# Simone Centellegher

*Curriculum Vitae*

*Trento, Italy*
✉ *centellegher.simone@gmail.com*
*github.com/scentellegher*
*it.linkedin.com/in/scentellegher*

## Education

2015–current **Information and Communication Technology**, *University of Trento, Fondazione Bruno Kessler (FBK), Italy*, Phd student

2012–2015 **Computer Science**, *University of Trento, Italy*, Master's degree

2008–2012 **Computer Science**, *University of Trento, Italy*, Bachelor's degree

## Experience

### Professional experience

2015–current **Phd Student**, *Fondazione Bruno Kessler*, Italy, in the Mobile and Social Computing Lab led by Bruno Lepri

Aug 2016-Sep 2016 **Visiting Student**, *Scalable Cooperation group, MIT Media Lab*, Cambridge, MA, US

2006–2013 **Summer Jobs**

## Languages

Italian **Native**

English **Fluent**

## Skills

Operating Systems *Advanced*: Linux, Windows
*Intermediate*: Mac OS X

Programming Languages *Advanced*: Python (Pandas, Numpy, Scipy, Matplotlib), JavaSE, Java J2EE, Java WebServices (JAX-WS, JAX-RS), HTML, CSS, SQL
*Intermediate*: C, C++, Javascript

Tools *Intermediate*: MapReduce frameworks (Apache Hadoop, MRJob), Amazon Web Services(AWS), git

## Research interests

My research interests include understanding human behaviours from data generated by mobile phones, social media, credit card transactions and transportation cards, as basis for designing effective strategies for behavioural change and to induce cooperative and positive behaviours.

## Attendance to Schools and Conferences

Jan 2017 **NetSciX 2017 Conference**, *Tel-Aviv, Israel*
Oral Presentation in the parallel session Social and political networks.

May 2016 **Complex network: theory, methods and applications II**, *Como, Italy*
Summer School.

## Publications

2017 **Personality Traits and Ego-network Dynamics**, *Centellegher, S., López, E., Saramäki, J., Lepri, B.*, PLOS ONE

We interact with a wide network of people on a daily basis, and these social relationships play an important functional role in our lives. However, there are costs to their maintenance, resulting in a trade-off between quality and quantity, a typical strategy being to put a lot of effort on a few high-intensity relationships while maintaining larger numbers of less close relationships. Here, we focus on how personality traits of individuals affect this picture, using mobile phone calls records and survey data from the Mobile Territorial Lab (MTL) study. On broader terms, our study shows that personality traits clearly affect the ways in which individuals maintain their personal networks.

2016 **The Mobile Territorial Lab: a multilayered and dynamic view on parents' daily lives**, *Centellegher, S., De Nadai, M., Caraviello, M., Leonardi, C., Vescovi, M., ..., Lepri, B.*, EPJ Data Science

The exploration of people's everyday life has long been of interest to social scientists. Recent years have witnessed a growing interest in analyzing human behavioral data generated by technology (e.g. mobile phones). In this paper, we describe the Mobile Territorial Lab (MTL) project, a longitudinal living lab which has been sensing by means of technology (mobile phones) the lives of more than 100 parents in different areas of the Trentino region in Northern Italy. We present the preliminary results of the most complete picture of parents' daily lives that include their social interactions, mobility routines, spending patterns, and personality characteristics.

2016 **ETSCH: Partition-centric Graph Processing**, *Guerrieri, A., Montresor, A., Centellegher, S.*, Proc. of the 2nd International Conference on Cloud and Big Data Computing

The "big data" phenomenon has created the need to manage and analyze larger and larger datasets. A distributed system is able to cope with potentially unlimited datasets, is more robust to hardware failures, is often cheaper and it is also much easier to use than it was a decade ago. This paper presents ETSCH, a novel paradigm for processing large graphs. ETSCH departs from the vertex-based approach of BSP frameworks like PREGEL in two ways: first, the units of computation are not the vertices, but rather a collection of subgraphs; second, the subgraphs are obtained through an edge-partitioning algorithm and computations over the graph are then expressed using classical centralized algorithms executed on each of the partitions, with the only additional burden of specifying simple reconciliation procedures when vertices are replicated in multiple computing nodes.