket att medföra komplexa analytiska tjänster, som kräver decentraliserad iterativ beräkning över distribuerad data, till domänen för DOSN. Den experimentella utvärderingen av våra föreslagna algoritmer med hjälp av dataset i realtid, bekräftar möjligheten att våra lösningar skapar effektiva ML-modeller på ett massivt parallellt och högskalbart sätt, utan att bryta mot integritetsskydd för begränsningar och uppnår bättre prestanda jämfört med befintliga centraliserade och globala tillvägagångssätt.

To the memory of my father …
Dad: in my heart you will always
be loved and remembered.

Acknowledgements

He who does not thank people, does not thank God.
— Prophet Muhammad, PBUH

I believe the majority of PhD students, including myself, consider PhD as a journey; a journey of acquiring knowledge alongside doing our research. During this journey, I am deeply thankful to the following people, without the help and support of whom, I would not have managed to complete this work.

I would like to open the long list of my owed thanks by acknowledging my appreciation and gratitude to my PhD advisor, *Assoc. Prof. Sarunas Girdzijauskas*. I am sincerely grateful for your advise, guidance, patience and time devoted to not only the work presented in this thesis, but also for the knowledge and motivation you gave me to become a better researcher. I could not imagine having a better mentor than you.

I would like to express my deepest gratitude to my second advisor, *Prof. Seif Haridi*. Your broad knowledge and expertise, as well as your continuous support, provided me with a great opportunity and excellent atmosphere for doing research.

Like every long journey, PhD begins with fear. I consider myself lucky to meet *Dr. Fatemeh Rahimian* when I started my PhD, then having her as a co-supervisor. I am so grateful to your constructive discussions and feedback. More than this, I am extremely thankful for your invaluable support through the hardships and disappointments that I faced.

I would also like to take the opportunity to acknowledge the support I received during this work from the iSocial EU Marie Curie ITN project (FP7-PEOPLE-2012-ITN). I would like also to express my thanks to all the iSocial project members, both students and supervisors, for the fruitful discussions and the sharing of knowledge we had. I am also very much obliged to *Prof. Elena Ferrari* and *Assoc. Prof. Barbara Carminati* for broadening my knowledge in the fields of privacy and trust in social networks.

I would like also to thank *Assoc. Prof. Vladimir Vlassov*, *Prof. Christian Schulte*, and *Alf Thomas Sjöland*. It has been a great opportunity working alongside you and benefiting from your advices, experiences and warm encouragements.

As it is commonly known that "Good company in a journey, makes the way seems shorter", I feel the need to express gratitude to my dearest friend *Dr. Leila Bahri*. I am thankful to you not only for the collaboration we had, but also for your generous help and support during the whole course of my studies. I would like also to thank my colleagues and friends at EECS, especially *Anis Nasir*, *Kambiz Ghoorchian*, *Shatha Jaradat*, *Kamal Hakimzadeh*, and *Zainab Abbass* for all the innovative talks we had during our lunch breaks.

I also must thank all of the reviewers of my work, for their instructive feedback and comments on my work.

I would like to thank the PhD office at ICT, particularly *Susy Mathew*, *Madeleine Printzsköld*, and *Emanuel Borg* for their support during all administrative issues during the course of this PhD.

I would also like to take the opportunity to thank my flatmate *Tara Munim*, for her support via all the funny discussions we had in order to get me out of bug-fixes and daily work routine.

Geographically we have been separated, but we have been so close to each other, even during our dreams. My dearest mom *Nadia Kotb* and my dearest sister *Asmaa Soliman*, thanks a lot for providing me the emotional support during the journey.

Lastly, but not least, big thanks to my dear friends. I am very proud of your friendship, and very grateful for your blissful existence in my life. Loads of love and thanks to you all!

Contents

Chapter 1

Introduction

The most valuable commodity I know of is information.
— Stanley Weiser and Oliver Stone, Wall Street (1987 film)

Big data has transformed our digital world in the last few years. The avalanche of collected data shows a great potential of extracting new knowledge about our world. Furthermore, big data plays a key role in advancing machine learning (ML) techniques and providing smart applications for mobile phones, smart houses, internet of things and artificial intelligence. Although, ML techniques have been used for years, yet only now they are capable of providing meaningful knowledge and insights by taking the advantage of data coming from online footprints of millions of users. Web cookies, online footprints, decades of search results, and Wikipedia, represent some of the examples of massive dataset sources for many ML applications, such as automatic translation, object recognition, medical diagnosis, etc. Though, the collected data provides the schooling that is needed for ML training, usually this data is not owned by the people who generated it in the first place. The data is being funneled to the centralized repositories managed by giant companies like Google, Facebook, etc., thereby undermining user privacy and permitting the potential rise of the omnipresent figure of Big-Brother. For example, face recognition technology powered with the data from millions of CCTV[1] cameras allows Chinese authorities to identify and apprehend any individual in a matter of minutes[2]. Online social networks (OSNs) are among the primary and significant sources of massive datasets that can be used to support ML in multiple domains. Users of OSNs do not merely consume data, but also produce data at a considerable rate. People share photos and videos, write posts, even advertise and sell personal items online. Undeniably, user data represents the new form of currency that is used to monetize the social services offered by current OSNs providers. Specifically, users upload their data to the OSN provider website and their data is also used for analytical purposes that might put user privacy at risk.

[1]Closed-Circuit Television (CCTV), also known as video surveillance, is the use of video cameras to transmit observations of the camera to be displayed on monitors.

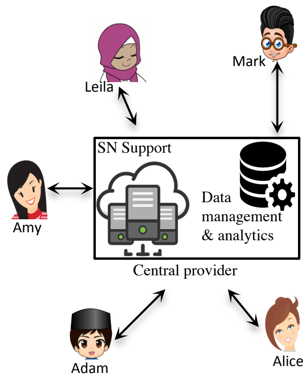
[2]https://techcrunch.com/2017/12/13/china-cctv-bbc-reporter/

OSN providers incentivize people with free services to share data in their OSNs public space, yet users do not have a clear idea about who utilizes their data and for what purposes. In addition to user friends, these pieces of information can be accessible by third-party applications or external data aggregators [1]. Specifically, this kind of data usage is done without users' knowledge in unexpected ways different from that for which it was originally shared. For example, some governmental authorities have used geo-located tagged data available on social networks to infer people's residence and mobility patterns, to supplement official population estimates [2]. More disturbingly, a recent study has used deep neural networks to extract features from facial images and relate that to very personal behavior that one might not want it to be revealed to avoid any discrimination or life threats [3].

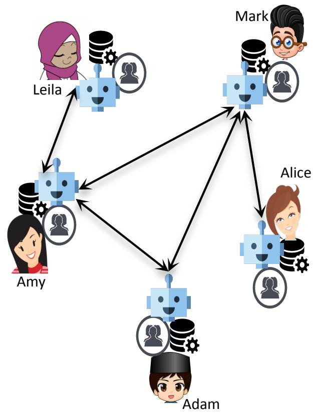
Regardless of the incentive behind the data analytic tasks, such repurposing use of data puts users' privacy at risk. Particularly, data analytics can be used to mine data for finding new insights and correlations between apparently disparate datasets, sometimes drawing conclusions about individuals in undesirable manner. As an attempt to protect users, information privacy or data protection laws have been agreed on, such as General Data Protection Regulation (GDPR)[3] in Europe, to prohibit the disclosure or misuse of information about individuals. Yet, there is a lack of reliable technologies that provide enhanced privacy preserving ML algorithms. Therefore, there is a need to develop new data management and analytic services that follow privacy preserving regulations. Arguably, decentralized learning schemes and data processing is one of the most promising ways to provide data analytics in DOSNs and eliminate any need for central data aggregation.

In the last decade, researchers and the open source community have proposed various decentralized OSNs (DOSNs) (e.g., [4, 5, 6, 7]) with the objective of preserving user privacy in both shared content and communication. DOSNs remove the dependency on a centralized provider, thus represent a promising alternative for diminishing forms of censorship or profiling. Most importantly, DOSNs aim primarily at giving the control back to the users over their personal data. DOSNs allow users to share various resources and information with the assumption that every user is equally privileged participant and provider of various resources inside the network. Accordingly, DOSNs can be modeled as distributed systems consisting of a set of nodes that communicate by sending messages over a network without central manager [1]. DOSNs are implemented using various topological overlays. One example of DOSN overlay topology can be similar to the social graph, where nodes communicate with their direct friends (i.e., fully decentralized). Structured overlay can be another possible topology for constructing DOSN overlay, where some nodes are selected as superpeers to act as proxies for other nodes in the system (i.e., federated architecture). Figure 1.1 shows how interactions among users are different in centralized and fully decentralized social network frameworks. Having a central provider enforces pure client-server coordination, such that every interaction goes through the central provider that operates as a mediator among users. Furthermore, data analytic tasks are performed using centrally aggregated data from all users. On the other hand, users in DOSNs have their dedicated agents that are responsible for providing both social network support and data manage-

[3]https://www.eugdpr.org/



(a) Centralized architecture of social network frameworks which requires uploading a copy of user's data to provider's datacenter. Also, every interaction among users has to go through the provider.



(b) Decentralized architecture of social network frameworks which requires preserving users' privacy. Users interact in P2P fashion, such that every node has an agent that is responsible for providing social network support and data analytic services.

Figure 1.1: Comparison of interactions among users in both (a) centralized architecture, and (b) decentralized architecture, of social network frameworks. In case of centralized architecture users have to upload their data to the provider's datacenter. On the other hand, data remain at the original user device in decentralized architecture.

ment services. Specifically, those agents communicate directly in a peer-to-peer (P2P) manner to fulfill the requirements of providing the social and management services, such as searching, messaging, access control, etc.

DOSNs have the potential to provide a better environment within which users can have more control over privacy, ownership, and dissemination of their data. However, currently there is a lack of reliable privacy preserving ML algorithms, thus data access and analytic services need to be accordingly designed for DOSNs. Specifically, ML algorithms follow iterative data-flow paradigm, such that the algorithms apply operations repeatedly using the whole data collection that is stored in one central location (or multiple federated locations), till the stopping criteria get achieved. However, in fully decentralized DOSNs the main challenge is to apply this iterative data-flow paradigm of ML algorithms under the limitation of not moving the users' data away from their trusted space (i.e., their personal deceives or other trusted storage service they are using). Accordingly, user data has to be processed not in any central repository, but locally at each participating node preferably communicating only the learned models, instead of raw-data. Furthermore, social networks are continuously evolving systems which produce not only the shared content, but also the rich metadata on the interactions happening among users. Many insights can be captured by analyzing users' interactions that are represented by the underlying social graph. Specifically, the social graph provides the metadata of how users are connected and this can support ML algorithms with insights about how users are grouped, how information is disseminated on top of social ties, etc. Thus, ML algorithms need to analyze not only the shared content, but also the clues from the underlying social graph in order to provide richer analytics that integrate the two dimensions of the available data (i.e., shared content and network information).

This thesis aims at developing algorithms and methods that support decentralized ML analytics on top of DOSNs. Our algorithms enable ML tasks to extract patterns while analyzing the data available from both the shared content and the topological structure information (i.e., graph-based analytics). This thesis enables graph-based analytics effectively work on top of fully decentralized topology, specifically we consider the data is fully distributed and nodes have only local knowledge about the underlying social graph (i.e., nodes are aware only of their direct friends). The main statement of this thesis is:

> *The proposed graph-based algorithms and methods in this thesis allow generation of efficient ML models using user local data in fully decentralized, iterative, massively parallel, and highly scalable manner, thereby enabling decentralized ML and analytic tasks on DOSNs, without violating privacy preservation constraints.*

1.1 Research Motivation

Our research motivation is twofold; performing ML analytics in privacy preserving manner and providing fully decentralized algorithms. Several techniques of privacy preserving data mining have been proposed in literature that deal with hiding an individual's sensitive identity without sacrificing the usability of data [8, 9]. The major area of concern is that even non-sensitive data may still deliver sensitive information, including personal information, facts or patterns. Thus, cryptography[4] and data perturbation[5] techniques have been used to hide users' identities when the data is being shared between multiple authorities [8, 9]. However, these methods not only are slow to be effective for large-scale big data applications, yet also do not allow performing data analytic tasks as the data is encrypted. Furthermore, the approach of federated data analysis has been proposed to perform distributed privacy-preserving analytics that can be executed in parallel to speed up the computation process [10, 11]. Specifically, federated data analysis assumes that the data is distributed among a set of trusted servers that coordinate among themselves the data distribution policy. After data partitioning, the data analysis tasks are performed in parallel, such that each server applies different types of ML algorithms (i.e., regression, classification, etc.) over its local data. Then, the estimation of global model parameters can be obtained by exchanging the computed statistics and aggregating them. Every node in a DOSN is going to be considered as a server in case of applying the federated data analysis scenario. However, DOSN environment is different from the environment for which

[4]Cryptography is a technique through which user data can be encrypted.

[5]Data Perturbation is used for modifying data using random process like adding, subtracting or any other mathematical formula.

federated data analysis methods are designed. First of all, federated data analysis methods assume that there is homogeneity among different data partitions, whereas, in DOSN the data at each node depends solely on user's behavior with no guarantees on the quality or sufficiency of this data for learning purposes. Additionally, federated data analysis assumes that there exists a direct communication among all participating servers, while in DOSNs users are only aware of their direct neighbors (i.e., in addition to friends in the social graph, neighbors also include nodes that are linked in DOSN overlay).

Most of the existing research in DOSNs targets issues that are related to overlay management [12, 13, 14], data availability [15, 16], access control [17, 18, 19, 20], and information dissemination [21, 22]. The objective of these solutions is to decentralize the existing functionality of OSNs, like providing methodologies for decentralizing data storage and update propagation, as well as searching and addressing protocols. The core of the proposed methods lies in organizing the participating nodes into structured and socially-aware P2P overlays that reflect the topology of the social graph and exhibit small-world phenomena [23]. Specifically, these overlays allow a socially-informed routing among the nodes participating in the P2P layer. In these overlays, users are connected to a number of users besides their direct friends in order to provide better reachability and minimize the communication overhead. However, all of these methods provide more simplistic services (storage, search, information dissemination, etc.), but not suitable for performing iterative ML and analytic tasks. Furthermore, communicating following friend-to-friend (F2F) links remains the most intuitive communication pattern for DOSNs, as it requires users to be only aware of their direct friends in the social graph.

In addition to these challenges, the models extracted using ML analytics should reflect the different behavioral patterns exhibited by the users in social networks. Particularly, social networks are commonly known to show presence of homophily that is the tendency of users to associate with those similar to themselves. This phenomenon affects the way users are connected in the underlying social graph, showing that users topologically cluster into groups (or communities) with intra-ties denser than inter-ties [24]. These communities tend to share common properties, such that interactions between similar users occurs at a higher rate than among dissimilar ones. For example, friendship ties are concentrated within groups of users representing scientists working on the same research topics, where collaborations are more natural. It would be more beneficial for these users to get recommendation on events that are related to their research topics, such as conference calls or talks, rather than the trending news of a famous singer, for example. Thus, ML analytics need to extract patterns that reflect the behavior exhibited in every community independently, instead of showing a global behavior. Yet, it is challenging to consider a wide variety of possible communities, such as families, work and friendship circles, villages, towns, and nations.

In this thesis, we address the following challenges, that are vital for performing privacy preserving analytics for fully decentralized DOSNs:

- Preservation of privacy is an important aspect of DOSN design goals. Accordingly, the privacy should be preserved in all of the steps of data analytic components. **Data needs to be accessed, processed, and utilized locally without sacrificing the pri-**

vacy of the participating nodes. Furthermore, people tend to limit their trust to the friends to whom they are connected in social networks. Thus, communication among participating nodes needs to follow friend-to-friend (F2F) links.

- The learning paradigm needs to reflect the properties of the underlying social graph, especially the community structure. **Thus, analytical components need to extract patterns that exist at the community level, instead of the global behavior**.

- Data analytic component is required to efficiently handle the iterative nature of ML paradigm. **Particularly, the data analysis tasks need to be performed in fully decentralized manner, while allowing only the exchange of locally computed results**. Furthermore, it is required to diminish the need for a priori knowledge regarding the application domain (i.e., apply unsupervised ML techniques) and execution environment in order to avoid violating the privacy preserving constraints.

- Services in DOSNs are provided with the cooperation among the participating nodes. Yet, it is not expected that all nodes behave exactly as described in the workflow protocols. Malicious and misbehaving nodes can easily participate in such open environments. **Thus, DOSNs require mechanisms that guarantee the system integrity and robustness under the existence of these malicious nodes**.

Figure 1.2 shows a comparison between the four possible system architecture settings, with respect to the system architecture and the assumed ML model, under which data analytics can be performed. As shown, the decentralized and unsupervised setting is the one under which DOSNs requirements can be met. Therefore, we believe that in order to have a successful shift from current OSNs to DOSNs, it is vital to adopt decentralized and unsupervised architecture setting to enable analytic capabilities on DOSNs.

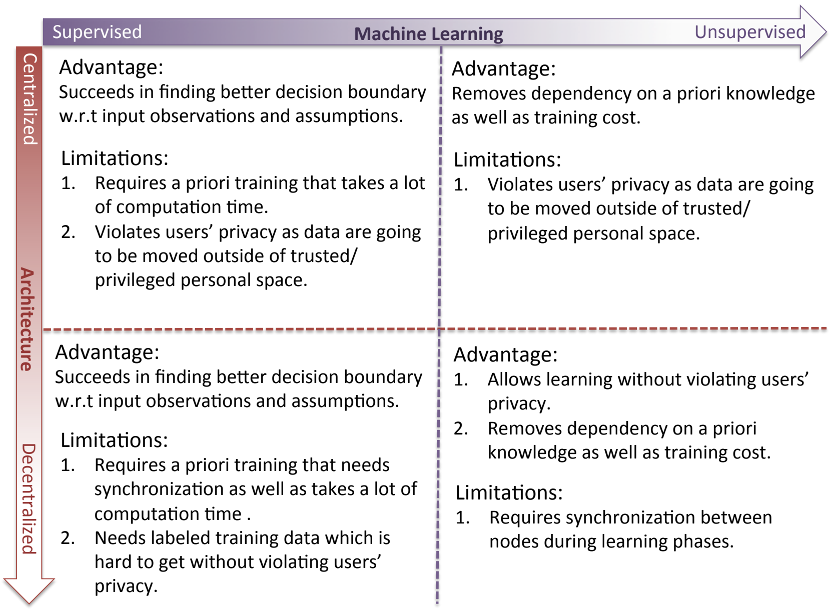


Figure 1.2: A comparison among the possible data analytic approaches with respect to the type of architecture (y-axis) and the adopted ML scheme (x-axis).

1.2 Research Objectives

Social networks are categorized as complex systems, capturing the fact that it is difficult to derive their collective behavior by extracting knowledge from shared data without considering the complex network that encodes the interactions between users. Accordingly, there is a need to investigate the best approaches to combine network information with user generated data when performing data analytics. Specifically, this integration extracts knowledge that encodes both of topological and behavioral interactions between users, as well as patterns extracted from the shared data. Furthermore, it is challenging to provide highly scalable ML components with respect to the huge number of users and the large amount of social data streams, such as expected in DOSNs.

Following a previously proposed general architecture of DOSNs [1], Figure 1.3 depicts our vision of the required components for DOSNs. As shown, from bottom to up, the lowest layer is the distributed network support, which is responsible for managing physical network communication among participating nodes, as well as managing the infrastructure resources needed for the system. On top of this comes the distributed storage layer, which is required to implement the functionalities of a distributed or P2P data management system to query, insert, and update various objects to the systems. The social networking layer implements all basic services and features that are provided by contemporary cen-

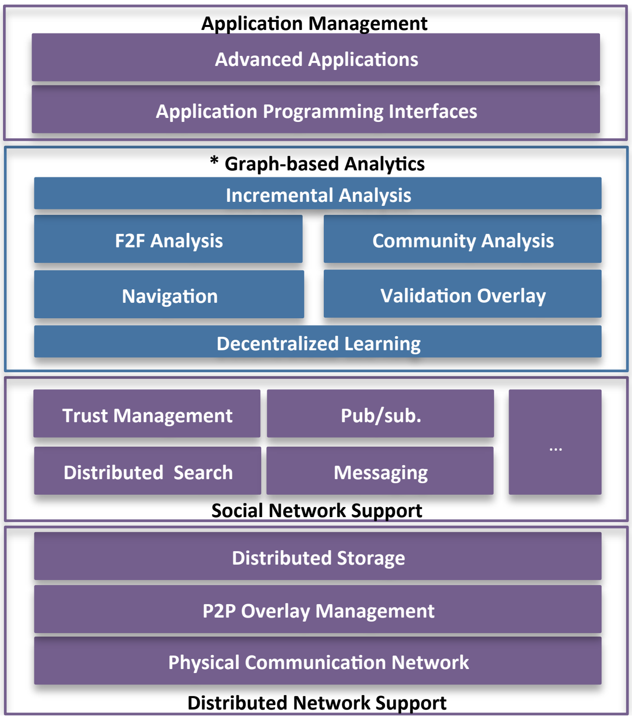


Figure 1.3: The proposed layered architecture for DOSN framework that extends the architecture proposed in [1]. The stack of services can be decomposed into different layers including physical communication, distributed storage, social network support, graph-based analytics, and application management, from bottom to up.

tralized social networking services, such as the capability to search the system for relevant information, the management of users and shared space, the management of security and access control issues, and finally the coordination and management of social applications developed by third parties. On top of this comes the layer of graph-based analytics, which is the focus of this thesis. This layer provides different algorithms for performing data and graph analytics in a decentralized and privacy preserving way. The highest layer of the architecture includes the application programming interfaces to the system and various applications built on top of the development platform provided by the DOSN. The application management is expected to provide the user with the necessary transparency to use the services as any other centralized OSN.

The main goal of this work is to research methods and techniques, as well as to build tools for easy, efficient, and privacy preserving analytics for decentralized social network frameworks. We aim at making data analytics in DOSNs richer, by integrating graph analytics for such distributed environments. Our vision is to eliminate the need for data collecting and processing in one central place, hence enabling processing user data locally without violating any privacy constraints. Additionally, our research is carried out and the developed algorithms are designed based on the fact that social networks show patterns of homophily, as similar users topologically cluster into groups with intra-ties denser than inter-ties. Following this behavior, our objective is to design graph-based analytic components for efficient coupling of data and network analysis. As shown in Figure 1.3, any learning task needs to be divided to multiple components that are trained separately using different training data, such that each component produces a model (i.e., Decentralized Learning). Then, the final outcome (model) is obtained by combining diverse models. Particular applications might require combining the models generated by node's direct friends (i.e., F2F Analysis), while others might require generating models that represent the communities existing in the social network (i.e., Community Analysis). Furthermore, the distributed and open nature of DOSNs gives the chance to malicious nodes to participate in the services provided by the system. This makes DOSNs vulnerable to attacks where malicious nodes present false information to either prevent the system from stabilizing or to distort the outcome of the learning process, resulting in inaccurate models. Thus, we need to provide services and tools that not only are resilient to malicious acts, yet also are capable of excluding the misbehaving nodes from the system. Therefore, ML on DOSNs need a decentralized mechanism by which reported information by any node is validated before being confirmed to the generated model (i.e., in this thesis we achieve this through our novel validation overlay). Yet, DOSNs enforce a restricted communication pattern, thus navigation mechanism is required that makes it possible to reach other nodes in the network that are participating in the validation overlay (i.e., navigation mechanism to construct paths to reach validators that are not directly connected in the social graph via some intermediate nodes).

Based on the above discussion, we can summarize the core requirements for ML analytics for DOSNs, that we have taken into account while designing all of the algorithms provided in this thesis:

- **Privacy preserving decentralized analytics**. DOSNs require data not to be collected to a central repository, instead data should be accessed locally. Accordingly, our proposed solutions follow vertex-centric approach which is centered on individual nodes performing local operations using only their local information.

- **Community-aware learning**. Social networks are known to exhibit homophily, where users tend to associate and bond with similar ones and form groups (communities). Thus, we perform the learning tasks on top of communities that are extracted using a decentralized algorithm designed to provide concrete views of all behavioral interactions and patterns exhibited by the users in a DOSN.

- **Adaptive and evolving methods**. Social networks are dynamic by nature due to rapidly evolving social activities and interactions among users. Therefore, we provide adaptive algorithms that enable data and graph analytics to cope with evolving user behavior.

- **Cooperation with low communication overhead**. DOSNs operate as distributed information management platforms on top of networks of trusted servers or P2P infrastructures. Therefore, performing analytics in such distributed environments requires a lot of direct communication among participating nodes. In our research, we design efficient cooperative protocols with low communication overhead.

- **Misbehavior detection**. Users in DOSNs participate with their devices to provide the full stack of services needed in the system without central monitoring or security checks. This can affect the system integrity and robustness, especially when it comes to adversarial manipulations. Therefore, we provide validation mechanism that detects misbehaving nodes and disconnects them from the system.

- **Unsupervised ML algorithms**. The learning tasks are required to employ unsupervised machine learning scheme in order to eliminate the need for labeled training data and training cost, as well as global knowledge about social graph. Therefore, all of our designed algorithms operate based on the unsupervised ML approach.

The work presented in this thesis follows these design objectives and provides scalable and decentralized graph-based analytics for DOSNs, without violating privacy preserving constraints.

1.3 Thesis Contributions

This thesis makes the following novel contributions in providing decentralized and community-aware ML for DOSNs:

- We propose novel algorithms that leverage both decentralization and privacy preserving properties, as well as support scalable data access solutions. Our algorithms follow node-centric approach which is centered on individual nodes instead of overgeneralizing global behavior paradigms. Specifically, node-centric approach translates the global application behavior in terms of local actions on each node, and individually allows the nodes to work in parallel and autonomously using their local knowledge. This makes our algorithms completely inline with DOSN requirements.

- We combine graph analytics with decentralized machine learning to take into account the underlying community structure. By this integration, we enable community-aware learning that allows us to analyze autonomous data sources as well as user interactions in a decentralized way that suitably fits DOSNs. Our algorithms employ community detection as a core analytic component for analyzing topological users interactions. Moreover, our algorithms follow unsupervised machine learning scheme, thus they remove the need for labeled training data and training cost.

Furthermore, this thesis makes the following novel contributions in the fields of community detection, spam detection, and identity validation for DOSNs:

- To enable community-aware learning, we present Stad (the 1st publication) that provides a decentralized community detection algorithm that employs stateful diffusion.

Differently from existing diffusion methods that operate with fixed diffusion speed, our approach boosts diffusion with conductance-based function that acts like a tuning parameter to control the diffusion speed at community level. Thus, our approach is able to extract communities more accurately in heterogeneous cases by overcoming the limitations of "one size fits all" model.

- We address the problem of providing a decentralized mechanism for spam detection. We present AdaGraph (the 2nd publication) which provides a graph-based spam detection technique that detects spam using graph clustering. Our approach performs graph-based spam detection using graph clustering on top of a newly constructed user similarity graph which encodes within its topology a holistic view of all behavioral patterns of social network users. Our method integrates community detection algorithm that categorizes the existing user behavioral patterns into more homogeneous and accurate clusters than binary classification.
- We present a novel validation mechanism that is capable of preventing adversarial manipulations in fully decentralized settings. We provide DLSAS (the 3rd publication) as anti-spam framework that provides defense mechanisms for DOSNs to prevent malicious nodes from participating in the system. Particularly, our framework creates a validation overlay to assess the credibility of the exchanged information among the participating nodes and excludes the misbehaving nodes from the system. Consequently, our framework preserves the core of spam detection proposed in AdaGraph from manipulation under the existence of adversarial nodes that deviate from the designed workflow protocol, in purely decentralized settings.
- We provide identity validation methods (publications 4 and 5) that allow users to assign trust levels to whomever they interact within a social network framework, without the need of any centralized monitoring or security checks. Our identity validation schemes can be successfully applied in both OSN and DOSN frameworks. Our methods conceptualize user online identities by mining the correlations among user profile attributes not from user population as a whole, but from individual communities, where the correlations are more pronounced.

1.4 List of Publications

List of the publications that are included in this thesis:

1. Soliman, A., Rahimina, F., and Girdzijauskas, S. Stad: Stateful Diffusion for Linear Time Community Detection. Under review, submitted to ICDCS.

   Contribution: The author of this thesis designed and implemented the proposed community detection approach presented in this paper, performed the experimental analysis, wrote majority of the text of the paper, and designed the figures.

2. Soliman, A. and Girdzijauskas, S. (2017). AdaGraph: Adaptive Graph-based Algorithms for Spam Detection in Social Networks. In International conference On

Networked Systems (NETYS 2017), Springer, pages 338-354.

Contribution: The author of this thesis designed and implemented the proposed spam detection approach presented in this paper, collected the datasets, performed the experimental analysis, wrote majority of the text of the paper, and designed the figures.

3. Soliman, A. and Girdzijauskas, S. (2016). DLSAS: Distributed Large-Scale Anti-Spam Framework for Decentralized Online Social Networks. Invited paper in the 2nd IEEE International Conference on Collaboration and Internet Computing, IEEE, pages 363-372.

   Contribution: The author of this thesis designed and implemented the proposed validation methods presented in this paper, performed the experimental analysis, wrote majority of the text of the paper, and designed the figures.

4. Soliman, A., Bahri, L., Carminati, B., Ferrari, E., and Girdzijauskas, S. (2015). DIVa: Decentralized Identity Validation for Social Networks. In International Conference on Advances in Social Network Analysis and Mining (ASONAM), 2015 IEEE/ACM, pages 383-391.

   Contribution: The author of this thesis designed and implemented the proposed identity validation scheme presented in this paper, performed the experimental analysis, wrote majority of the text of the paper, and designed the figures.

5. Soliman, A., Bahri, L., Girdzijauskas, S., Carminati, B., and Ferrari, E. CADIVa: Cooperative and Adaptive Decentralized Identity Validation Model for Social Networks. Social Network Analysis and Mining 6, no. 1 (2016): 1-22.

   Contribution: The author of this thesis designed and implemented the proposed identity validation model presented in this paper, performed the experimental analysis, wrote majority of the text of the paper, and designed the figures.

List of the publications that are not included in this thesis:

6. Bahri, L., Soliman, A., Squillaci, J., Carminati, B., and Ferrari, E. and Girdzijauskas, S. (2016). Beat the DIVa -Decentralized Identity Validation for Online Social Networks. Demo paper in the 32nd IEEE International Conference on Data Engineering. **Contribution**: The author of this thesis participated in designing the game model proposed in this paper.

1.5 Thesis Organization

This thesis is organized as follows. Chapter 2 covers the background related to the proposed algorithms in this thesis. Chapter 3 provides a summary of the papers included in this thesis, while Chapter 4 concludes the contributions of the performed research. The complete publications, on which this thesis is based, are presented afterwards.

Chapter 2

Background

> Never trust anything that can think for itself if you cannot see where it keeps its brain.
> — J.K. Rowling, Harry Potter and the Chamber of Secrets

In this chapter, we provide background information on the core protocols and concepts based on which we built our data analytic solutions for DOSNs. We start the chapter by describing the existing approaches for distributed machine learning. Then, we discuss more practical decentralized learning solutions that can suit large-scale and decentralized environments, such as DOSNs. We also discuss the research topics that are related to this thesis, namely: community detection, spam detection, and identity validation.

2.1 Distributed Machine Learning

Nowadays, the amount of data to be analyzed can be too large to be effectively handled by a single machine learning task. For example, users generate content with petabytes of data every day on Facebook[1]. Thus, performing a machine learning task (e.g., training a classifier) with such a vast amount of data is usually not practical, yet partitioning the data into smaller subsets, training different classifiers with different partitions of data in parallel, and combining their outputs using an intelligent combination rule often proves to be a more efficient approach. Not only size of data presents a motive behind designing different approaches for distributed machine learning, but also the technological advances in producing multi-core architectures and the use of graphical processing units for high performance computing drive the development of distributed machine learning. The discipline of distributed machine learning has been largely investigated with main focus on

[1] Big data challenges as one of Facebook's open data problems, reported on https://research.fb.com/facebook-s-top-open-data-problems/

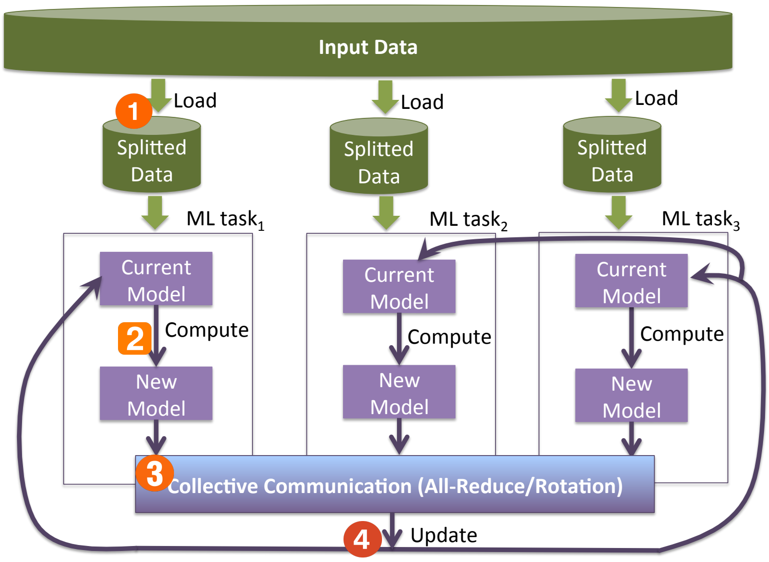


Figure 2.1: Steps required for performing machine learning (ML) tasks in distributed environments. First, the input data can be partitioned among different worker (Step 1: load). Then, each worker can compute its local model based on its local data (Step 2: compute). Afterwards, workers need to exchange their local models in order to achieve consensus by reaching a global model (Step 3: collective communication). Workers can broadcast their local models to everyone, then aggregate incoming models, or they can agree among themselves on the order of exchange for updating local models (e.g., rotation: $ML\ task_1$ contacts $ML\ task_2$ while sending its local model, $ML\ task_2$ updates its local model and sends the updated model to $ML\ task_3$, and finally $ML\ task_3$ updates its model with the newly received model and sends back to $ML\ task_1$ the aggregated model). After synchronizing the computed models at this iteration, the workers proceed by repeating the same steps for the successive iterations till convergence.

providing new abstractions for programming and computation models, especially with the spread adoption of MapReduce[2] as a distributed programming paradigm.

Figure 2.1 describes the main steps of a distributed computing model. First, data can be stored in a distributed file system and be processed by multiple workers. Each worker executes a separate machine learning task, and calculates the data statistics based on the local data sources, to later exchange the statistics with other workers to achieve a global data distribution view. By exchanging local results between multiple sources, new global patterns can be synthesized by aggregating computed results across all workers. To better understand this process, we give the example of a recommendation system that wants to predict the rating that a user would give to an item (e.g., the rating a Netflix user would give to a new movie). Such a recommendation system would usually be achieved using collaborative filtering techniques, that predict the rating user *A* would assign to an item based on the ratings that have already been assigned by other users who are similar to *A* (e.g., people who watched the same movie set as *A*). This task can be solved using the described distributed machine learning model by performing the following steps [25, 26]. (1) Split the input samples among participating workers, where input data describes taste information for every user, for example movie ratings. (2) Learn the model parameter based on the training dataset at each worker, where the model parameters can be estimated and further optimized using different techniques, such as matrix factorization or stochastic gradient descent. (3) Upload obtained parameters by each worker to a common parameter pool to collaboratively compute the set of parameters to be used later in the following iterations. (4) Update the current model at each worker, in order to proceed with the successive iterations.

It is important to notice that, such distributed programming model preserves user privacy at personal data level. Particularly, participating workers exchange only computed statistic aggregates without disclosing individual user data. However, it cannot be successfully implemented in DOSNs for practical reasons. First of all, workers are going to be the agents responsible for each user in a DOSN. Accordingly, the local learning step is going to be performed for every node in the social network. Even with the expected huge number of users who would be engaged, this local learning can be performed in parallel, not affecting as such the performance of the system. However, the bottleneck lies in the successive steps needed for synchronization (i.e., the collective communication and updating local models). Furthermore, existing distributed programming models assume that the participating machines maintain a global knowledge of the system (i.e., information of participating machines and how to reach them, as well as the agreement of collective mechanism with minimum communication overhead). On the other hand, fully decentralized DOSNs support only F2F communication, as nodes are only aware of their direct friends. Noting that, F2F analysis is expected to produce learning models that are biased to the very limited data with also limited variety, resulting as such in under-trained models with low performance and that cannot be generalized. Accordingly, DOSNs require designing efficient aggregation mechanisms that can be used to achieve stable representative models that overcome the limitations of F2F analysis, without breaking the DOSN privacy

[2]https://hadoop.apache.org/

requirements. We discuss these approaches in the next section.

2.2 Gossiping Protocols

Gossip protocols were initially used as a way to maintain consistency on databases that were replicated at several hundred sites [27]. From then on, gossip protocols have been used to solve other problems, like data aggregation (e.g., calculating averages across a network of nodes) [28], failure detection [29, 30], or topology construction and maintenance (i.e., building an overlay of nodes in a network) [31, 32, 33, 34, 35, 36]. Gossip protocols are distributed communication protocols that propagate messages in a way that is inspired by epidemics and human gossiping. Gossip protocols deal with a networked group of communicating agents, each holding a private information that represents agent's state, and aim at reaching a situation in which agents know each other state. Specifically, the information spreads within the network by pairwise communications among the set of agents. Pairwise communications occur periodically in rounds, such that gossip protocols usually work as the following:

- A node in the network selects a partner with which it is going to exchange some information.
- A data exchange process happens between these two nodes.
- Each node updates its own state using the received information.
- These steps are periodically repeated by the nodes in the network in order to disseminate information.

Accordingly, gossip protocols disseminate information without performing multicast communication, in which a node broadcasts a message to all the nodes in the network. With gossip, instead of contacting all nodes, each node sends the message to only a small subset of the nodes in the network. Thus, gossip protocols have many attractive properties particularly for large-scale distributed systems, such as speed, simplicity, robustness, and lack of a central control. Specifically, gossip protocols are scalable, thereby they are used in situations where it is not feasible to use traditional communication protocols due to both scale and dynamism of the underlying communication network. In practice, gossip protocols require $O(\log N)$ iterations, where N is number of nodes in the network [37, 38, 39] based on the assumptions that nodes are aware of the entire membership (i.e., all the nodes participating in the system), and they can choose their gossiping partner uniformly at random from the whole set of available nodes. Yet, this can limit the applicability of gossip protocols in large-scale networks due to the resources required to store and maintain the complete membership views at each node. Furthermore, the dynamism of the underlying communication network requires gossip protocols to be robust to node failure scenarios, such as node churn and catastrophic failure. Therefore, peer sampling services have been introduced to address these limitations as described next.

2.2.1 Peer Sampling Service

Peer sampling services (PSS) have been introduced by storing much smaller random subsets (i.e., local views) of the global membership at each node, and allowing nodes to choose gossip partners only from their local views [40, 41]. Gossip protocols are commonly used to implement a PSS. Particularly, each node stores a small subset of the participating nodes (i.e., a partial view of the network), then chooses a gossiping partner randomly from its local view, and exchanges its local view with this gossiping partner. By repeating this process periodically, each node is going to maintain a small, yet continuously changing set of other peers in the system. PSS protocols can be synchronous (all nodes are updated simultaneously) or asynchronous (some nodes are chosen uniformly at random to perform the exchange in the round, so some nodes may be updated more than once). However, PSS protocols assume that the underlying network allows direct and immediate communication between nodes, which is not the case in DOSNs, where nodes are restricted to communicate with their direct friends in the social graph, and their neighbors in the constructed P2P overlays. Authors in [42] propose a gossip-based PSS that is capable of running on top of restricted networks, such as fully decentralized DOSNs. Authors allow nodes to store routing tables in their local views. In these routing tables, the PSS can efficiently construct the paths towards the destination nodes on top of the underlying social network.

In our work, we employ a light-weight PSS that follows the same approach proposed in [42], for providing each node with a uniformly random subset of existing nodes in the system. Such service allows our systems to work without the need for any global knowledge of the whole network at any point.

2.3 Gossip Learning for Fully Distributed Data

Gossip protocols can provide a scalable and privacy preserving scheme for distributed machine learning. Ormándi *et al.* show that gossip protocols can be used to provide collaborative machine learning algorithms over fully distributed data in P2P networks, without collecting the data to a central location [43]. Specifically, nodes can reach a global model by creating their local models and exchanging them with a small subset of the other nodes in the network. Their algorithm relies on a PSS [44] that provides every node with uniform random samples of the nodes in the network. Algorithm 1 depicts the main operations performed by each node in the network as proposed in [43]. The skeleton of the proposed algorithm starts by *initModel* method that initializes the learning model according to the local data at each node. Afterwards, nodes exchange their local results by applying *sendModel* function. This function periodically selects a gossip partner (i.e., using *selectPartner* method) among the set of nodes that is provided using PSS. The process of exchange is repeated for some specific number of iterations that can be decided beforehand, or until some convergence criteria are met, and this can be associated with the objectives of the learning task/algorithm.

One of the most important features of gossip learning is that each node has more than one model that are available locally, which allows the nodes to perform their local decision at each time period. Furthermore, the nodes can apply many possible aggregation

Algorithm 1: Gossip Learning at node v_i

$m_i \leftarrow initModel\ ()$
Procedure *sendModel* (m_i)
  while *(condition)* **do**
    $wait\ (\Delta\ t)$
    $p \leftarrow selectPartner()$
    $exchange\ (m_i, p)$
  end
Procedure *onReceivedModel* (m)
  $updateModel\ (m_i, m)$

techniques to combine their locally generated models with the received ones, such as generating a new model by averaging the available models or using voting mechanism of the available models predictions. In most of the work presented in this thesis, we have been inspired by gossip learning, in order to ensure a scalable privacy preserving learning environment while minimizing the communication overhead. However, our algorithms extend the proposed mechanism and integrate PSS that complies with DOSN requirements.

In the next sections, we briefly describe the addressed research topics in the thesis, mainly community detection, spam detection, and identity validation.

2.4 Community Detection in Graphs

Real-networks can be analyzed using graph structures, such that they can be represented by a set of nodes (vertices) and a set of edges that represent the connections between a pair of nodes (neighbors). Community detection is a well-known and well-studied problem in graph theory [24]. Communities (also sometimes referred to as modules or clusters) are usually defined as groups of nodes that have better internal connectivity than external connectivity. For example, a community in social networks represents a group of individuals who interact with each other more frequently than with those outside this group. Furthermore, communities can be disjoint or overlapping. In case of overlapping communities, nodes are allowed to have multiple community memberships. For example, a person usually has connections to multiple groups or communities, such as family members, colleagues, friends, and co-workers. Communities in real-world networks vary in their characteristics, such as their internal cohesion and size. Therefore, many objective functions have been introduced to find such groups of nodes that are internally connected better than externally. However, like any clustering task that is known to be NP-hard, it is prohibitively expensive to search all such communities for the optimal value of internal and external connectivity. So as, different heuristics [45, 46] and approximation algorithms [47, 48] have been introduced to optimize one of the objective functions, while maintaining good performance in a reasonable time [49, 50].

Most of the existing community detection algorithms are centralized and they are designed to perform global optimization with a lot of assumptions on data availability. Particularly, heuristic mechanisms require the knowledge of the whole graph in order to optimize their associated objective functions. For example, conductance is a popular objective function that is used to evaluate the quality of communities by measuring the fraction of inter-cluster edges per each community to the intra-cluster edges [51]. Thus, calculating conductance requires global information about the whole graph, as well as the extracted communities. Besides the data constraint, existing community detection algorithms usually run with high computational cost as they require to load the whole graph into memory. On the other hand, community detection for DOSNs require localized techniques that operate using only the available partial knowledge of the social graph at each participating node.

Some approaches that provide local community detection approaches have been proposed to speed-up community detection for large-scale graphs and support execution in distributed environments, such as Label Propagation (LP) [52, 53, 54], and Random Walks (RW) [55, 56, 57]. These methods follow iterative approach and run with linear complexity in each iteration that equals to the number of edges in the graph. RW methods explore local structure around very few subset of nodes chosen as seed set, such that those nodes are labeled using their ground-truth community memberships. Accordingly, these methods require some known members as the prior for the semi-supervised clustering. Furthermore, RW methods suffer from the limitation of controlling their mixing time (i.e., the number of required iterations). This limitation is further discussed in Section 3.1. LP is an iterative algorithm that starts by initializing each node with a community identifier indicating its community membership, then propagates these identifiers in the network for some iterations. Nodes decide their community memberships based on their neighbors at each iteration. Specifically, each node joins the community to which majority of its neighbors belong. As labels propagate, densely connected nodes reach an agreement on a unique label, and these densely connected groups continue to expand forming communities. In comparison to other algorithms, LP is capable of extracting communities without requiring prior information about the network structure. However, LP techniques require communities size not be skewed. However, several studies show that distributions of community sizes seem to follow power laws in many cases [58, 59]. Therefore, there is a need to have localized community detection techniques that successfully extract communities with skewed size distribution.

DOSNs require a decentralized and scalable community detection mechanism. RW and LP employ a localized and scalable scheme that fit DOSN requirements, yet they require further fine-tuning to be able to detect communities accurately in heterogeneous cases. This thesis introduces a decentralized community detection approach (Stad) that performs optimization at community level, thereby tuning optimization parameters for each individual community. Experimental results with both real-world and synthetic datasets show that Stad outperforms the state-of-the-art techniques, not only in the community size scale issue but also by achieving higher accuracy that is twice the accuracy achieved by the state-of-the-art techniques.

2.5 Spam Detection in Social Networks

With the widespread usage of user generated content in Online Social Networks (OSNs), spam has increased and has become an effective vehicle for malware and illegal advertisement distribution. Spam not only pollutes the content contributed by normal users, resulting in bad user experiences, but also misleads and even traps legitimate users. Furthermore, OSNs have also led to new methods of delivering spam, such as spammy apps, social bots, and fake accounts. Spotting spammers is very challenging especially with the dynamic nature of social networks where activities and interactions among users evolve rapidly. Additionally, the problem becomes more challenging due to the huge amount of data shared by users. Therefore, the research community has produced a substantial number of mechanisms for automated spam detection based on binary classification mechanisms. The design of such spam detection mechanisms, in general, is guided by the behavior dissimilarity exhibited by legitimate users than spammers. The central premise as proved in the existing work is that spammer behavior appears anomalous relative to normal user behavior along some features that could be extracted from textual content (i.e., content-based features such as number of URLs, hashtags, and mentions used per post) and OSN friendship graph (i.e., graph-based features that are calculated from the friendship graph such as local clustering coefficient and betweenness centrality). However, all of the existing techniques rely on supervised binary classification methods [60, 61, 62, 63, 64].

Although the proposed binary classification methods succeed at detecting spam content, they implicitly require offline training with statistically sufficient and representative labeled training set of different user behaviors in order to achieve good detection coverage. This requirement itself is hard to satisfy, not to mention the difficulty of adapting to different behavior patterns that emerge in the future. Furthermore, the number of features required to discriminate spammers increases due to the diverse user activists in OSNs, the evolving spam patterns, as well as the limited the amount of labeled data. For example, Zhu et al. [63] use 1,680 different user activities in their supervised detection approach and Thomas et al. [64] train their URL spam filtering method using a sparse feature space with possible number of features up to 10^7. Additionally, binary classification methods result in false positive rate that could range between 5.7% and 0.8% [65, 62] resulting in some legitimate users are identified as spammers and get disconnected from the network. Particularly, derived from the remark that spammers hijack trending topics and include many URLs in their posts, content-based classification methods distinguish spammers by the extensive use of URLs, hashtags, and mentions. Consequently, legitimate users such as the official news channels that continuously broadcast posts with diverse topics containing URLs and hashtags are going to be classified as spammers.

DOSNs introduce new difficulties to obtain a comprehensive source of ground truth training samples with the plenty of measurements adopted in existing detection mechanisms. Therefore, DOSNs require unsupervised spam detection mechanism that successfully detects spam without using prior training with labeled samples. Furthermore, the deployed spam detection mechanism needs to adapt to the continuously evolving spam behavioral patterns. The work presented in this thesis (AdaGraph) addresses these limitations and provides a novel graph-clustering mechanism to detect spam. Additionally, our thesis empowers DOSNs with anti-spam framework (DLSAS) that provides defense mechanisms for DOSNs to prevent malicious nodes from participating in the system. Extensive experiments using Twitter datasets show that AdaGraph detects spam with accuracy 92.3%. Furthermore, the false positive rate of AdaGraph is less than 0.3% that is less than half of the rate achieved by the state-of-the-art approaches. Moreover, DLSAS ensures spam-detection integrity and reliability by detecting at least 94.7% of adversarial data manipulation that can be performed by malicious nodes.

2.6 Identity Validation Services for Social Networks

All of the social network sites employ a lightweight process for obtaining membership identities (i.e., confirming a valid email address) to facilitate their smooth joining and fast adoption. Moreover, when users create their profiles on these sites, they are given the complete freedom to fill up the records of their profiles without validating them. Consequently, such convenience increases the vulnerability of such networks to undergo security threats such as spam, malware, and phishing attacks [66, 67, 68].

One of the recently trending threats to OSNs is the spread of fake accounts that are seeking to get social [69, 70]. Fake accounts are nothing new to the online world in general and to social networks in particular. Despite all the efforts to aid the detection of fake accounts, they still make a considerable proportion of the active online population of today's major social networks. For instance, as of December 2015, Facebook has been reported to have 1.49 billion accounts out of which at least 83 million are known to be fake.[3] What is more dangerous than the existence of these fake accounts is their exploitation to build social trust; hence making honest targets more willing to trust dangerous content or putting the privacy of their information at risk [69, 70]. This social trustworthiness is mainly achieved by means of creating personal connections with honest users. Indeed, most of the techniques available for fake accounts detection rely on the premise that fake accounts exhibit tendencies of densely connected groups that are weakly connected to the rest of the social network, or outlying behavior that is skewed compared to common trends [71, 72]. As such, once a fake account succeeds in befriending honest users, its chances of getting detected would be considerably diminished. Moreover, the established connections may allow the fake account to inherit some of the trust according to the befriended honest account; thus give to the fake account more credibility resulting in higher chances of fooling other honest users [70]. This suggests that there may be a need for a mechanism that facilitates the validation of profiles in a social network to allow honest users to take better informed decisions before accepting a new connection in the network.

Several approaches have been proposed to address the problem of identity validation of users in social networks. Particularly, online identity validation targets the estimation of trustworthiness of a user profile in terms of linking this profile to a true social human identity. However, all of the existing approaches tend to compromise user privacy in their trial to achieve some security goals. For example, some of them identify users by utilizing their sensitive information such as geo-locations they usually visit and time-stamps of the

[3] https://zephoria.com/top-15-valuable-facebook-statistics/

information they share [73]. In [74], authors use typing patterns to identify users, whereas chatting patterns are exploited in [75]. Additionally, other validation approaches have suggested relying on human feedback. For example, in [76] the authors suggest evaluating an identity on a given network based on the feedback of her connections on another one. Generally, all of these techniques are derived from the incentive to validate online identities, yet they fail to limit the boundary of information to be used to fulfill their objective without violating users privacy or revealing their sensitive information to other entities who are not privileged to access it.

DOSNs require unsupervised and scalable identity validation schemes, such that user data needs to be processed locally at each participating node, without violating any privacy preservation constraints. This thesis introduces identity validation models (DIVa and CADIVa) that allow users to assign trust levels to whomever they interact with. Our identity validation models conceptualize user online identities by mining the correlations among user profile attributes and provide community-aware validation rules. Our models can be successfully applied in both OSN and DOSN frameworks. Experimental results show that reliance on revealing the highly expressed patterns inside communities resulted in extracting community-aware validation rules with average improvements up to 50% than the universal rules that only reveal the global patterns.

Chapter 3

Summary of Papers

> In all summaries, the problems seem simpler than they actually are.
>
> — Rollo May

This thesis is comprised of five papers. In this chapter, we describe the research questions and methodology of each paper, as well as we summarize the content and list the contributions of each paper.

3.1 Stad: Stateful Diffusion for Linear Time Community Detection

Communities in real-world networks vary in their characteristics, such as their internal cohesion and size. Despite a large variety of methods proposed to detect communities so far, most of the existing approaches fall into the category of global approaches. Specifically, these global approaches adapt their detection model focusing on approximating the global structure of the whole network, instead of performing approximation at the communities level. Furthermore, recent studies show that community size distributions in real-world networks follow power-law distributions [58, 59]. Accordingly, applying global approaches on such skewered cases results in low accuracy in terms of the extracted communities. Existing approaches are designed to tune their model parameters globally, thus either they fail to detect small communities if the parameters are tuned with respect to large communities, or the other way around, large communities might be over-partitioned and detected as multiple communities. Furthermore, the globally based detection algorithms are developed for centralized environments, as well as they usually run with high computational cost.

Research Problem and Methodology

DOSNs require a decentralized and scalable community detection mechanism. Random walks and diffusion-based techniques have been adopted to extract disjoint communities with low computational overhead. Specifically, these techniques can be implemented using node-centric programming model without requiring any global knowledge. The intuition behind random walks and diffusion is that once a random walker (or the diffusion process) enters a region, it tends to stay there for a long time, and movements between regions are relatively rare via one of the few outgoing edges. Specifically, diffusion captures how a flow starting from a specific node spreads on a graph, such that the spread of that flow mixes fast within well connected regions, and slowly in less connected ones. Thus, this phenomenon can be used for community detection by capturing the boundaries of well connected regions. Mixing time is commonly used to indicate the convergence rate of the random walker, in our case we relate the mixing time to the state when a random walker is stabilized inside a community. Yet, mixing time needs be controlled, as very long random walk reaches a stationary distribution which is not expected to indicate the extraction of well connected clusters.

In this paper, we present Stad as a decentralized and scalable diffusion-based community detection approach. Stad introduces the concept of stateful diffusion and boosts diffusion with adaptive speed functions that control the mixing time for each individual community. Most importantly, Stad addresses the limitations of existing approaches that are related to extracting communities having heterogeneous community size distribution without optimizing the model's parameters globally.

Our Contributions

- **Algorithm**: Stad provides a novel diffusion-based community detection algorithm that requires no prior knowledge about the ground-truth community memberships. Particularly, Stad employs a vertex-centric approach which is fully decentralized and highly scalable, and requires no global knowledge. Furthermore, Stad extracts disjoint as well as overlapping communities.

- **Optimization**: Stad performs optimization at community level, such that Stad introduces the concept of stateful diffusion which controls the mixing time of the diffusion process according to the size of each extracted community. Moreover, Stad treats the random walkers independently and controls the diffusion process at each node in the graph.

3.2 AdaGraph: Adaptive Graph-based Algorithms for Spam Detection in Social Networks

With the widespread usage of user generated content in social networks, spam has increased and has become an effective vehicle for malware and illegal advertisement distribution. Spam not only pollutes the content contributed by normal users, resulting in bad user experiences, but also misleads and even traps legitimate users. Spotting spammers is very challenging especially with the dynamic nature of social networks where activities and interactions among users evolve rapidly. Additionally, the problem becomes more challenging due to the huge amount of data shared by users. Therefore, the research community has produced a substantial number of mechanisms for automated spam detection using classification mechanisms. The first family of spam detection mechanisms includes techniques using blacklists to identify URL on OSNs websites directing to spam content [64, 77]. However, URL blacklisting has several practical challenges. First, those blacklists are publicly available, hence spammers can evade them by changing their domain names or hiding them behind some redirecting pages. Second, URL blacklisting becomes ineffective with the spread usage of URL shortening services such as bit.ly and t.co. Furthermore, a rich corpus of research work lies in adopting supervised machine learning based methods using hybrid features extracted from textual content and social friendship graph [61, 77]. The design of such spam detection mechanisms, in general, is guided by the behavior dissimilarity exhibited by legitimate users than spammers.

Research Problem and Methodology

Although the proposed binary classification methods succeed at detecting spam content, they implicitly require offline training with statistically sufficient and representative labeled training set of different user behaviors in order to achieve good detection coverage. This requirement itself is hard to satisfy, not to mention the difficulty of adapting to different behavior patterns that emerge in the future. Furthermore, the number of features required to discriminate spammers increases due to the diverse user activists in social networks, the evolving spam patterns, as well as the limited the amount of labeled data. In addition to that, it is difficult to obtain a comprehensive source of ground truth for measurement in decentralized environments like DOSNs.

This paper provides unsupervised graph-based clustering technique for spam detection. The essence of AdaGraph is to construct a user similarity graph that encodes within its topology a holistic view of all behavioral interactions and patterns of users in the social network. Afterwards, AdaGraph performs graph clustering by applying community detection on top of the newly created graph. Spam detection using graph-based clustering not only diminishes the training cost, but also achieves low false positive rate. Graph-based clustering provides meaningful insights into the existing behavioral patterns, therefore, categorizes the existing patterns into more homogeneous and accurate clusters than binary classification. More importantly, AdaGraph provides a decentralized spam detection approach, such that it allows every node to independently process its data and only communicate with its direct neighbors.

Our Contributions

- **Detection Algorithm**: AdaGraph presents unsupervised spam detection approach that requires no a priori labeling while maintaining low false positive rate. Furthermore, AdaGraph proposes a novel graph-based spam detection technique that detects spam using graph clustering on top of a constructed user similarity graph which encodes user behavioral patterns within its topology.

- **Optimization**: AdaGraph introduces adaptive algorithms that enable similarity-based community detection to evolve with respect to the behavioral changes of the users in purely decentralized manner.

3.3 DLSAS: Distributed Large-Scale Anti-Spam Framework for Decentralized Online Social Networks

Although, our graph-based approach achieves better results compared to the current state-of-the-art in spam detection, in its current form it is not possible to adopt for DOSNs due to several key challenges. Specifically, the incentive for defeating spam detection in decentralized environments increases due to distributed nature of the system and non-avoidable cooperation among participating nodes. For example, malicious parties can destroy communication pathways, prevent the system from stabilizing, equivocate by giving different information to different nodes, and provide false provenance information. Therefore, our graph-based approach requires a defense mechanism to detect any data manipulation during the phases of similarity graph creation and community detection.

Research Problem and Methodology

This paper addresses these issues and preserves the core of spam detection (i.e., similarity graph creation and community detection) from manipulation under the existence of adversarial nodes that deviate from the designed workflow protocol. Secure cooperation can be achieved when misbehaving or deviating nodes are detected and disconnected from the system. Thus, the exchanged data among nodes requires verification mechanism that is able to find a proof that a node deviated from the prescribed protocol and participated in adversarial activities. Accordingly, in DLSAS every node is required to keep a communication log that records all communication events. Consequently, any misbehaving node can be detected by comparing its manipulated log with the logs of the other nodes involved in communication. However, malicious nodes can collude with each other to evade the validation and detection mechanism. Therefore, every validation task has to be assigned randomly in order to prevent adversaries from inferring any details about the participating nodes. Accordingly, DLSAS creates on-the-fly random validation overlay to validate the data reported by the participating nodes.

Our Contributions

- **Validation Scheme**: DLSAS proposes a uniformly random validation mechanism that is capable of detecting and disconnecting misbehaving nodes from the system. Specifically, the proposed validation mechanism is fully decentralized and is capable of maintaining DOSNs robustness and integrity against adversarial attacks.

3.4 DIVa: Decentralized Identity Validation for Social Networks

Users can join social networks frameworks (either OSNs or DOSNs) easily by confirming a valid email address. However, this lightweight process for obtaining identities underpins the vulnerability of such networks to undergo different attacks. Moreover, users are given the complete freedom to fill up the records of their profiles without validating them. Thus, malicious users can provide misleading information or they can easily claim to be someone else [78, 79], making the networking environment unsafe. Into a step towards the automation of identity validation, [80] suggests validating profiles based on correlations between their attributes. The authors demonstrate that existing correlations between profile attributes can be reliably used to estimate a profile's identity trustworthiness from its attribute values only. For example, a user who specifies her occupation as a *computer science student* is expected to be attending a *university* that offers such major. More precisely, the authors suggest a two phase system. In the first one, they exploit a supervised crowd-based learning strategy to extract profile attribute correlations that are meaningful from an identity validation perspective. They do this by gathering human feedback from a group of trusted users on a centralized profiles training dataset. Once these correlations are identified, they are used in the second phase, that also engages users' feedback, to estimate the identity trustworthiness of a target profile.

Research Problem and Methodology

Although profile exploitation approach is very promising, there are several challenges that hinder its success in terms of practicality. First, and given the number of users in current OSNs, it is not realistically scalable to rely on trusted users feedback for the learning of attribute correlations. Besides, collecting the data centrally violates DOSNs privacy constraints as it is often not allowed to move users' data outside their direct connections. Furthermore, it is hard to identify who the trusted users are to ensure the accuracy of learning attribute correlations. Finally, relying on users' feedback introduces privacy risks and is not applicable for DOSN models. Additionally, decoupling profiles data from the semantic of users' connectivity during attribute correlations extraction degrade the performance as it might result in infirm and/or prejudiced validation. Specifically, social networks exhibit homophily by which users topologically cluster into communities.

This paper conceptualizes user online identities by mining the correlations among user profile attributes not from user population as a whole, but from individual communities, where the correlations are more pronounced. The key of DIVa's performance lies in its ability to leverage on homogeneity among users' profiles inside every existing community instead of human-feedback based learning. This allows DIVa to extract more meaningful profile attribute correlations within communities than any of the existing methods. DIVa achieves this in a three phase process that starts with each node learning the collection of its local correlated attribute sets by exploiting association rule mining over the profiles of its direct friends only. In the second phase, a community detection mechanism is deployed to define the communities existing in the network. Thereafter, every node, knowing the communities to which it belongs, communicates its learned collection of attribute correlations to the super nodes of its membership communities. These super nodes, referred to as diva nodes, are unique in each community and are responsible of receiving all attribute correlation collections from all the nodes in their community and aggregating them to generate the community level correlated attribute sets.

Our Contributions

- **Validation Scheme**: DIVa provides a novel unsupervised and incremental learning scheme that leverages on mining techniques to find correlations among user profile attributes.

- **Scalability**: DIVa extends the ensemble learning paradigm in distributed machine learning and works on fully distributed datasets without collecting the data into one central location.

- **Privacy**: DIVa preserves user privacy, such that nodes have access only and solely to the local data, that is available given particular privacy settings. In DIVa, nodes extract their local correlations using profiles of their direct neighbors, then exchange only the set of generated models within their communities to agree on the community correlations.

- **Performance**: DIVa's key performance relies on providing community-aware validation. DIVa allows users to have multiple community memberships, thus DIVa succeeds in identifying meaningful rules for each community.

3.5 CADIVa: Cooperative and Adaptive Decentralized Identity Validation Model for Social Networks

DIVa demonstrated good results in meeting the goal of designing an identity validation model for social network frameworks that uses minimal information (i.e., profile information only); however, DIVa does not provide a fully decentralized solution. In fact, it assumes the availability of some super nodes (i.e., diva nodes) that are exploited as central hubs within each community and are used to aggregate the final community-based profile attribute correlations. These super nodes might constitute single points of failure or performance bottlenecks in the system as the process depends on their availability and on their ability to perform the tasks entrusted to them. Moreover, the assumption of super nodes does not fully align with the fully decentralized spirit of DOSNs. In addition to that, DIVa bases on static assumptions across all communities for the threshold values adopted to learn significant correlations within each community. That is, all communities adopt the same threshold value for the learning of a valid correlation between profile attributes of their members, ignoring the specific characteristics of every community such as size, homogeneity, etc.

Research Problem and Methodology

The objective of this paper is to address DIVa's limitations. In this paper, we present CADIVa that is fully decentralized and adaptive. CADIVa exploits gossip learning to provide fully decentralized and cooperative learning, not only to preserve users privacy, but also to increase the system reliability and to make it resilient to mono-failure. Furthermore, CADIVa tunes the statistically significant threshold for selecting profile attribute correlations according to the number of nodes belonging to each community. Adaptive thresholds increase the freedom of each community to have the value that reflects the level of homogeneity among its constituent members. Furthermore, CADIVa continuously updates communities validation rules while new nodes being added to the communities. The first part of CADIVa's adaptability lies in computing different threshold values according to the statistical strength of attribute pairs frequency inside every community independently. Secondly, CADIVa monitors the topological changes in the communities after adding the new nodes/edges. Afterwards, CADIVa re-performs the rule extraction in the regions where communities topologically change.

Our Contributions

- **Identity Validation Model** : CADIVa provides cooperative, massively parallel and reliable identity validation model that preserves users privacy and operates without super nodes support, hence it suitably fits DOSNs.
- **Adaptability**: CADIVa is capable of tuning the model parameters to reflect the existing homophily level inside every community. Furthermore, CADIVa monitors the evolving changes in the underlying social graph and updates communities validation rules.

Chapter 4

Conclusions

> Someone is sitting in the shade today because someone planted a tree a long time ago.
>
> — Warren Buffett

With no doubts, current social network frameworks entertain people, allow them to communicate with virtually anyone in the world instantaneously. However, social network users have no clear idea about who accesses their data when it is uploaded to provider's datacenter. Seeking information privacy is not about secrecy, yet it is about transparency and choice. Users should be informed about the handling of their personal information, and they can freely decide how their data can be shared, processed and utilized. DOSNs bring a promise to bridge this gap and allow users to enjoy the different social network services without losing the ownership of their data. Decentralized data processing is the most promising way to provide privacy preserving data analytics. Our work is a step in that direction.

Our graph-based analytics go beyond current data mining methods that extract patterns from textual data only, and leverage interlinked nature of social network that connects the shared content with the underlying social network. Therefore, in this thesis, we propose the concept of community-aware learning that extracts patterns for each individual community where the correlations are more pronounced. To enable community-aware learning, we propose *Stad* as a decentralized community detection approach that extracts communities more adaptively compared to the existing solutions. We showed that *Stad* manages to extract communities in real-world networks that follow skewed community size distributions, while avoiding size limitation issues exhibited in the existing state-of-the-art methods. In particular, *Stad* succeeds at preserving the small communities from being merged with bigger ones, as well as avoiding over-partitioning the big communities into multiple communities. Accordingly, building any community-aware learning task on top of *Stad* gives the opportunity to every individual community to extract its own behavioral patterns.

The proposed graph-based algorithms and methods in this thesis allow generation of efficient ML models using user local data in fully decentralized, iterative, massively parallel, and highly scalable manner. We addressed the challenge of providing decentralized spam detection, and presented a framework that combines unsupervised spam detection (*AdaGraph*) with misbehavior detection (*DLSAS*) mechanisms. *AdaGraph* employs graph-based spam detection technique that requires no prior training. *AdaGraph* extends the friendship graph and creates a similarity graph in which similarly behaving users are linked. We showed that *AdaGraph* extracts more homogeneous clusters, hence achieves low false positive rate compared to the existing classification approaches. Furthermore, *AdaGraph* employs different adaptive mechanisms that allow similarity graph to evolve with user behavioral updates. In order for *AdaGraph* to be effectively applied in DOSNs, we presented *DLSAS* that preserves the core of spam detection from any manipulations. *DLSAS* uses a gossip-based peer sampling service to construct a validation overlay that detects and disconnects misbehaving nodes from the system. Furthermore, our solutions for identity validation (*DIVa* and *CADIVa*) provide unsupervised and incrementally updated models that leverage on mining the correlations among user profile attributes. The key performance of our solutions relies on extracting community-aware validation rules in reliable, scalable, and privacy preserving manner. We showed that our models can provide adaptive validation rules for each individual community better than centralized and global approaches.

In summary, our graph-based analytics bring complex analytic services, which require decentralized iterative computation over distributed data, to the domain of DOSNs. Furthermore, the proposed algorithms in this thesis can be further combined to provide more services for DOSNs rather than the ones discussed in this thesis. For example, *DLSAS*'s core can be easily adapted to any collaborative learning task for DOSNs, as the essence of *DLSAS* is protecting the workflow protocol form adversarial manipulations. Additionally, similarity graph construction technique proposed in AdaGraph can be combined with the gossip learning approach used in CADIVa to provide a recommendation system for DOSNs.

Bibliography

[1] A. Datta, S. Buchegger, L.-H. Vu, T. Strufe, and K. Rzadca, "Decentralized online social networks," in *Handbook of Social Network Technologies and Applications*. Springer, 2010, pp. 349–378.

[2] G. McNeill, J. Bright, and S. A. Hale, "Estimating local commuting patterns from geolocated twitter data," *EPJ Data Science*, vol. 6, no. 1, p. 24, Oct 2017. [Online]. Available: https://doi.org/10.1140/epjds/s13688-017-0120-x

[3] Y. Wang and M. Kosinski, "Deep neural networks are more accurate than humans at detecting sexual orientation from facial images," *Journal of Personality and Social Psychology*, 2017.

[4] D. Koll, J. Li, and X. Fu, "Soup: an online social network by the people, for the people," in *SIGCOMM'14*. ACM, 2014, pp. 193–204.

[5] S. Nilizadeh, S. Jahid, P. Mittal, N. Borisov, and A. Kapadia, "Cachet: a decentralized architecture for privacy preserving social networking with caching," in *Proceedings of the 8th international conference on Emerging networking experiments and technologies*. ACM, 2012, pp. 337–348.

[6] P. Kapanipathi, J. Anaya, A. Sheth, B. Slatkin, and A. Passant, "Privacy-aware and scalable content dissemination in distributed social networks," *The Semantic Web–ISWC 2011*, pp. 157–172, 2011.

[7] A. Bielenberg, L. Helm, A. Gentilucci, D. Stefanescu, and H. Zhang, "The growth of diaspora-a decentralized online social network in the wild," in *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on*. IEEE, 2012, pp. 13–18.

[8] R. Lu, H. Zhu, X. Liu, J. K. Liu, and J. Shao, "Toward efficient and privacy-preserving computing in big data era," *IEEE Network*, vol. 28, no. 4, pp. 46–50, 2014.

[9] M. Bojarski, A. Choromanska, K. Choromanski, and Y. LeCun, "Differentially-and non-differentially-private random decision trees," *arXiv preprint arXiv:1410.6973*, 2014.

[10] J. Vaidya, B. Shafiq, W. Fan, D. Mehmood, and D. Lorenzi, “A random decision tree framework for privacy-preserving data mining,” *IEEE transactions on dependable and secure computing*, vol. 11, no. 5, pp. 399–411, 2014.

[11] W. Dai, S. Wang, H. Xiong, and X. Jiang, “Privacy preserving federated big data analysis,” in *Guide to Big Data Applications*. Springer, 2018, pp. 49–82.

[12] M. A. U. Nasir, S. Girdzijauskas, and N. Kourtellis, “Socially-aware distributed hash tables for decentralized online social networks,” in *Peer-to-Peer Computing (P2P), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1–10.

[13] S. Antaris, D. Stasi, M. Högqvist, G. Pallis, and M. Dikaiakos, “A socio-aware decentralized topology construction protocol,” in *Hot Topics in Web Systems and Technologies (HotWeb), 2015 Third IEEE Workshop on*. IEEE, 2015, pp. 91–96.

[14] S. G. G. P. M. D. Nuno Apolonia, Stefanos Antaris, “Select: A distributed publish/subscribe notification system for online social networks,” in *Proceedings of the 32nd International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE, 2018.

[15] B. Greschbach, G. Kreitz, and S. Buchegger, “User search with knowledge thresholds in decentralized online social networks,” in *IFIP PrimeLife International Summer School on Privacy and Identity Management for Life*. Springer, 2013, pp. 188–202.

[16] R. Sharma and A. Datta, “Supernova: Super-peers based architecture for decentralized online social networks,” in *Communication Systems and Networks (COMSNETS), 2012 Fourth International Conference on*. IEEE, 2012, pp. 1–10.

[17] S. Jahid, S. Nilizadeh, P. Mittal, N. Borisov, and A. Kapadia, “Decent: A decentralized architecture for enforcing privacy in online social networks,” in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on*. IEEE, 2012, pp. 326–332.

[18] L. Bahri, B. Carminati, and E. Ferrari, “Cards-collaborative audit and report data sharing for a-posteriori access control in dosns,” in *Collaboration and Internet Computing (CIC), 2015 IEEE Conference on*. IEEE, 2015, pp. 36–45.

[19] L. Bahri, B. Carminati, E. Ferrari, and W. Lucia, “Lamp-label-based access-control for more privacy in online social networks,” in *IFIP International Conference on Information Security Theory and Practice*. Springer, 2016, pp. 171–186.

[20] O. Bodriagov, G. Kreitz, and S. Buchegger, “Access control in decentralized online social networks: Applying a policy-hiding cryptographic scheme and evaluating its performance,” in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2014 IEEE International Conference on*. IEEE, 2014, pp. 622–628.

[21] G. Rodriguez-Cano, B. Greschbach, and S. Buchegger, "Event invitations in privacy-preserving dosns," in *IFIP International Summer School on Privacy and Identity Management*. Springer, 2014, pp. 185–200.

[22] R. Sharma and A. Datta, "Godisco++: A gossip algorithm for information dissemination in multi-dimensional community networks," *Pervasive and Mobile Computing*, vol. 9, no. 2, pp. 324–335, 2013.

[23] S. Girdzijauskas, "Designing peer-to-peer overlays: a small-world perspective," *EPFL thesis no. 4327, advisor: Karl Aberer*, vol. 154, 2009.

[24] S. Fortunato and D. Hric, "Community detection in networks: A user guide," *Physics Reports*, vol. 659, pp. 1–44, 2016.

[25] S. Schelter, C. Boden, M. Schenck, A. Alexandrov, and V. Markl, "Distributed matrix factorization with mapreduce using a series of broadcast-joins," in *Proceedings of the 7th ACM Conference on Recommender Systems*. ACM, 2013, pp. 281–284.

[26] R. Xu, S. Wang, X. Zheng, and Y. Chen, "Distributed collaborative filtering with singular ratings for large scale recommendation," *Journal of Systems and Software*, vol. 95, pp. 231–241, 2014.

[27] A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry, "Epidemic algorithms for replicated database maintenance," in *Proceedings of the sixth annual ACM Symposium on Principles of distributed computing*. ACM, 1987, pp. 1–12.

[28] M. Jelasity, W. Kowalczyk, and M. v. Steen, "Newscast computing," 2012.

[29] A. M. Ricciardi and K. P. Birman, "Using process groups to implement failure detection in asynchronous environments," in *Proceedings of the tenth annual ACM symposium on Principles of distributed computing*. ACM, 1991, pp. 341–353.

[30] R. Van Renesse, Y. Minsky, and M. Hayden, "A gossip-style failure detection service," in *MiddlewareâĂŹ98*. Springer, 1998, pp. 55–70.

[31] M. Jelasity, R. Guerraoui, A.-M. Kermarrec, and M. Van Steen, "The peer sampling service: Experimental evaluation of unstructured gossip-based implementations," in *Proceedings of the 5th ACM/IFIP/USENIX international conference on Middleware*. Springer-Verlag New York, Inc., 2004, pp. 79–98.

[32] A. J. Ganesh, A.-M. Kermarrec, and L. Massoulié, "Peer-to-peer membership management for gossip-based protocols," *IEEE transactions on computers*, vol. 52, no. 2, pp. 139–149, 2003.

[33] M. Jelasity and O. Babaoglu, "T-man: Gossip-based overlay topology management," *Engineering Self-Organising Systems*, vol. 3910, pp. 1–15, 2005.

[34] F. Rahimian, S. Girdzijauskas, A. H. Payberah, and S. Haridi, "Vitis: A gossip-based hybrid overlay for internet-scale publish/subscribe enabling rendezvous routing in unstructured overlay networks," in *Parallel & Distributed Processing Symposium (IPDPS), 2011 IEEE International*. IEEE, 2011, pp. 746–757.

[35] M. A. U. Nasir, F. Rahimian, and S. Girdzijauskas, "Gossip-based partitioning and replication for online social networks," in *Advances in Social Networks Analysis and Mining (ASONAM), 2014 IEEE/ACM International Conference on*. IEEE, 2014, pp. 33–42.

[36] F. Rahimian, T. L. N. Huu, and S. Girdzijauskas, "Locality-awareness in a peer-to-peer publish/subscribe network." in *DAIS*. Springer, 2012, pp. 45–58.

[37] A. Montresor, "Gossip and epidemic protocols," *Wiley Encyclopedia of Electrical and Electronics Engineering*.

[38] B. Pittel, "On spreading a rumor," *SIAM Journal on Applied Mathematics*, vol. 47, no. 1, pp. 213–223, 1987.

[39] R. Karp, C. Schindelhauer, S. Shenker, and B. Vocking, "Randomized rumor spreading," in *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*. IEEE, 2000, pp. 565–574.

[40] S. Voulgaris, D. Gavidia, and M. Van Steen, "Cyclon: Inexpensive membership management for unstructured p2p overlays," *Journal of Network and Systems Management*, vol. 13, no. 2, pp. 197–217, 2005.

[41] N. Tölgyesi and M. Jelasity, "Adaptive peer sampling with newscast," in *European Conference on Parallel Processing*. Springer, 2009, pp. 523–534.

[42] M. Khelghatdoust and S. Girdzijauskas, "Short: Gossip-based sampling in social overlays," in *Networked Systems*. Springer, 2014, pp. 335–340.

[43] R. Ormándi, I. Hegedűs, and M. Jelasity, "Gossip learning with linear models on fully distributed data," *Concurrency and Computation: Practice and Experience*, vol. 25, no. 4, pp. 556–571, 2013.

[44] M. Jelasity, S. Voulgaris, R. Guerraoui, A.-M. Kermarrec, and M. Van Steen, "Gossip-based peer sampling," *ACM Transactions on Computer Systems (TOCS)*, vol. 25, no. 3, p. 8, 2007.

[45] G. Karypis and V. Kumar, "A fast and high quality multilevel scheme for partitioning irregular graphs," *SIAM Journal on scientific Computing*, vol. 20, no. 1, pp. 359–392, 1998.

[46] I. S. Dhillon, Y. Guan, and B. Kulis, "Weighted graph cuts without eigenvectors a multilevel approach," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 11, 2007.

[47] R. Andersen, F. Chung, and K. Lang, "Local graph partitioning using pagerank vectors," in *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on*. IEEE, 2006, pp. 475–486.

[48] K. Lang and S. Rao, "A flow-based method for improving the expansion or conductance of graph cuts," in *International Conference on Integer Programming and Combinatorial Optimization*. Springer, 2004, pp. 325–337.

[49] M. Chen, K. Kuzmin, and B. K. Szymanski, "Community detection via maximization of modularity and its variants," *IEEE Transactions on Computational Social Systems*, vol. 1, no. 1, pp. 46–65, 2014.

[50] J. Leskovec, K. J. Lang, and M. Mahoney, "Empirical comparison of algorithms for network community detection," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 631–640.

[51] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[52] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical review E*, vol. 76, no. 3, p. 036106, 2007.

[53] F. Rahimian, S. Girdzijauskas, and S. Haridi, "Parallel community detection for cross-document coreference," in *Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 02*. IEEE Computer Society, 2014, pp. 46–53.

[54] A. Guerrieri, F. Rahimian, S. Girdzijauskas, and A. Montresor, "Tovel: Distributed graph clustering for word sense disambiguation," in *Data Mining Workshops (ICDMW), 2016 IEEE 16th International Conference on*. IEEE, 2016, pp. 623–630.

[55] Y. Li, K. He, D. Bindel, and J. E. Hopcroft, "Uncovering the small community structure in large networks: A local spectral approach," in *Proceedings of the 24th international conference on world wide web*. International World Wide Web Conferences Steering Committee, 2015, pp. 658–668.

[56] M. W. Mahoney, L. Orecchia, and N. K. Vishnoi, "A local spectral method for graphs: with applications to improving graph partitions and exploring data graphs locally," *arXiv preprint arXiv:0912.0681*, 2009.

[57] K. He, Y. Sun, D. Bindel, J. Hopcroft, and Y. Li, "Detecting overlapping communities from local spectral subspaces," in *Data Mining (ICDM), 2015 IEEE International Conference on*. IEEE, 2015, pp. 769–774.

[58] M. E. Newman, "Fast algorithm for detecting community structure in networks," *Physical review E*, vol. 69, no. 6, p. 066133, 2004.

[59] A. Clauset, M. E. Newman, and C. Moore, “Finding community structure in very large networks,” *Physical review E*, vol. 70, no. 6, p. 066111, 2004.

[60] H. Gao, Y. Chen, K. Lee, D. Palsetia, and A. N. Choudhary, “Towards online spam filtering in social networks.” in *NDSS*, 2012.

[61] C. Yang, R. Harkreader, and G. Gu, “Empirical evaluation and new design for fighting evolving twitter spammers,” *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 8, pp. 1280–1293, 2013.

[62] A. A. Amleshwaram, N. Reddy, S. Yadav, G. Gu, and C. Yang, “Cats: Characterizing automation of twitter spammers,” in *COMSNET’13*. IEEE, 2013, pp. 1–10.

[63] Y. Zhu, X. Wang, E. Zhong, N. N. Liu, H. Li, and Q. Yang, “Discovering spammers in social networks.” in *AAAI’12*, 2012.

[64] K. Thomas, C. Grier, J. Ma, V. Paxson, and D. Song, “Design and evaluation of a real-time url spam filtering service,” in *Security and Privacy (SP), 2011 IEEE Symposium on*. IEEE, 2011, pp. 447–462.

[65] J. Martinez-Romo and L. Araujo, “Detecting malicious tweets in trending topics using a statistical analysis of language,” *Expert Systems with Applications*, vol. 40, no. 8, pp. 2992–3000, 2013.

[66] M. Huber, M. Mulazzani, E. Weippl, G. Kitzler, and S. Goluch, “Friend-in-the-middle attacks: Exploiting social networking sites for spam,” *Internet Computing*, 2011.

[67] W. Luo, J. Liu, J. Liu, and C. Fan, “An analysis of security in social networks,” in *Dependable, Autonomic and Secure Computing, 2009. DASC’09. Eighth IEEE International Conference on*. IEEE, 2009, pp. 648–651.

[68] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer, “Social phishing,” *Communications of the ACM*, 2007.

[69] R. M. Robinson, “Social engineering attackers deploy fake social media profiles,” December 2015, security Intelligence. [Online]. Available: https://securityintelligence.com/social-engineering-attackers-deploy-fake-social-media-profiles/

[70] G. Stringhini, “Stepping up the cybersecurity game: Protecting online services from malicious activity,” Ph.D. dissertation, UNIVERSITY OF CALIFORNIA Santa Barbara, 2014.

[71] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman, “Sybilguard: defending against sybil attacks via social networks,” *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 4, pp. 267–278, 2006.

[72] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao, “Sybillimit: A near-optimal social network defense against sybil attacks,” in *Security and Privacy*. IEEE, 2008.

[73] O. Goga, H. Lei, S. H. K. Parthasarathi, G. Friedland, R. Sommer, and R. Teixeira, "Exploiting innocuous activity for correlating users across sites," in *WWW'13*. International World Wide Web Conferences Steering Committee, 2013.

[74] P. Chairunnanda, N. Pham, and U. Hengartner, "Privacy: Gone with the typing! identifying web users by their typing patterns," in *PASSAT'11*. IEEE, 2011.

[75] G. Roffo, C. Segalin, A. Vinciarelli, V. Murino, and M. Cristani, "Reading between the turns: Statistical modeling for identity recognition and verification in chats," in *AVSS'13*. IEEE, 2013, pp. 99–104.

[76] M. Sirivianos, K. Kim, J. W. Gan, and X. Yang, "Assessing the veracity of identity assertions via osns," in *COMSNETS'12*. IEEE, 2012.

[77] K. Levchenko, A. Pitsillidis, N. Chachra, B. Enright, M. Félegyházi, C. Grier, T. Halvorson, C. Kanich, C. Kreibich, H. Liu *et al.*, "Click trajectories: End-to-end analysis of the spam value chain," in *Security and Privacy (SP), 2011 IEEE Symposium on*. IEEE, 2011, pp. 431–446.

[78] A. C. Squicciarini, C. Griffin, and S. Sundareswaran, "Towards a game theoretical model for identity validation in social network sites," in *PASSAT'11*. IEEE, 2011, pp. 1081–1088.

[79] H. Krasnova, O. Günther, S. Spiekermann, and K. Koroleva, "Privacy concerns and identity in online social networks," *Identity in the Information Society*, 2009.

[80] L. Bahri, B. Carminati, and E. Ferrari, "Community-based identity validation on online social networks," in *ICDCS'14*. IEEE, 2014, pp. 21–30.