CENTER FOR
ADVANCED
INTERNET STUDIES

Gioele Barabucci

Comparative Document Analysis as a Tool to Investigate Social Developments

CAIS Report

Fellowship
Mai bis Oktober 2017

GEFÖRDERT DURCH

Ministerium für
Kultur und Wissenschaft
des Landes Nordrhein-Westfalen

Comparative Document Analysis as a Tool to Investigate Social Developments

Humans have a natural tendency to produce documents that record what happens in the world around them. As time passes by, these documents are in turn modified to reflect how the world is changing. Studying how documents change, gives us insights into how societies have evolved and which trajectory they are taking.

The introduction of digital documents does not prospectively change this, but it requires the introduction of new scholarly methods to decipher and understand how this digitally stored information has been modified. The advent of networked computers, and in particular of the Internet, puts this problem on a global scale.

During my stay at the Center for Advanced Internet Studies, I investigated algorithms and computer-based techniques to explain these changes in human terms. The aim is to understand how these sterile modifications done to computer files relate to changes that happened in the real world and in society.

As a test case, I studied the evolution of the digital maps provided by OpenStreetMap, a collaborative project where thousands of volunteers are creating a freely available and extremely detailed map of the world. Many city councils, governments and other public bodies have contributed data from their systems to OSM. This data has been mechanically imported into OSM and then manually improved by many volunteers. Unfortunately, technical reasons make it impossible to give back to these virtuous entities the improved data.

To address this problem, I developed a prototype tool that can explain in human terms how a region of OSM has changed since a certain point in time. In addition to these explanations, the tool can also display in graphical form these modifications. This tool makes use of and binds together various novel conceptual frameworks: the universal delta model, the CMV+P document model, the extended unified patch format and the diffi comparison toolkit.

This tool has many possible applications:

- The human-readable explanations of the changes can be used by employers of the municipality to check if equivalent changes have been stored in their systems. Law enforcement agencies can check if the recorded changes agree with the local codes. Doing the same things using the description of the changes expressed in the OSM language is practically infeasible.

- The graphical form of these explanations can be used by journalists and fellow citizens to illustrate how parts of their cities are being transformed.
- The technical description of the changes can be used as a simplified input for the computer system of the public bodies that provided the original data.

My stay at CAIS allowed me to study in depth this particular case of changes that have a relevant impact on society. This study has confirmed that modifications of documents reflect changes in society. The study of document changes is, therefore, fundamental. Additionally, my inquiry has also highlighted the need for novel methods for the analysis of changes in digital documents. The current methods are focused on low-level technical aspects that fail to take into account the context and the semantics of the digital documents that form the backbone of our modern society.

Changes as Fundamental Entities for the Communication Between Computer Systems

Aside from its theoretical aspects, the concept of change and the techniques to detect and describe changes are central in many practical applications, in particular as a compression mechanism for networked computer systems.

Practically all networked computer systems need to maintain a synchronized state of the information they deal with. For instance, Google Docs need to keep the documents constantly updated for all the users that are editing them. Similarly, a chat program needs to keep the history of messages coherent for all participants in a chat and video games need to know in every moment the state of each player and their actions.

Applications do not send each other the whole state each time a change is made. Instead, the change is understood, encapsulated in what can be called an edit script, and sent to all the other participating applications that are on the network.

Unfortunately, few applications talk the same language when it comes to describing changes. Each application has its own “conceptual model” (i.e. the set of changes one application can detect and describe) as well as its own “wire format” (i.e. the exact binary representation of the detected changes).

The result is that different applications cannot synchronize with each other, even though they all manipulate the same content, often in similar manners.

The Import Problem: Why Virtuous Municipalities Are Unable to Benefit from the Work of OpenStreetMap

A concrete example of this synchronization problem, caused by the lack of a common language to describe changes, can be seen in OpenStreetMap, in particular with regards to data that has been donated by public bodies.

The OpenStreetMap is a collaborative online project whose aim is to create a complete and freely accessible map of the world. Thousands of volunteers contribute every day to OSM, for example by adding new streets, updating records about which shop is present in which building, describing roadworks and temporary blocks, etc.

The OSM project started in 2004 and has reached an almost complete coverage of all developed countries since then. As of 2018, most of the edits done, for instance, in Europe are changes to existing data based on changes in the real world (e.g., recently opened construction sites, changes of use of land allotments, changes of course of rivers).

During the years, many city councils, governments and other public bodies (to which we will collectively refer as municipalities) have contributed data from their archives and their systems to OSM – for example street names, building contours, infrastructure locations or suburbs borders. In technical terms, these data contributions are called imports.

The data donated by these public bodies contains many mistakes (e.g., outdated street names, incomplete drawings of newly developed areas, erroneous coordinates of building's outlines). After an import, OSM users spend a considerable amount of time cleaning up the imported data, bringing it up to the OSM quality standards.

Issues in Getting the Data Back

It would be nice, if the cities could take back the improvements made by the OSM volunteers after the import. This is, however, not possible, mostly for two reasons: one technical and one legal.

From a technical point of view, the main issue is that there is no easy way to isolate what has been changed. And even if there were, the systems used by the municipalities would probably not understand these changes expressed in the OSM data format. The municipalities face, thus, an all-or-nothing decision: replace their data with the OSM data or do nothing. Obviously, no municipality is going to move their systems over to OSM.

There are also legal issues: Is it possible at all to take back data from a volunteer project like OSM? Can the volunteers be trusted? Is the license of the OSM data compatible with the legal requirements of the municipality?

The Import Problem in Other Domains

The underlying issues that do not permit cities to receive improvements from OSM or provide updated data, are also the cause of similar problems in other domains, for example in document editing.

Documents edited using different applications (say, Open Office and Microsoft Word), cannot be synchronized. For example, it is possible to convert a document from one format to another (e.g. from DOCX to ODT), but afterwards the changes made in the ODT document cannot be merged back in the DOCX document. All collaborators are thus forced to use the same application. The initial conversion is equivalent to the import of data from the municipality into OSM, the subsequent inability to synchronize the two documents is equivalent to the municipalities being unable to integrate back the changes done after the import.

Solving (the Technical Part of) the Import Problem

These difficulties arise from the lack of a theoretical and algorithmic basis that allows a bidirectional flow of information. Many different pieces are missing:

- No formalization of high-level geographical data. Geospatial databases process and manipulate points, lines and areas. Humans work at a higher level of abstraction: streets, buildings, parks. These high-level concepts are not part of the data representation used by the databases.
- No formalization of the kinds of high-level changes that are possible on geographical data. Low-level changes tracked by geospatial databases are described in terms of points being moved around or lines being removed. Cartographers work with different and more structured categories such as "building boundaries have been redrawn to match ground reality" or "Victoria Avenue ends now at km 7, the rest is now called Alexander Avenue."
- No algorithms for reconciling operations done by different actors on maps. Most of the existing algorithms focus on simple plain text documents. For structured data like that recorded in geospatial databases there are only a few rudimentary algorithms available, and none can deal with high-level geographical data.

The proposed solution to address these issues consists of four steps:

1. Formalizing the types of data present in maps.
2. Formalizing the kinds of operations done on maps by expert cartographers.
3. Implementing of a proof-of-concept algorithm that converts low-level changes done to geospatial databases into high-level changes, allowing changes done by multiple actors to be explained using the terms of the art.
4. Using the output of the algorithm to present these changes in a machine-readable way as well as in a human-readable way, for example as short textual description or, graphically, as symbols superimposed on a map.

With these pieces in place, it will be possible to enable a sound bidirectional flow of updates from city systems to OSM and back, solving the presented problems.

A Theory and a Tool to Understand Changes in OpenStreetMap

The prototype I developed during my stay at CAIS is based on the following theoretical frameworks and programming toolkits that I previously developed in the field of document changes.

The **Universal delta model** allows algorithms to describe modifications not only in terms of low-level changes (e.g., these bytes have changed or those lines have been moved) but also in terms of high-level, domain-specific changes (e.g., "These 27 houses have been renumbered (+1) because of the addition of this building").

The **CMV+P document model** (Content, Model, Variant + Physical embodiment) allows applications to see documents as stacks of level of abstraction, each built on top of the others. For example, a map is, at the same time, a representation of various geographical entities as well as a complex structure of lines and points, a set of coordinates and a series of bits. Under the CMV+P document model, all these views coexist at the same time and the data at one level is connected with the relevant data at other levels of abstraction.

The **Extended Unified Patch format** is a file format that allows the description of changes detected at various CMV+P abstraction levels. A single EUP file generated comparing the state of an OSM region in two points in time describes both how the low-level features have changed (node N moved to posi-

tion X,Y, line L new ending point is at V,W) as well as how high-level features have changed (semaphore light has been moved 3 meters east, a new residential house has been built at coordinate X,Y).

The **diffi comparison toolkit** is a document comparison tool that compares content instead of just files. Using the aforementioned frameworks UniDM and CMV+P, it can detect changes done to various kinds of content (including OSM maps) at various levels of abstraction. Found changes are then stored using a EUP diff file.

The **explain-osm** tool explains in human-readable terms the differences found by diffi. It analyses the changes stored in a EUP diff file and generates a textual description of these changes as well as a visual representation of these changes.

The textual descriptions produced by explain-osm use a vocabulary that is familiar to the public and, in particular, to the experts of survey and cadastre systems. Other OSM-based comparison tools use a more obscure and technically oriented OSM-specific language to describe the same changes.

The visual representation of the changes combines the textual explanations with a symbolic graphical representation of the changes (the graphical representation has been adapted from the open source project *Show Me The Way*). The symbols codify which changes have been detected and are overlaid on top of satellite photos of the area in question.

Impact and Open Questions

In the particular case of geographic data and OSM, the ability to track changes and easily understand their impact has many positive effects. It can:

- let city councils, governments and other public bodies save money by reducing the time they need to spend on cumbersome but necessary data-maintenance tasks, making use of the work of volunteer citizens;
- increase the transparency of real-estate data;
- make sure that residents and tourists always have up-to-date maps.

Solving the technical problems does not, however, solve the legal and social issues. There are still some questions left open:

- Have public bodies the right to integrate or to use data coming from OSM?
- Should every change be vetted by an authorized person or can changes be applied without human supervision?
- How extensive should this supervision be? Should every change be checked “on the ground”?
- If some changes are not accepted, is there an ethical obligation to publicize this decision?

Table of Figures

Photo Titlepage: CAIS, Matthias Begenat

Contact

Gioele Barabucci
Cologne Center for eHumanities, Universität Köln