Exploiting Semantics of Web Services for Geospatial Data Fusion

Pedro Szekely, Craig A. Knoblock, Shubham Gupta, Mohsen Taheriyan, Bo Wu

University of Southern California
• Decision makers have lots of data available
  — Satellite imagery
  — Street maps
  — Structured online sources (e.g., phone books)
  — Cyber data (e.g., domain registration sites)
  — Social network data (e.g., Facebook)
• Difficult to fuse this information into an integrated view
  — Even harder to apply various reasoning techniques
• Our goal
  — An integration framework where users can interactively fuse geospatial and other types of data
KARMA: A General Information Integration Tool

- A fusion-by-example approach for extracting, modeling, cleaning and integrating geospatial sources
  - Does not require any programming or widget knowledge.
  - Focus on data, not on the process
  - Users specify fusion tasks by examples
  - Fusion results automatically displayed on a map

Data Types Supported

Spreadsheet Type Interface

Information Integration Operations

Data Types Supported
Motivating Example

• Problem: Identify the address associated with each building that can be identified in the satellite imagery

• Solution:
  — Step 1: Identify the street vector data, building locations and the phonebook data for the given area (data retrieval task)
  — Step 2: Reasoning over the data to generate a mapping between the addresses and building locations (geospatial reasoning task)
Motivating Example (cont.)

Data Sources

- Points: Locations of buildings
- Streets: Labeled road network
- Yellow Pages: Businesses info in area
- White Pages: People info in area

Reasoning Services

- Find Closest Vector: Find the streets that each building could be located
  - [Building Coordinate, Street Name(s)]
- Parse Address: Takes address and extracts street name and building number
  - [Street Name, Building Number]
- Map Points to Addresses: Returns possible address of each building
  - [Coordinate, Building Number, Street Name]

How to integrate reasoning algorithms and services into Karma?
Karma’s Approach

• Build a semantic model of reasoning services based on provided ontology
  — Data types of inputs and outputs, plus relationships between them

• Interactively invoke services using semantic model of sources and services
  — Which services can be invoked using available data?
  — Which sources can satisfy service inputs?

• Integrate outputs of service invocation with the other data
Karma uses wrappers to extract web pages information
Karma uses learned transformation rules to remove all instances of `&nbsp`.

User provides examples of address without `&nbsp`.

<table>
<thead>
<tr>
<th>String</th>
<th>User Defined Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>CARA LAZARA 15/2, BEOGRAD</td>
<td>CARA LAZARA 15/2, BEOGRAD</td>
</tr>
<tr>
<td>CARA LAZARA 13/21, BEOGRAD</td>
<td>CARA LAZARA 13/21, BEOGRAD</td>
</tr>
<tr>
<td>NIKOLE SPASIĆA 2, BEOGRAD</td>
<td>NIKOLE SPASIĆA 2, BEOGRAD</td>
</tr>
</tbody>
</table>
• Karma automatically builds models of data according to provided ontology
  — Models help user to process data and integrate them

• Identify the semantic types
  — Supervised machine learning technique (CRF Model)
    ▪ A. Goel, C. A. Knoblock, K. Lerman, Using conditional random fields to exploit
      token structure and labels for accurate semantic annotation, 2011

• Identify relationships among the data columns
  — Find the minimal tree that connects the semantic types
    ▪ C. A. Knoblock, P. Szekely, J. L. Ambite, S. Gupta, A. Goel, M. Muslea, K.
      Lerman, Interactively Mapping Data Sources into the Semantic Web, 2011
I. Karma uses CRF technique to assign labels to each data column
II. Karma selects the smallest tree that connects these semantic types and shows it at the top of the data worksheet.
Modeling Web Services

- Semantic models of web services facilitate service invocation, discovery, and composition

- Karma allow the user to interactively build a model
  - User provides examples of service input and output
  - Modeling services can be done like data sources
Modeling of Parse Address Service in Karma

Importing Sources
Data Cleaning
Source Modeling
Service Modeling
Data Fusion
Visualization

Final service model
• Ability for users to interactively invoke services on other data sources

• Semantic models make it possible to:
  — Automatically determine which services apply to the available data
  — Perform automatic transformations on data to get it into the required format to apply a service
  — Automatically compose services and sources to generate required data

Karma identifies which source and which elements of that source satisfy this invocation.
### Matching Sources and Services

**Importing Sources**
- Data Cleaning
- Source Modeling
- Service Modeling
- Data Fusion
- Visualization

#### Yellow Pages
- Points
- Streets
- Find Closet Vector
- Parse Address
- Map Points to Addresses

#### White Pages
- Parse Address

#### Input
- **Address**
  - (hasValue) AddressString
  - (hasName) String
  - (hasStreet) Street
  - (hasBuildingNumber) BuildingNumber

#### Output

<table>
<thead>
<tr>
<th>String</th>
<th>BusinessCategory</th>
<th>AddressString</th>
<th>Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>Državna Lutroša Srbije</td>
<td>igre na sreću</td>
<td>Serbia, Beograd, Vraćar, Uškača</td>
<td>Phone: 011 202 9292</td>
</tr>
<tr>
<td>Fun casino</td>
<td>kazina</td>
<td>Serbia, Beograd, Uškačka 4</td>
<td>Phone: 011 627 605</td>
</tr>
<tr>
<td>Grand t.t.</td>
<td>turistička agencije</td>
<td>Serbia, Beograd, Uškačka 7</td>
<td>Phone: 011 328 4955</td>
</tr>
<tr>
<td>Menjačnica srbijskog</td>
<td></td>
<td>Serbia, Beograd, Uškačka 4</td>
<td>Phone: 011 263 1846</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>String</th>
<th>BusinessCategory</th>
<th>AddressString</th>
<th>Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>CARA LAZARA 15/2, BEograd</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CARA LAZARA</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CARA LAZARA 13/21, BEograd</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NIKOLE SPASIC 2, BEograd</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NIKOLE SPASICA</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Invocation Results

- Results of invocation are returned as another source that can be refined, integrated with other sources, visualized or published.
Visualize Final output

- yellowpage.csv
- whitepage.csv
- Beo_streets.kml
- Beo_points.kml
- Find Closest vector
- Parse Address
- Map Points to Addresses
• **Exploit ontologies to attach semantics to geospatial services**
  - [L. Di, et. al., 2006], [P. Yue, et. al., 2010]
  - User has to manually annotate the services according to an ontology like OWL-S
  - They model input and output types but not relationship among them

• **Linked Open Services (LOS)**
  - [B. Norton, R. Krummenacher, 2010]
  - Services that consume linked data as input and also return linked data as output
  - Use SPARQL to describe service inputs and outputs
  - Describing services might be easy for Linked Data community, but not for average Internet users

• **Google Fusion Tables**
  - [H. Gonzalez, A. Halevy, et al. 2010]
  - Import data from various source types and invoke web services
  - Allows advanced visualization
  - Integrating data from different sources is possible but without exploiting semantics
• Karma allows users to quickly and easily dynamically fuse a wide variety of geospatial data sources

• Modeling geospatial services is a big step in geospatial data fusion

• Based on provided ontology, Karma semi-automatically builds a semantic model of reasoning services including both input/output datatypes and their relationships

• Semantic descriptions enable user to easily find the desired service and invoke it using available data sources
Future Work

- Applying the service modeling techniques to available REST web services
  - Create the service model just based on service invocation samples
- Answer queries like “Can I have the street names of the cities whose distance to Los Angeles is less than 50 miles?”
  - Automatically compose available web services using loaded data sources
- Publishing semantic description of web services in formats such as LOS