Reality Mining at the Convergence of Cloud Computing and Mobile Computing

by Matthias Steinbauer, Ismail Khalil and Gabriele Kotsis

During any social interaction, nonverbal social signals convey just as much information as the conversation itself. While transmitting and analysing conversation is quite a common task for machines, the transmission and analysis of social signals is not. The convergence of cloud computing and mobile computing leads to a situation where insight into social systems is possible, thus paving the way for exciting new applications.

Over the last decade, we have witnessed an increasing usage of mobile devices for capturing, analysing, and predicting human behaviour in everyday activities.

Most modern mobile devices are equipped with a plethora of sensors that capture every aspect of the user's physical context represented by attributes such as time, location, light, sound, weather, temperature, or even physiological state. Combined with social computing applications such as blogs, email, instant messaging, social networking (Facebook, Twitter, Linked, Google+), Wikis, and social bookmarking, computers are able to capture the social context of the users in terms of interpersonal relationships and roles.

These physical and social contexts describe the existence of a relationship between two entities. Although the structure and nature of such relationships can be interpreted as a semantic network that can be used as the basis for understanding the meaning of an interaction, it fails to reflect the dynamics of a relationship over time. The dynamic patterns of interaction are essential in formalized procedures such as workflows, recurring sequences of actions (such as routine tasks), types of motion (such as walking, running and standing), tasks (such as having lunch, washing dishes, and driving) and goals (for example, socializing, hiring, selling, keeping fit or simply having fun) [1].

Reality mining is the collection and analysis of machine-sensed environmental data pertaining to human social behaviour, with the goal of identifying predictable patterns of behaviour, including how computers can learn to extract social clues from social systems [2].

In reality mining, techniques inherited from data mining and data analysis are applied to data generated by human interactions, ie phone call logs, e-mail messages, Bluetooth proximity logs and cell tower logs, etc.

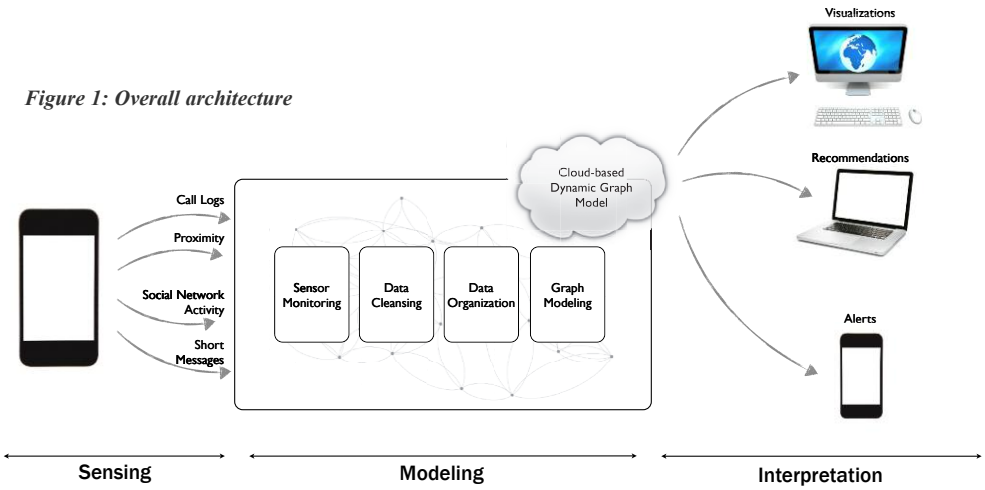
Reality mining analyses traces left by mobile devices, social networks and communication systems in the environment to extract social clues about the social network from which they originate.

Such clues can be used to detect communication bottlenecks in organizations (such as when a single individual connects departments) and behavioural stereotypes applying to each member of a work group in order to determine the roles of individuals within groups (who is a leader in which group, who is an expert in which group).

In our research we focus on cloud-based reality mining. We are aiming to provide tools and techniques that allow work groups to analyse their social network in near real time. Our objective is to improve group performance by providing a holistic overview of the group, its activities, situations and goals, in order to improve the group's overall performance.

The overall architecture of our approach is displayed in Figure 1. Reality mining comprises three phases: Sensing, modelling and interpretation.

Figure 1: Overall architecture



The sensing part is concerned with the recording, storing and transmission of sensor data that are generated by mobile phone sensors. These data are sent to a cloud based dynamic graph model for further processing.

The modelling phase, which shows the data processing pipeline in the cloud, consists of a sensor monitoring component in which raw sensor data are retrieved from roaming mobile devices. A REST service allows mobile clients to post their data to the service. In the data

cleansing component, the raw data are cleaned of duplicates and unnecessary content. This step also allows us to clean data from blocked phone numbers and e-mail addresses.

The data organization component performs a first alignment of the raw data to the dynamic graph model. Here we look up user profiles, and conversation topics or create them on demand if they did not exist.

Finally, in the graph modelling component, the data are inserted into the graph model as vertices and edges in a single transaction.

The final phase is the data interpretation in which the graph model is used to generate visualizations, recommendations and alerts.

Current research on reality mining tends to focus on one of two aspects: (1) implementing mechanisms and tools that run on mobile devices or (2) implementing methods to analyze data sets in the cloud with low latency.

Our research focuses on dynamic graph models. These models keep their history as the graph changes. In this way we are able to analyse various phenomena, for instance changes in the structure of social networks. On top of these dynamic models we use queries that adapt their results whenever the graph changes in order to reflect the new model.

We are working on integrating methods learned from this research into the design of collaborative applications and in applications of computer supported cooperative work.

Mobile devices have become an important platform for understanding social dynamics and influence, because of their pervasiveness, sensing capabilities and computational powers. The convergence of mobile and cloud computing is forming the breeding ground for real world applications in a research field known as reality mining. This is undoubtedly an interesting field, which is gaining new momentum with the convergence of mobile and cloud computing.

References:

[1] B. Begole: "Ubiquitous computing for business", Upper Saddle River, NJ: Financial Times Press; 2011
[2] Nathan Eagle and Alex Pentland: "Reality mining: sensing complex social systems", Personal and Ubiquitous Computing, 2006.

Please contact:

Matthias Steinbauer, Ismail Khalil, Gabriele Kotsis
Institute of Telecooperation, Johannes Kepler University Linz, Austria
E-mail: {matthias.steinbauer, ismail.khalil, gabriele.kotsis}@jku.at

Positioning Terminals in Mobile Computing Networks

by Francisco Barcelo-Arroyo, Israel Martin-Escalona and Marc Ciurana-Adell

The ability to pinpoint a terminal's position is useful for many applications of mobile computing and for network optimization (eg handovers, tariffs, resource management). A range of techniques are available to obtain a terminal's position [1]. GPS, for example, is used externally to the network and achieves good accuracy outdoors, with the trade off of increased energy consumption. Communication devices, however, are frequently used indoors, connecting to private networks, such as WLAN. Since GPS is inaccurate indoors owing to signal blockage and multipath errors, further research on indoors localization through communication networks is required. Mobile computing is linked to indoors positioning in applications such as: aged care, remote health control and security of buildings such as hospitals.

The fingerprinting technique is used extensively for WLAN positioning. The terminal collects the received signal strength from several access points and, during a precalibration phase, compares the achieved vector to the vectors previously recorded along with their positions. This technique does not involve modifications to the hardware. Other techniques use the time of flight (ie the time needed by a signal to travel between two nodes) to estimate the distances to several access points at known positions and then apply a trilateration process. The time of flight is more consistent than the signal strength. But in order to avoid modifications to the terminal's hardware, the time of flight must be obtained from communication messages by using only software.

Recent research at the Technical University of Catalonia has led to a procedure to measure distances between terminals [2]. This procedure is aimed at obtaining the time of flight after adding timestamps to messages sent and to the corresponding acknowledgements received. The round trip time (RTT) is computed as the difference between both timestamps, and the distance between the nodes is inferred by considering that the trip occurs at the speed of the radio signal. The software must interact with the link layer of the protocol stack of the device. A simple approach is to use a network interface capable of providing time measurements made by the hardware, but the timestamps performed do not have an acceptable accuracy for many location applications (eg the characteristics of IEEE 802.11 lead to a resolution of one microsecond corresponding to 300 metres in distance.)

A more sophisticated approach is presented in Figure 1. The protocol stack of the terminal is updated by introducing two software layers that are registered in each terminal. The registration process is run once and replaces the network interrupt handler (responsible for handling the events related with the net-