Building Linked Data from Historical Maps

Chun Lin[1], Hang Su[1], Craig A. Knoblock[1], Yao-Yi Chiang[2], Weiwei Duan[2], Stefan Leyk[3], and Johannes H. Uhl[3]

[1] University of Southern California
Information Sciences Institute and Department of Computer Science
{chunlin,hsu271,knoblock}@isi.edu
[2] University of Southern California
Spatial Sciences Institute and Department of Computer Science
{yaoyic,weiweidu}@usc.edu
[3] University of Colorado, Department of Geography
{stefan.leyk,johannes.uhl}@colorado.edu

Abstract. Historical maps provide a rich source of data for social science researchers since they contain detailed documentation of a wide variety of factors, such as land-use changes, development of transportation networks, changes in waterways, destruction of wetlands, etc. However, these maps are typically available only as scanned documents and it is labor intensive for a scientist to extract the needed data for a study. In this paper, we address the problem of how to convert vector data extracted from multiple historical maps into Linked Data. We describe the methods for efficiently finding the links across maps, converting the data into RDF, and querying the resulting knowledge graphs. We present preliminary results that demonstrate that our approach can be used to efficiently determine changes in the Los Angeles railroad network from data extracted from multiple maps.

Keywords: historical maps · Linked Data · vector conflation

1 Introduction

Historical map archives contain valuable geographic information on both natural and man-made features across time and space, but the information is only available as scanned images. There exist many studies in developing technologies for extracting information from scanned historical maps to then integrate the extracted information with other datasets in Geographic Information Systems (GIS) [2, 4]. This line of work enables long-term spatiotemporal analyses such as detecting the changes in railroad networks between two map editions of the same region (Figure 1), which can be useful for the development of transportation infrastructure, etc.

However, there are still major challenges with integrating datasets extracted from scanned maps and using them for analytical tasks. First, existing work on integrating vector datasets such as conflation [11] focuses on reconciling different sources for improving data accuracy or enriching data attributes and does not consider changes over spatial scale or time. Second, once the vector datasets have

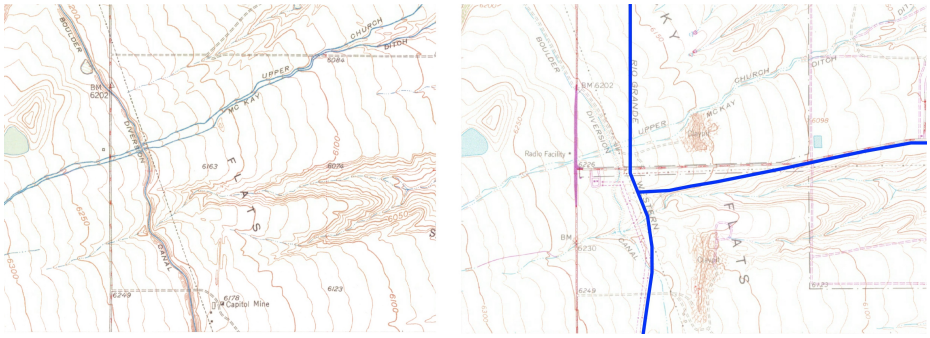

Fig. 1. *Railroads appearance (blue) from 1950 to 1965 in Louisville, Colorado*

been aligned and matched across space and time, the information still needs to be organized and related to other datasets to support efficient analysis.

To enable change analysis over time periods and across spatial scales, we present an approach to integrating map data using Linked Data as the representation. This approach is not only helpful in making the data widely available to researchers, but also in enabling the ability to answer complex queries from social sciences researchers, such as investigating the interrelationships between human and environmental systems. The approach also benefits from the open and connective nature of Linked Data. Compared to existing tools such as PostGIS[4] that can only handle queries related to geospatial relationships within local databases, Linked Data can utilize other widely available knowledge sources (e.g. GeoNames) in the Semantic Web and enable rich semantic queries.

In this paper, we present a general pipeline for constructing the semantic representation of extracted map data, which is demonstrated using railroad geographic features. The pipeline consists of three major steps (Figure 2). The preprocessing step includes 1) automatic line segmentation, using PostGIS to process railroad features, and 2) data preparation to generate necessary metadata for semantic modeling. Then, we use Karma[5] to map the geographic features to an ontology and publish the mapped data in RDF. Finally, we run SPARQL queries against Apache Jena, which stores the geospatial Linked Data.

[4] `https://postgis.net/`
[5] `http://karma.isi.edu/`

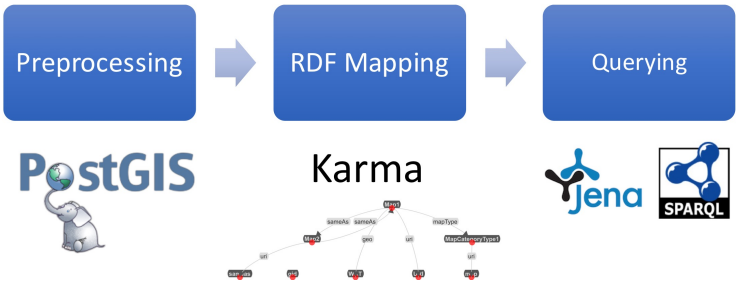

Fig. 2. *General pipeline for processing the extracted map feature data*

2 Data Preprocessing and Linking

Given a set of railroad map vector data extracted from separate map editions covering the same region, the goal is to match the line segments that represent the same line segments across multiple maps published at different points in time. Such relationships are key to support the queries for finding changes in the feature of interest through time.

2.1 Automatic Segmentation and Linking

The first challenge that we need to address is vector-to-vector matching. Consider an example consisting of vector features from two maps (Figure 3), where map A represents an older map edition and map B is the latest map edition with a part of the railroad that has been changed. In order to link only the parts that are common across the two maps, we split each of these vector features into several features, as shown in the figure.

In order to efficiently identify the portions of the railroads that are common across two maps, we use PostGIS, which is a powerful PostgresSQL extension for spatial data storage and query. PostGIS offers various functions to manipulate and transform geographic objects in databases. As shown in Figure 3, first we create buffers for the vector data in Map A and B for a particular buffer size using the PostGIS function `ST_Buffer`. Second, we extract the buffer intersection with the function `ST_Intersection`. Third, we use this buffer intersection to run `ST_Intersection` again with each of the original vector features from map A and map B to determine where to split the line segments in each of the original maps. The process records both the new segmentation and the "sameAs" relationships among the split line segments if their buffers intersect.

The segmentation process can also handle vector datasets from more than two map sources. With multiple vector datasets as the input, the segmentation process first processes two datasets and then integrates more datasets one at a time. During the integration step, the segmentation process stores the "contains" hierarchical relationship (Figure 4) for tracking the segmentation result for each additional dataset and uses the hierarchical relationship to handle the "sameAs" relationships for multiple sources.

The data structure and steps for handling multiple datasets are as follows. In the hierarchical relationship, each node in the tree stands for a line segment from

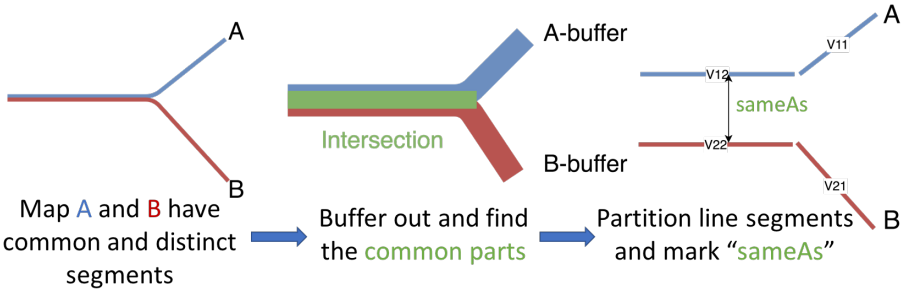

Fig. 3. *Line segmentation for two map vector features: spatial buffers are used to identify the same line segments considering potential positional offsets of the data.*

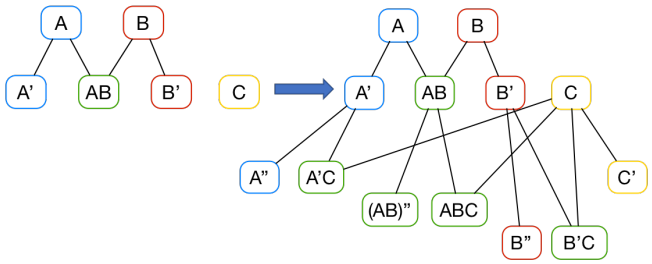

Fig. 4. *"Contains" relationship in segmentation of more than two maps*

one or more vector data sources. The nodes with two parents are the common line segments shared by two maps. When a third map source, C, comes in, the segmentation process repeats the `ST_Buffer` and `ST_Intersection` procedures using the line segments of C and all of the leaf nodes at the second level of the tree, generates new segmented line segments shown as the new leaf nodes, and records the newly identified "sameAs" and "contains" relationships (Figure 5).

2.2 Data Preparation

Once we identify the "sameAs" relationships, we need to convert the data into Linked Data. To do so, we need to perform several additional data preparation steps.

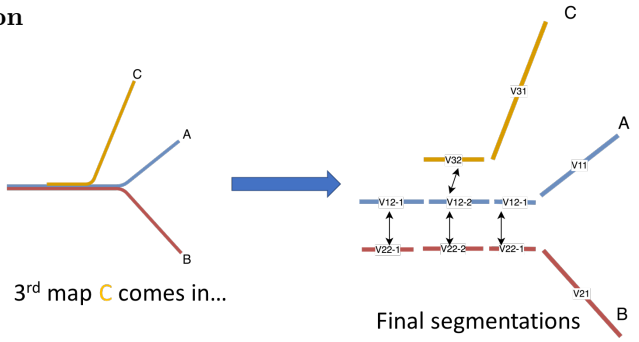

Fig. 5. *Incrementally integrate vector datasets from multiple map sources*

Geometry representation There are several popular methods for representing geometries, such as GML (Geography Markup Language), WKT (Well-Known Text), and WKB (Well-Known Binary). We need to describe the geometries in a compact and human-readable way; therefore WKT format is chosen for further preprocessing.

URI The first thing we need for representing the integrated vector data as Linked Data is the unique IDs for individual line segments. We could use the geometry representation of a line segment as its unique ID because two geographic line segments typically are not exactly same. However, a line segment in the WKT format is a long string, containing commas and spaces which are invalid for a URI. To overcome this problem, we apply a hash function on the line segment in WKT to obtain a relatively short string to describe the URI. Considering hash collision, especially two line segments from different maps may be similar, we add additional metadata of individual line segments (i.e., the source map) to the corresponding hash results.

Relationship between line segments Since line segments are represented by URIs, which are persistent, we can add relationships between two line segments using their corresponding URIs. The relationships between line segments include "sameAs" to link segments across maps and "contains" to describe how a segment is broken down into subsegments. If an existing line segment has to be split into smaller segments to link with a new map source, the split segments will have their own URIs and will be linked to the original line segment with the "contains" relationship.

Metadata Most line segments from a map share the same metadata, so we only store the metadata for map entities and link the line segments to their source map instead of storing the metadata for each line segments.

3 Creating Vector Linked Data

To represent geographic line segments in RDF, we use the semi-automatic tool Karma [6] to map structured data generated from the previous steps to ontologies defined by schema.org. The output is map data stored as RDF triples in a knowledge graph based on the ontologies.

Entity The map metadata are represented by *schema:MapType*, and the geographic vector data are described by *schema:Map*.

Relationship If two line segments from multiple maps are identified as the same entity, their relation is *schema:sameAs*. If a line segment is derived from a historical version of a segment (i.e., existing segment from a previous source), their relation is *schema:contains*.

We want to avoid updating all of the existing geographic vector data. Adopting this method, we store historical version of knowledge graph first, and when a new map comes in, we just need to handle it incrementally, i.e. just import new vector data and their connections resulting from the steps described in Section 2.

4 Querying

With the knowledge graph generated using Karma, we can set up queries to solve some interesting, real-world problems. Figure 6 provides an example query for finding the changed segments of railroads in two map editions (circa 2000 and 2005).

```
PREFIX schema: <http://schema.org/>
select distinct ?a ?mapa
where {?a schema:geo ?geo.
       ?a schema:mapType ?mapa.
       ?mapa schema:releaseDate "2000".
       filter not exists{
          ?a schema:sameAs ?b.
          ?b schema:mapType ?mapb.
          ?mapb schema:releaseDate "2005".}
       minus{?a schema:contains ?x}}
```

Fig. 6. *Query to find the changes in railroads between maps*

Line segments from more than one source are also stored in the knowledge graph to track the provenance. In the query result, we only return the latest version of the line segments that meets the query criteria, i.e., the leaf nodes, because they are the smallest line components after the split.

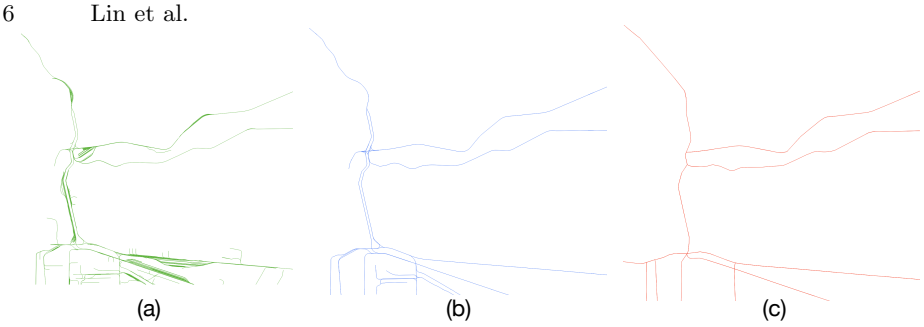

Fig. 7. *Railroad map data of Los Angeles from three different sources*

5 Case Study

To test our approach on real datasets, we used the railroad data from three sources: 1) USGS vector data for Los Angeles, California[6] (822 features), 2) California Rail Network[7] (149 features), and 3) National Atlas of the United States[8] (45 features). All the data was input in ESRI shapefile format and we cropped them for the area of Los Angeles (Figure 7).

Line segmentation in PostGIS We transformed all data to the same Coordinate Reference System (CRS), EPSG:4269,[9] buffered map (a) and map (b) with 0.0005 in degrees, segmented the vector data, and generated the "sameAs" and "contains" relationships. When we added map (c), we buffered line segments from maps (a) and (b) as well as line segments from map (c) with 0.0015 in degrees. This larger buffer for handling map (c) is because map (c) has a smaller map scale compared to maps (a) and (b) due to increasing levels of generalization in feature representation, hence the positional offsets is potentially larger (than between maps (a) and (b)). Then we repeated the intersecting step until all relationships of "sameAs" and "contains" among three maps were identified. It took 36 seconds altogether to run all queries computing buffers and intersections for processing the three maps.

Data preparation The identified relationships and segmented vector data were stored in PostgresSQL. In this step, we exported the tables from PostgresSQL to CSV files. We used the WKT format to represent geometries and added URI and other metadata including map year and source so that we could perform semantic queries to find changes of the vector data by years in the next step. Here we recorded map (a) with the year 2000 and map (c) with the year 2005.

Create vector Linked Data CSV files were loaded into Karma to map the data to ontologies and produce output in RDF format.

Querying RDF files were then stored in Apache Jena, where we could query to find the difference between maps using SPARQL. Using the query in Section 4 we produced the result shown in Figure 8 in 63ms. The visualization was created with the resulting WKT using QGIS.

[6] `https://viewer.nationalmap.gov/basic/`
[7] `http://www.dot.ca.gov/hq/tsip/gis/datalibrary/Metadata/Rail_13.html`
[8] `https://nationalmap.gov/small_scale/`
[9] `https://epsg.io/4269`

6 Related Work

The mainstream work on geospatial data integration focuses on vector data conflation, which is a reconciliation process of two maps in the same area to achieve a better positional accuracy for one of the datasets (e.g., [1]). Ruiz et al. [10] discussed different types of conflation methods, including vector to vector conflation and semantic conflation. The former involves feature matching between different datasets (e.g., [3, 8]), which is also a crucial step for the automatic line segmentation in our work. On measuring the similarity of vector geometries, Sherif et al. [13] conducted a survey on existing ten point-set measures in the context of geospatial Linked Data.

Fig. 8. *Query result: railroads in 2000 but not in 2005, difference in green*

Using Linked Data to integrate geospatial information is a growing topic, which enables the data integration process to take advantage of the open and distributed Linked Data sources with great opportunities and challenges at the same time [7]. U.S. Geological Survey (USGS) has developed an initial approach for The National Map to build ontologies, match features, and support Linked Data queries [14]. Using Linked Data is also useful in semantic conflation, allowing non-geometric features to play a key role in the integration process, but this approach is mainly studied for conflating points of interest data [12, 15]. Our approach solves the integration problem of geospatial linear feature matching using the RDF representation published as Linked Data.

In addition, current work on geospatial change analysis (e.g. glacier change [9] and geomorphic change [5]) were mainly accomplished with GIS software such as PostGIS and ArcGIS. Our approach opens the possibility of analyzing geospatial data with richer attribution and increasing detail.

7 Discussion

We presented an end-to-end approach to integrating geographic data from multiple sources and publishing the integrated data as Linked Data. We demonstrated the approach with railroad vector data extracted from three maps and showed queries for identifying changes in the vector data over time. This approach would also work for other types of linear features, and the case study uses railroads to explain the idea. The resulting tools and datasets will be beneficial for geography and social science researchers for conducting change analyses related to transportation infrastructure, land use, and land cover.

In future work, we plan to explore the use of GeoSPARQL for querying geographic RDF data and creating links to other existing sources to answer other types of research questions. For example, the addition of demographic data as Linked Data would allow us to examine relationships between changes in population distributions and changes in transportation infrastructure.

8 Acknowledgement

This material is based on research sponsored in part by the National Science Foundation under Grant Nos. IIS 1563933 (to the University of Colorado at Boulder) and IIS 1564164 (to the University of Southern California).

References

1. Chen, C.C., Knoblock, C.A.: Conflation of geospatial data. In: Shekhar, S., Xiong, H. (eds.) Encyclopedia of GIS, pp. 133–140. Springer US, Boston, MA (2008)
2. Chiang, Y.Y., Leyk, S., Knoblock, C.A.: A survey of digital map processing techniques. ACM Computing Surveys (CSUR) **47**(1), 1–44 (2014)
3. Cobb, M.A., Chung, M.J., Foley III, H., Petry, F.E., Shaw, K.B., Miller, H.V.: A rule-based approach for the conflation of attributed vector data. GeoInformatica **2**(1), 7–35 (1998)
4. Duan, W., Chiang, Y.Y., Knoblock, C.A., Leyk, S., Uhl, J.H.: Automatic generation of precisely delineated geographic features from georeferenced historical maps using deep learning. In: The 22nd International Research Symposium on Computer-based Cartography and GIScience (Autocarto/UCGIS) (2018)
5. James, L.A., Hodgson, M.E., Ghoshal, S., Latiolais, M.M.: Geomorphic change detection using historic maps and dem differencing: The temporal dimension of geospatial analysis. Geomorphology **137**(1), 181–198 (2012)
6. Knoblock, C.A., Szekely, P., Ambite, J.L., Goel, A., Gupta, S., Lerman, K., Muslea, M., Taheriyan, M., Mallick, P.: Semi-automatically mapping structured sources into the semantic web. In: Extended Semantic Web Conference. pp. 375–390. Springer (2012)
7. Kuhn, W., Kauppinen, T., Janowicz, K.: Linked data-a paradigm shift for geographic information science. In: International Conference on Geographic Information Science. pp. 173–186. Springer (2014)
8. Li, L., Goodchild, M.F.: An optimisation model for linear feature matching in geographical data conflation. International Journal of Image and Data Fusion **2**(4), 309–328 (2011)
9. Raup, B., Racoviteanu, A., Khalsa, S.J.S., Helm, C., Armstrong, R., Arnaud, Y.: The glims geospatial glacier database: a new tool for studying glacier change. Global and Planetary Change **56**(1-2), 101–110 (2007)
10. Ruiz, J.J., Ariza, F.J., Urena, M.A., Blázquez, E.B.: Digital map conflation: a review of the process and a proposal for classification. International Journal of Geographical Information Science **25**(9), 1439–1466 (2011)
11. Saalfeld, A.: Conflation: automated map compilation. International Journal of Geographical Information System **2**(3), 217–228 (1988)
12. Sehgal, V., Getoor, L., Viechnicki, P.D.: Entity resolution in geospatial data integration. In: Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems. pp. 83–90. ACM (2006)
13. Sherif, M.A., Ngomo, A.C.N.: A systematic survey of point set distance measures for link discovery. Semantic Web Journal, pg.18 (2015)
14. Usery, E.L., Varanka, D.: Design and development of linked data from the national map. Semantic web **3**(4), 371–384 (2012)
15. Yu, F., McMeekin, D.A., Arnold, L., West, G.: Semantic web technologies automate geospatial data conflation: Conflating points of interest data for emergency response services. In: LBS 2018: 14th International Conference on Location Based Services. pp. 111–131. Springer (2018)