Matching and Conflation of Open Government Data with OpenStreetMap Data

Measuring the Similarity of Points-of-Interest

Student

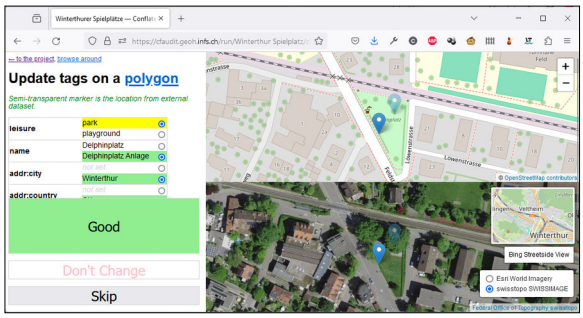
Claudio Bertozzi

Initial Situation: Data stewards seeking to integrate Open Government Data (OGD) into their strategic frameworks are driven by the overarching goal of improving data accessibility. OpenStreetMap (OSM), a prominent geospatial platform, has emerged as a focus for such efforts. The pressing question facing OGD implementation initiatives concerns the selection of tools and methodologies that can effectively facilitate this integration. A previous thesis has already evaluated some tools to achieve this goal. The web application OSM Conflator (Audit) was identified as the most suitable tool for data conflation (Fig. 1). They also suggested improvements to help the user find duplicates, one of the many difficulties in automating this task (Fig. 2).

Definition of Task: The main goal of this work is to improve the automatic matching of two POIs and to improve the process of conflating OGD or other sources like the "All The Places" project into OSM. The hypothesis is that new machine learning based algorithms could help here. OpenLR has to be analyzed as part of the state of the art. It is an open standard for encoding places on digital maps, independent of the map provider. The overall goal is to improve existing methods and increase the efficiency and effectiveness of geoinformation systems, ultimately contributing to more effective and user-friendly geospatial processes, not limited to OGD.

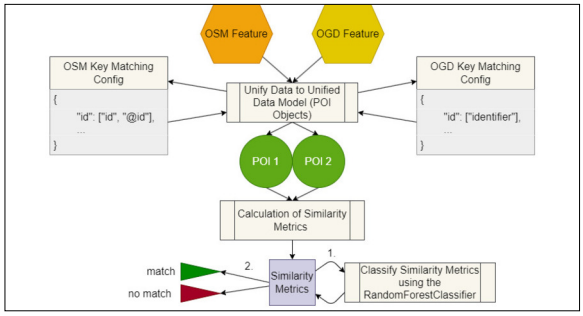
Result: Our research has shown that straight text comparisons and proximity are often insufficient to find all matching POIs from OGD in OSM. An OpenLR was found to be unsuitable because it focuses mainly on matching locations based on their geometric properties, especially lines, whereas point geometry and a more flexible data structure are required. In a paper analyzing different methods for POI matching, a combined approach of distance measurement and text comparison was found to be effective and used in a classifier. Their work is based on another paper "Towards Automatic Points of Interest Matching" (2020). This approach was used as a basis here. The final algorithm, implemented in Python, works as follows: The geospatial input data (JSON) is transformed into a unified data model. Then, a configuration file specifies the important parts in the JSON to match the keys to the members of the unified model, i.e., to indicate the importance of selected attributes. Finally, the two unified POIs are used to compute the similarity metrics. These metrics are then used by a random forest classifier to determine whether the POIs match or not (Fig. 3). To develop this algorithm, a representative dataset of matching and non-matching POIs was crucial. Since no such dataset was found, a dataset of 200,000 POIs from OSM within Switzerland was used to create an artificial dataset of matching and non-matching POI pairs for training. This was created by taking two different POIs or by performing different data manipulations on an existing POI. Either the newly created POI is different enough to represent another POI, or it is a match. Such two matching POIs can also be identical. The test results are satisfactory and our algorithm is ready to be integrated into OSM Conflator and other software. However, a more flexible data mapping approach to convert OGD to the unified model is inevitable. I.e. additional tests with other real data are needed. Further work will focus on integrating the algorithm into OSM Conflator with appropriate improvements.

Web-based OSM Conflation Audit tool guides data stewards through semi-automated data conflation in OSM
Own presentment



The difficulty of POI matching, no standardized data structure leads to a complex problem: When are two POIs equivalent?
Own presentment with OpenStreetMap standard map

Custom dataflow diagram of our algorithm used to unify and classify JSON data from different sources
Own presentment



Advisor
Prof. Stefan F. Keller

Subject Area
Data Science, Computer Science