An Initial Homophily Indicator to Reinforce Context-Aware Semantic Computing

Alejandro Rivero-Rodriquez, Paolo Pileggi and Ossi Nykänen
Department of Mathematics
Tampere University of Technology
Tampere, Finland
e-mail: {alejandro.rivero, paolo.pileggi, ossi.nykanen}@tut.fi

Abstract—**The vast increase of personal sensor information is driving the rise in popularity of context-aware applications. Users crave and very often expect tailored services that are based on the users' context or personal preferences. The users themselves, using forms, often provide such information. An inference solution typically addresses this problem. In this paper, we present and show by way of a real-world example, the first step towards incorporating information of the user's social networking behavior in the inference task. We define an initial indicator of a particular social phenomenon, called Homophily, and describe how the indicator measures the presence of homophily at certain moments, also capturing the degree to which it is present. Different from existing indicators, ours lends itself to indicating the presence of homophily in a way that is easier to comprehend, so that it may be easily integrated into and reinforce context-aware semantic computing.**

Keywords-Social Network Analysis; Homophily; Context-aware Computing.

I. INTRODUCTION

Computing devices perform many operations automatically and faster than humans do. However, unlike computers, humans adapt more easily to new situations that may arise. One natural way to improve computational intelligence is to enable computers to understand context [1]. This has been broadly studied in the field of context awareness [2].

The relevance of Smartphones has increased tremendously in recent years. On one hand, technically they have advanced significantly, and nowadays they are considered to be small computers. On the other hand, the percentage of the population who owns a Smartphone has increased from as little as 1% in 2006 to 22% in 2013 [3]. In some countries, people own on average more than one mobile device, and use them to communicate with friends, family, colleagues, and even businesses and governments, in social networks.

Probably the most revolutionary aspects of modern Smartphones are the inclusion of sensors, and the possibility for third-parties to easily develop a variety of applications. By combining both these aspects, the context-aware application was born, where the user is provided a service, depending on his or her context, i.e., any information related to the user, such as its location.

The number of context-aware services has increased significantly, which include social networks of a diverse nature like *Facebook* and *Foursquare*; personal assistants like *Google Now*; and movement tracking applications like *Moves* or *RunKeeper*.

These applications offer services based on location, called Location Based Services [4], in other words, on the data obtained using the sensors built into the users' devices.

Social Network Analysis (SNA) can provide relevant information about the users that, in turn, can be exploited to develop better context-aware applications.

In particular, *Homophily* is a well-known occurring phenomenon in social networks. Users with similar contexts tend to connect at a higher rate [5,6]. For example, *CICSyN* organizers are highly connected to each other. Therefore, we would assume that a *CICSyN* organizer is more likely to be connected to another organizer of the conference than to an external person.

Using the concept of homophily, contextual cues, called *attributes*, can be transferred within communities that form a highly connected group of users [7]. Then, continuing with the example, we could infer that one is a *CICSyN* organizer if the person has very strong relationships with many of the event organizers.

In this paper, we propose a normalized homophily indicator that is compact and relatively easy to understand, that benefits context inference. We experiment with real-world data, comparing our results to those of a similar indicator that exists.

In the sequel, we delve into context management, mentioning relevant and proposed architectures, and describe how SNA plays an essential role in context management and context-aware computing, in general. In Section III, we present an indicator of homophily that captures the degree to which homophily occurs in the social network. We apply our indicator to analyze real-world data and compare it to another indicator in Section IV. Finally, in Section V, we conclude by highlighting several aspects of the future work needed to result in methods derived by using or incorporating our indicator, when we have shown to be easier for the application developer to understand, and at least as lightweight as existing indicators.

II. BACKGROUND

A. Context-Aware systems and architectures

The term context-aware (computing) appeared first in the early 1990s, with the beginning of context-aware system research [8]. *Context*, also referred to as contextual information, refers to any information that can be used to characterize the situation of an entity, where an entity can be a person, place, or physical or computational object [9].

Since then, a significant amount of effort was invested into context-aware computing [8]. These systems capture many types of context in addition to time and position, such as places, things, commitments and user preferences [10]. The main components of a context-aware system include context providers and context-aware services [11].

Several architectures and frameworks have been used to manage and reason about user context, such as the well-known *Context Managing Framework*, *Context Broker Architecture* or *Service-Oriented Context-Aware Middleware* [12]. In particular, we draw the readers' attention to our software service, called the *Context Engine* (CE) [13].

The CE collects and reasons about information from a variety of sources, including physical sensors and user applications. In the architecture of the CE, shown in Figure 1, the *End User* uses an application that needs access to his or her contextual information. The application requests contextual information from the CE through the *CE API*. When appropriated, i.e., according to permissions granted to the application, privacy policies and user preferences, the CE will access contextual information or infer it using context inference tools, ultimately providing the requested information to the application. For further information about the CE, we refer to previous work, in which we explained the software service in greater detail [13].

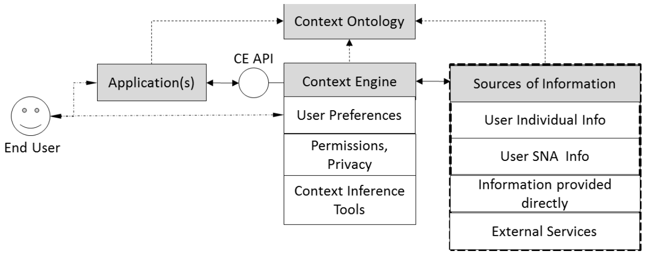

Figure 1. The Context Engine architecture simplified.

We illustrate the idea with an example. Consider an application whose function is to be an umbrella reminder: given the weather forecast on a particular day and the location of the user, it notifies the user whether or not to take the umbrella. In order to do so, it needs to access contextual information.

The inclusion of the CE in Smartphones encourages the development of context-aware applications, since application developers can delegate the context inference task to the CE, which in turn provides the contextual information automatically.

Moreover, different inference tools can be integrated into the CE. Typical examples of these context inference functions include activity recognition [14] and place detection [15].

B. System modelling using homophily

Social Network Analysis (SNA) focuses on the discovery and evolution of relations among entities (people, organizations, activities, etc.) [16]. SNA plays a major role in fields such as e-commerce [17]. Such e-commerce platforms analyze the social network in terms of tasks, e.g., purchases, searches and user similarity, with the ultimate objective of recommending relevant products to the user.

In particular, *homophily* is a social phenomenon often described as *the principle that a contact between similar people occurs at a higher rate than among dissimilar people* [5], shown to be ubiquitous in social networks [6] and is well-studied in the social sciences [5-7,18-22]. For instance, a study of the relationships among American high school students showed that they exhibit homophily by race and gender [18]. In other words, students tend to be more in contact with other students of the same gender and race.

Homophily has been used in numerous cases to model social networks [7,22-28]. Most of these investigations assume homophily to be present and create a homophily-based model, aimed at improving inference of the network. However, these models only assume homophily to be present but do not use their indicators in the final solutions.

Measuring the degree of homophily present in a system is relevant, since model-driven solutions can be built based on this characteristic. This allows comparisons between social networks. Ideally, these should be easy to understand.

Inverse homophily, also known as *heterophily*, is the inverse mechanism, where users tend to become connected to dissimilar users. A network that represents romantic relationships between students in an American high school, for instance, exhibits heterophily by gender [19].

It naturally follows to build an indicator of homophily that captures the degree to which homophily occurs in the system. To the best of our knowledge, a few indicators of homophily have been described [7,24,25] but are not always easy to interpret and seemingly fail to capture and utilize the heterophilic behavior of the network, i.e., they only capture homophilic behavior. For example, Tang *et al* investigate the use of three popular rating similarity measures as, what they called, the *homophily coefficient* [28]. On the other hand, Mislove *et al* derive their *affinity* indicator to represent the degree of homophily in the network with respect to a particular attribute [7]. Affinity, although derived along a similar train of thought as our homophily indicator *Hom,* which we define next, affinity remains unbounded and hard to manage (interpret and integrate) in context-aware solutions.

III. FORMAL DEFINTION OF HOMOPHILY

A. Network Definition

We introduce some basic graph notation such that $\mathrm{G} = (\mathrm{V}, \mathrm{E})$ denotes a finite undirected graph with nodes $\mathrm{V} = \{\mathrm{v}_1, \dots, \mathrm{v}_\mathrm{n}\}$, and edges $\mathrm{E} = \{\mathrm{e}_1, \dots, \mathrm{e}_\mathrm{m}\}$, where $\mathrm{n}, \mathrm{m} \in \mathbb{Z}$ are the number of nodes and edges in G, respectively. E contains the unordered pairs of nodes

$$\mathrm{e}_\mathrm{k} = (\mathrm{v}_\mathrm{i}, \mathrm{v}_\mathrm{j}) \; \forall \; \mathrm{k} \in \{1, \dots, \mathrm{m}\}, \mathrm{i}, \mathrm{j} \in \{1, \dots, \mathrm{n}\}$$

In short, we define $\#\mathrm{V} = \mathrm{n}$, $\#\mathrm{E} = \mathrm{m}, \mathrm{V}(\mathrm{G}) = \mathrm{V}$, and $\mathrm{E}(\mathrm{G}) = \mathrm{E}$ for convenience.

Particularly, we are interested in graphs with nodes annotated with contextual attributes. To model this, we define C as a function from nodes to finite vectors of Boolean attributes, i.e., $\mathrm{C}: \mathrm{V} \rightarrow \mathrm{B}^{\mathrm{S}}, \mathrm{S} \in \mathbb{Z}$, representing the size of B. We then reference $\mathrm{v_i}$'s contextual attributes as $\mathrm{C}(\mathrm{v_i}) = \{\mathrm{c_{i,1}}, \dots, \mathrm{c_{i,S}}\}$.

B. Quantifying homophily

Based on the graph definitions, next we characterize and measure the phenomenon of homophily. We derive our initial indicator *Hom* to quantify the potential degree to which homophily may be present at a single observation point in network G.

Since homophily emerges from the context, attribute $\mathrm{c_i}$ is used in the formulation of its definition:

Define two types of nodes in G, according to the binary value of $\mathrm{c_i}$, namely types p and q, where $\mathrm{V_p(G)}$ and $\mathrm{V_q(G)}$ are the sets of each type of node. The number of elements in each set is given by $\mathrm{n_p}$ and $\mathrm{n_q}$.

We consequently also define two types of edges, where edges between nodes of the same type are called homogeneous edges $\mathrm{E^+(G)}$ and edges between nodes of different types are called heterogeneous edges $\mathrm{E^-(G)}$.

Considering complete graph K, spanned from G, basic graph theory gives

$$|\mathrm{E^+(K)}| = \mathrm{n_p}\,\frac{\mathrm{n_p} - 1}{2} + \mathrm{n_q}\,\frac{\mathrm{n_q} - 1}{2}$$
$$|\mathrm{E^-(K)}| = \mathrm{n_p n_q}$$

Next, we define $\mathrm{r_G^+}$, $\mathrm{r_G^-} \in \mathbb{R}^+$ as the ratios of homogeneous and heterogeneous edges present in G, respectively, with respect to the homogeneous and heterogeneous edges in K. We have

$$\mathrm{r_G^+} = \frac{|\mathrm{E^+(G)}|}{|\mathrm{E^+(K)}|}$$
$$\mathrm{r_G^-} = \frac{|\mathrm{E^-(G)}|}{|\mathrm{E^-(K)}|}$$

Assuming at least one edge is present in G, we define our homophily indicator Hom for graph G as

$$\mathrm{Hom(G)} = \frac{\mathrm{r_G^+} - \mathrm{r_G^-}}{\mathrm{r_G^+} + \mathrm{r_G^-}}$$

The homophily indicator lies in the range [-1,1]. Positive values of *Hom* indicate that the networks exhibits a high potential of homophily, while negative values of *Hom* indicate that the network exhibits potential of heterophily, i.e., users are connected with dissimilar people. When the homophily value is close to 0, between -ε and ε, the system does not exhibit homophily. ε is thus the homophily threshold and it varies in different networks, depending on the size of the graph and the density of edges. The threshold is the way in which one deals with translating the theoretical definition of homophily into a practical working definition, i.e.,

$$\mathrm{Hom}\begin{cases} < -\varepsilon, & homophily \\ -\varepsilon \leq \mathrm{Hom} \leq \varepsilon, & \text{no homophily} \\ > \varepsilon, & heterophily \end{cases}$$

as mentioned by Easley and Kleinberg [22].

IV. Real-world Example

A. Nodobo dataset

We use the *nodobo* dataset for our real-world example. The dataset is publicly available and contains social interaction data of twenty-seven senior students in a Scottish high school. The data was collected using a software suite by the same name, developed by researchers at the University of Strathclyde, Scotland. They collected both device usage patterns and social interactions from *Google Nexus One* Smartphones [29].

They collected data over an interrupted period of roughly five months, namely from September 2010 to February of the following year. The data consisted of cellular tower transitions, Bluetooth proximity logs, and communication events, including calls and text messages. We build our social network graph from this data, as described next.

B. Experiment settings

We constructed social graphs from the dataset using only the data until the end of 2010, because four users matriculated and left school at that time. We built our graph *G=(V,E)* based on the Bluetooth proximity logs. We did not consider days when data were not collected. Hence, we considered a total of *D=105* days.

In order to study the behavior of the homophily in the system over time, and therefore the behavior of our indicator, we discretize time into *L* periods (or steps) of duration *W* days each.

Therefore, we have a sequence of *L* graphs, G_1, G_2, G_L, each representing the social interactions during the period *l*, whose state is observable at the end of that period.

Each participant in the experiments is represented as a vertex in *G*. A connection exists in G_l if and only if two students (vertices) have been in proximity to each other for an average of 60 minutes a day. We consider an edge from vertex A to vertex B to be *homogeneous* when the number of common friends of A and B is greater than integer *f* common friends, and heterogeneous otherwise. We thus have two control variables, namely *f* and *W*, that are varied to obtain different experiment settings.

We conducted two experiments, each calculating *Hom* and Affinity (*Aff*) [7] for the constructed graphs G_l. Intuitively, we expect to observe homophily in the graph because it represents social interactions.

The parameter setting for each of Experiment A and Experiment B are

- **Experiment A:** W=15, f =2
- **Experiment B:** W=5, f =3

The selection of these variables was at our discretion but we made sure to select values that explore two

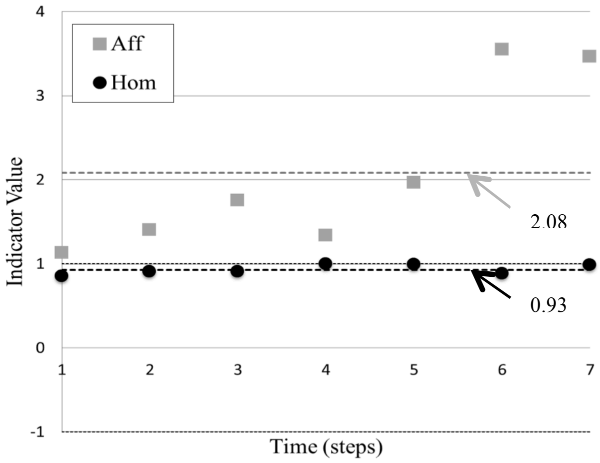

Figure 2. *Aff and Hom indicator values reported for Experiment A.*

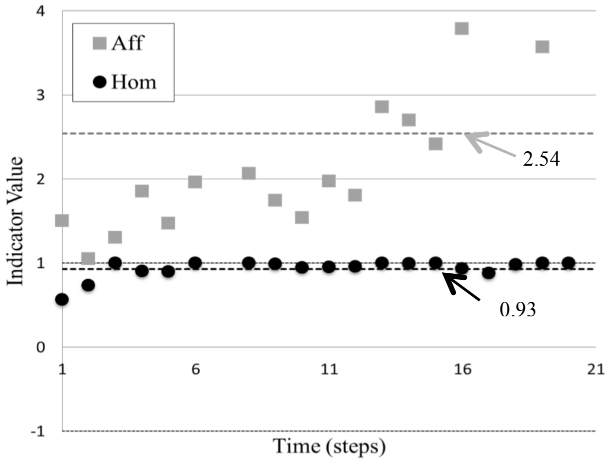

Figure 3. Aff and Hom indicator values reported for Experiment B.

configurations that result in two graph sets G_L that are different yet reasonable to experiment with.

C. Results

Values for both homophily indicators *Hom* and *Aff* are reported in Figures 2 and 3, for Experiments A and B, respectively. In Figure 3, Steps 7 and 21 have undefined values for both indicators and consequently, no value is shown. Since edges are not only introduced into the network but also removed, it is possible to have steps where there are no edges at all. Hence, as confirmed by both indicators reporting an undefined value, this is expected and verified.

Moreover, we expect to see homophilic system states to be reported by both indicators: almost all *Hom* values are greater than 0.9, where *Hom=1* implies complete homophily, whereas *Aff* values are all greater than 1, indicating homophily as well, and seems to increase over time.

Aff values vary to a greater extent than *Hom* values in both experiments. If an *Aff* value of around 3.5 is reported, as is shown in Figure 1 (*see* Step 7), it is relatively more challenging to understand what the relative difference means with respect to say about 1.5, reported for Step 2 of the same figure. The fact that there is no fixed and clear upper bound that allows for insight into the absolute values of the indicator and its difference is a big disadvantage of this indicator.

On the other hand, *Hom* values appear to be steadier, i.e., they do not vary as much. This is perhaps due to the normalization of our indicator, built into its definition. It sets upper and lower limits (-1 and 1) for the indicator and can be interpreted more easily and independently of other factors, such as the size of the network and the absolute number of edges present.

Furthermore, for each experiment, and each indicator, we show the mean of the values reported, as shown in the figures. The mean value for *Aff* differs by around 0.5 for each of the experiment settings. This can probably be interpreted by an expert of the indicator itself and SNA but even so, it might prove rather challenging.

However, attesting to the advantage of *Hom*, the mean value in both experiments was 0.93, with a small trailing difference. This accurately identifies that both experiments are of the similar systems, which was not suggested at all by *Aff*. The slight insignificant difference in mean values of *Hom* is most likely due to the discretization parameters we selected when configuring the experiments.

V. Conclusion and Future Work

By considering Social Network Analysis, one can reinforce context-aware computing, resulting in a better understanding of system behavior that needs to be predicted. We focused specifically on a phenomenon called homophily, and proposed an indicator *Hom* to report the potential of a system's state of homophily (or heterophily, for that matter). Our indicator can be used for descriptive purposes, i.e., for understanding the nature of the network. We also compared it to another indicator from the literature, called affinity.

The nature of each homophily indicator differs: affinity is unbounded on one end, having the range $[0,\infty)$, where a value of less than 1 indicates a state of heterophily. With this indicator, it is not easy to understand the degree of homophily in the network in terms of the absolute value reported, nor is it simple to compare to other systems without significant effort and knowledge about both systems and the indicator itself. When the system exhibits heterophily, the range would be much smaller, making the matter even more challenging.

To simplify and reduce the efforts needed by the average application developer, i.e., the non-expert, we make available *Hom*, bounded by the range [-1,1]. Positive values of *Hom* correspond to a state of homophily, while negative values correspond to a state of heterophily. This is easier to understand and interpret, especially since the homophily and heterophily values are symmetric.

These indicators are intended to be used as part of an inference solution, above and beyond simply modeling behavior. They need to be light-weight and simple, both of which features *Hom* embodies.

To extend our indicator and utilize it to predict context-related behavior in the stochastic system, more work needs to be done in terms of extending the network definition to account for time periods extending beyond a single time step.

Other noise features need to be filtered, accounting for behavior that opposes the natural phenomenon of homophily. A model-driven solution for context inference will benefit significantly if the factors of social network activity can be isolated and better understood.

Finally, the Context Engine requires tools and techniques that are not only accessible, accurate and effective for the non-expert, but also light-weight yet powerful. We are convinced that this initial homophily indicator is a step in the right direction towards reinforcing context-aware semantic computing.

Acknowledgment

This work was financially supported by EU FP7 Marie Curie Initial Training Network MULTI-POS (Multi-technology Positioning Professionals) under grant nr 31652.

References

[1] H. Lieberman and T. Selker, “Out of Context: Computer Systems That Adapt to, and Learn from, Context,” IBM Syst. J., vol. 39, no. 3–4, pp. 617–632. Jul. 2000.

[2] C. Perera, A. Zaslavsky, P. Cristen, and D. Georgakopoulos, “Context Aware Computing for The Internet of Things: A Survey,” Comm. Surveys & Tutorials, vol. 16, no. 1, pp. 414–454. 2014.

[3] J. Heggestuen, “Smartphone And Tablet Penetration - Business Insider.” Accessed online: 07/05/2015, http://www.businessinsider.com/smartphone-and-tablet-penetration-2013-10.

[4] B. Rao and L. Minakakis, “Evolution of Mobile Location-based Services,” Commun. ACM, vol. 46, no. 12, pp. 61–65. 2003.

[5] N. E. Friedkin, “A Structural Theory of Social Influence,” Cambridge University Press. 2006. ISBN 978-0-521-03045-8.

[6] M. McPherson, L. Smith-Lovin, and J. M. Cook, “Birds of a Feather: Homophily in Social Networks,” Annual Review of Sociology, vol. 27, no. 1, pp. 415–444. 2001.

[7] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, “You Are Who You Know: Inferring User Profiles in Online Social Networks,” in Proceedings of the Third ACM International Conference on Web Search and Data Mining, New York, NY, USA, pp. 251–260. 2010.

[8] G. Chen and D. Kotz, “A Survey of Context-Aware Mobile Computing Research,” Technical Report, Dartmouth College, Hanover, NH, USA. 2000.

[9] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, “Towards a Better Understanding of Context and Context-Awareness,” in Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing, London, UK, pp. 304–307. 1999.

[10] P. Mehra, “Context-Aware Computing: Beyond Search and Location-Based Services,” IEEE Internet Computing, vol. 16, no. 2, pp. 12–16. Mar. 2012.

[11] T. Gu, H. K. Pung, and D. Q. Zhang, “A service-oriented middleware for building context-aware services,” Journal of Network and Computer Applications, vol. 28, no. 1, pp. 1–18. Jan. 2005.

[12] M. Baldauf, S. Dustdar, and F. Rosenberg, “A Survey on Context-Aware Systems,” Int. J. Ad Hoc Ubiquitous Comput., vol. 2, no. 4, pp. 263–277. Jun. 2007.

[13] O. A. Nykänen and A. Rivero-Rodriguez, “Problems in Context-Aware Semantic Computing,” International Journal of Interactive Mobile Technologies (iJIM), vol. 8, no. 3, pp. pp. 32–39. Jun. 2014.

[14] L. Chen, J. Hoey, C. D. Nugent, D. J. Cook, and Z. Yu, “Sensor-Based Activity Recognition,” IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol. 42, no. 6, pp. 790–808. Nov. 2012.

[15] A. Rivero-Rodriguez, H. Leppakoski, and R. Piche, “Semantic labeling of places based on phone usage features using supervised learning,” in Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS), pp. 97–102. 2014.

[16] N. Belov, J. Patti, and A. Pawlowski, “GeoFuse: Context-Aware Spatiotemporal Social Network Visualization,” in Proceedings of the 13th International Conference on Human Computer Interaction. 2009.

[17] J. B. Schafer, J. Konstan, and J. Riedl, “Recommender Systems in e-Commerce,” in Proceedings of the 1st ACM Conference on Electronic Commerce, New York, NY, USA, pp. 158–166. 1999.

[18] J. Moody, “Race, School Integration, and Friendship Segregation in America,” American Journal of Sociology, vol. 107, no. 3, pp. 679–716. Nov. 2001.

[19] P. S. Bearman, J. Moody, and K. Stovel, “Chains of affection: The structure of adolescent romantic and sexual networks,” American Journal of Sociology, vol. 110, pp. 44–91. 2002.

[20] E. David and K. Jon, “Networks, Crowds, and Markets: Reasoning About a Highly Connected World,” New York, NY, USA: Cambridge University Press. 2010. ISBN 978-0-521-19533-1.

[21] D. B. Kandel, “Homophily, Selection, and Socialization in Adolescent Friendships,” American Journal of Sociology, vol. 84, no. 2, pp. 427–436. Sep. 1978.

[22] N. D. Lane, et al, ”Exploiting Social Networks for Large-Scale Human Behavior Modeling,” IEEE Pervasive Computing, vol. 10, no. 4, pp. 45–53. 2011.

[23] S. Aral and D. Walker, “Tie Strength, Embeddedness, and Social Influence: A Large-Scale Networked Experiment,” Management Science, vol. 60, no. 6, pp. 1352–1370. 2014.

[24] C. C. Aggarwal, “Social Network Data Analytics,” 1st ed. Springer Publishing Company, Incorporated. 2011. ISBN 978-1-441-98461-6.

[25] L. Wu, L. Yang, N. Yu, and X.-S. Hua, “Learning to Tag,” in Proceedings of the 18th International Conference on World Wide Web, New York, NY, USA, pp. 361–370. 2009.

[26] N. Eagle, A. Pentland, and D. Lazer, “Inferring friendship network structure by using mobile phone data,” PNAS, vol. 106, no. 36, pp. 15274–15278. Sep. 2009.

[27] R. Xiang, J. Neville, and M. Rogati, “Modeling Relationship Strength in Online Social Networks,” in Proceedings of the 19th International Conference on World Wide Web, New York, NY, USA, pp. 981–990. 2010.

[28] J. Tang, H. Gao, X. Hu, and H. Liu, “Exploiting Homophily Effect for Trust Prediction,” in Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, New York, NY, USA, pp. 53–62. 2013.

[29] S. Bell, A. McDiarmid, and J. Irvine, “Nodobo: Mobile Phone as a Software Sensor for Social Network Research,” Proc. Veh. Tech. Conf., pp. 1–5 . 2011.