

Using an Ontology-based Approach for Geospatial Clustering Analysis

Xin Wang

Department of Geomatics Engineering
University of Calgary
Calgary, AB, Canada T2N 1N4
xcwang@ucalgary.ca

Abstract. Geospatial clustering is an important method for geospatial information analysis. However, most clustering methods do not consider semantic information during the clustering process. In this paper, we present a formal geospatial clustering ontology framework, which can provide the background for geospatial clustering. Using the ontology, geospatial clustering can become a knowledge-driven process.

1. Introduction

Geospatial clustering is an important topic in knowledge discovery and geospatial information systems (GIS) research. It aims to group similar objects into the same group (called *cluster*) based on their connectivity, density and reachability in space. It can be used to find natural clusters (e.g., extracting the type of land use from the satellite imagery), to identify hot spots (e.g., epidemics, crime, traffic accidents), and to partition an area based on utility (e.g., market area assignment by minimizing the distance to customers).

In geospatial analysis, points nearby to a point usually influence more than the points farther away. It is not the only factor we should consider during the geospatial clustering. Domain knowledge and users' requirements during the geospatial clustering also plays important roles. For example, users may be interested in finding the clusters in Western Canada based on the population density or they would like to find the clusters based on the traffic reachability. Different clusters could be discovered based on the two different goals. However, existing clustering methods and clustering process are solely focusing on data itself without considering these factors. Thus, clustering occurs at the data level instead of the knowledge level, which prevents users from precisely identifying their targets and understanding the clustering results.

Although some existing clustering methods consider various constraints, they only consider sharply limited knowledge concerning the domain and the users, for example, users might want to discover clusters from a wild animal habitat dataset with the goal that each bird should be in the same cluster with its nest. The constraint is at instance level and clustering method does not help users to specify the constraint. Users need to specify their goal (bird and nest) at instance (data) level instead of knowledge level.

In addition, non-spatial attributes can contain domain-related information and lead to meaningful results in geospatial clustering, but most geospatial clustering methods have no or very restricted consideration on them, for example, the INCOME attribute might be considered if we need to find population clusters to identify the best location for a retail store targeting the middle-class people.

Thus, how to incorporate domain knowledge in the geospatial clustering methods and clustering process becomes an important topic in geospatial analysis research. A more sophisticated and systematic framework is needed to support semantics in geospatial clustering.

An *ontology* is a formal explicit specification of a shared conceptualization [5]. It provides domain knowledge relevant to the conceptualization and axioms for reasoning with it. Especially in the domain of geographical, the clustering usually highly depends on the geography features. In the following section, we will discuss a framework GEO_CLUST which integrates the ontology in geospatial clustering.

2. Geospatial Clustering Ontology

The data mining research process can be seen as aspects of three phases: understanding the problem, understanding the data, and performing data processing[8]. Right now, clustering is regarded as occurring in the third phase, data processing, which purely operates on data. Arguably, the most appropriate clustering algorithm should be selected after taking into account factors such as the user's goal, relevant domain-specific knowledge, characteristics of the data, and available clustering algorithms. However, if queries were posed to the user about those factors in an arbitrary manner, it would be confusing.

An ontology is an explicit representation of knowledge. It is a formal, explicit specification of shared conceptualizations, representing the concepts and their relations that are relevant for a given domain of discourse [5]. It consists of a representational vocabulary with precise definitions of the meanings of the terms of this vocabulary plus a set of axioms. An ontology can provide a systematic way of organizing these factors such that they can contribute to the selection process and an orderly description of this process to the user.

In this paper, we propose a general geospatial clustering ontology. It includes five main classes: `ClusteringTask`, `GeospatialEntity`, `GeospatialRelation`, `GeospatialData`, and `ClusteringMethod`.

1) `ClusteringTask` is an abstract class. It is the superclass of all possible spatial clustering tasks that users may perform, including `FindHotSpotsTask` and `PartitionIntoClustersTask`. Each type of clustering task is connected to some classes of clustering algorithms. Based on the purpose of the clustering and the domain, domains, an appropriate clustering algorithm and dataset is selected. For example, two tasks are finding the best locations for shopping malls based on the population density and finding the best locations for shopping malls based on transportation convenience. According to our spatial clustering ontology, the former task should operate on population data with a density-based clustering method, and the latter task should operate on transportation data with a partitioning clustering method.

2) `GeospatialEntity` is an abstract class. Its subclasses provide the basic classes of spatial-related concepts or entities. In our spatial clustering ontology, the `GeospatialEntity` class includes three subclasses, `GeometricThing`, `Place`, and `Border`. For each class, attributes and constraints are also defined.

Because in geospatial clustering, all data are represented as geometric shapes for processing, the `GeometricThing` class includes all the kinds of shapes known to be relevant to clustering. Under `GeometricThing`, two subclasses are included: `AbstractShape` and `Angle`.

Under the class `Place`, we have four subclasses, `ContactLocation`, `GeographicalRegion`, `EcologicalRegion` and `Planet`. Under `GeographicalRegion`, we have the `LandBody` and `BodyofWater` subclasses. `Continent`, `Country`, `Province/State`, and `City` belong to the `LandBody` class. `Sea`, `Gulf`, `Stream`, `Harbor`, and `Lake` are included under `BodyOfWater`.

3) `GeospatialRelation` represents a spatial relation among the objects in `GeospatialEntity`. The three kinds of spatial relations are direction relations, distance relations, and topological relations. Some examples of direction relations are north, south, up, down, behind, and front. Some examples of distance relations are far and close-to (near). Some examples of topological relations are contain, overlap, and meet.

4) `GeospatialData` represents the properties of the spatial data that is registered for the web service. Basic properties of the class include types of the data, formats of the data storage, subjects of the data, and the general location described by the data. The data type could be raster and vector data. Example formats are Access database, text file, and XML. Any datasets used in clustering is an instance of `GeospatialData`.

5) `ClusteringMethod` represents a list of all available clustering methods and their features. As shown in Figure 1, the methods are classified based on clustering techniques as hierarchical methods, partitional methods, density-based methods, and grid-based methods. For the hierarchical methods, it can be either agglomerative or divisive. Every method is connected with some geospatial clustering tasks. The attributes of the clustering methods, such as the parameters required for the method, and the shape of the clusters generated by the method.

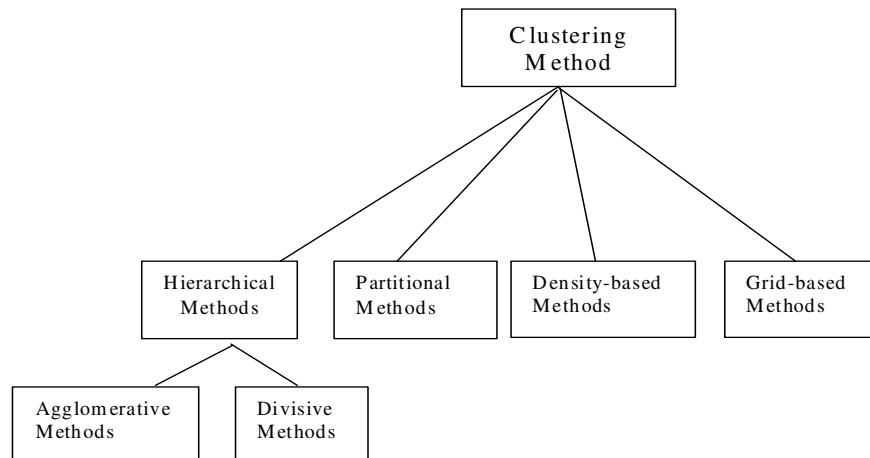


Figure 1. A Hierarchical View of the `ClusteringMethod` Class

3. Ontology-based Geospatial Clustering

Based on the above analysis, we propose a framework called GEO_CLUSTER for ontology-based clustering, as shown in Figure 2. In GEO_CLUSTER, the *geospatial clustering ontology* component is used when identifying the clustering problem and the relevant data. Within this component, the *task ontology* specifies the potential methods that may be suitable for meeting the user's goals, and the *domain ontology* includes all classes, instances, and axioms in a geospatial domain. A domain ontology could be built by users or domain experts, or derived from existing ontologies.

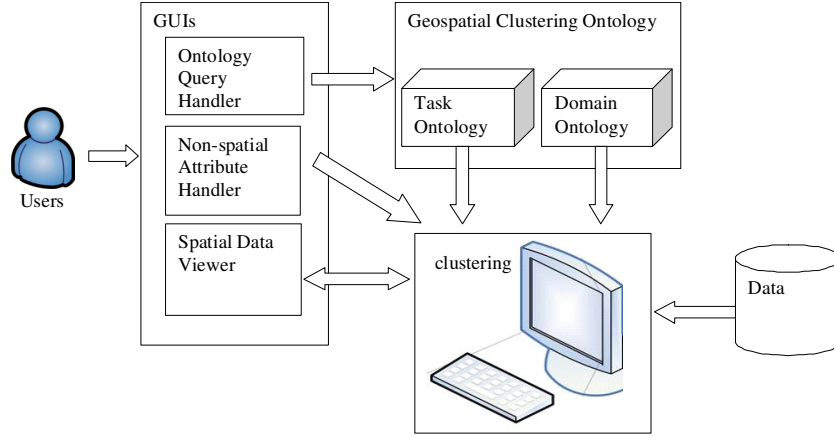


Figure 2. The GEO_CLUSTER framework of ontology-based clustering

Ontology Query Handler will include a GUI and a plug-in of Protégé-OWL [12]. It will be used to support rule-based reasoning for spatial clustering ontology. Java-based rule engine such as JESS [15] will be used for development. The query will provide the result including appropriate clustering methods and datasets together with possible explanation data.

The Spatial Data Viewer will allow users to effectively explore and select appropriate data relevant to user's task. The basic functions of Spatial Data Viewer will include visualize data in different scales, locate interesting spots and areas, select extra data layers, and parameter tests. Spatial Data Viewer will also link the data with metadata, which contains the information about datasets. The clustering results will be represented through the Spatial Data Viewer as well.

Non-spatial Attribute Handler will allow user to tailor their own way to handle non-spatial attributes in the data. Users can define the similarity function and purity function for non-spatial attributes based on their goals. The functions will be integrated with the new clustering methods.

The framework provides a template for performing geospatial clustering using the following steps. First, the user's goal is translated into queries that perform reasoning on the ontology. Relevant algorithms and geospatial data sets are selected and instantiated from the ontology with respect to the user's goal. Secondly, the selected clustering algorithm performs clustering based on the results produced from queries. Finally, the results are explained through the ontology. We assume the geospatial clustering ontology is represented in a web ontology language.

With the framework, users first give their goals for clustering. The goals are translated into the ontology query language and matched with task instances in the task ontology. The goals are also used to search the domain ontology. The results of these queries identify the proper clustering methods and the appropriate datasets. Based on these results, clustering is conducted. The clustering result can be used for statistical analysis or it can be interpreted using the task ontology and the domain ontology. The final result is returned to the user in an understandable format.

4. Application to Canadian Population Data

In this section, we first describe how we materialized the above high-level spatial ontology to represent knowledge in the application to Canadian population data. Then we present an example showing how to reason using the ontology to facilitate spatial clustering for Canadian population data.

The data used in the application is the Canadian populations of geographic areas from the 1996 Census of Canada. The population and dwelling counts are provided by individual postal codes. The postal codes were transformed to longitude and latitude using GeoPinPoint [13].

Because OWL can formally represent the meaning of the domain terminology and it allows performing useful reasoning tasks on these documents, we used it as the language to represent the ontology. We use the OWL Plugin [14] of Protégé-2000 to construct the spatial clustering ontology.

The current ontology has 51 classes, including the high-level classes mentioned in Section 2 and some low-level classes to help building properties or relationships for high-level classes. For example, `GeospatialData` is a high-level class with a property named `DataFormat`, which is used to describe potential formats for the data. Since the formats cannot be represented by any of the standard data types in OWL, we created a low-level class called `DataFormat` in the ontology.

Over one hundred instances exist in the ontology. It includes all Canadian provinces plus major cities, rivers, and lakes. For example, Saskatchewan is an instance of class `Province`, and Regina is an instance of class `City`.

Suppose the user's goal is "to find the population clusters/groups of western Canada". Without geographic knowledge of Canada, the traditional clustering algorithm cannot proceed due to the lack of the definition of "western Canada". In the following discussion, we will use this example to explain how the knowledge is represented in the ontology and how the reasoning is being done to help find the proper databases and clustering methods.

Since in the geospatial ontology, all provinces and major cities are represented as instances of the `Province` and `City` classes, respectively. Other geographical units, such as western Canada, are also defined. Figure 3 shows the OWL representation for "Western Canada", an instance of `GeographicalRegion`, produced by the OWL Plugin in Protégé-OWL. It shows that 'Western Canada' is a geographical region inside 'Canada' (which is an instance of `Country`). It includes four provinces: Alberta, British Columbia, Manitoba, and Saskatchewan, each of which is an instance of `Province`.

In OWL, the relations among classes can be defined as properties. In this example, the properties can be classified into two kinds. The first kind of properties concerns pure spatial relations, such as *eastOf*, *farAway*, and *inside*. The “include” relationship in Figure 3 is defined as a property in OWL. The second kind of properties are used to describe the properties or attributes of a class. For example, *hasName* is used to define the names of instances of each class.

```

<GeographicalRegion rdf:ID="westerncanada">
  <hasName>Western Canada</hasName>
  <inside>
    <Country rdf:ID="canada">
      <hasName>Canada</hasName>
    </Country>
  </inside>
  <include>
    <Province rdf:ID="alberta">
      <hasName>Alberta</hasName>
    </Province>
  </include>
  <include>
    <Province rdf:ID="britishcolumbia">
      <hasName>British Columbia</hasName>
    </Province>
  </include>
  <include>
    <Province rdf:ID="manitoba">
      <hasName>Manitoba</hasName>
    </Province>
  </include>
  <include>
    <Province rdf:ID="saskatchewan">
      <hasName>Saskatchewan</hasName>
    </Province>
  </include>
</GeographicalRegion>

```

Figure 3. OWL representation for Western Canada

After building the classes and instances in the spatial clustering ontology, the next question is how to reason about the OWL representation of the ontology. We use the above example ontology to illustrate how an ontology query could be processed. In the ontology, we have an instance of *GeographicalRegion* called ‘Western Canada’ as shown above. It includes four province instances: Alberta, British Columbia, Manitoba, and Saskatchewan. Each province instance is inside of Western Canada, as shown in Figure 4. The ‘include’ and ‘inside’ relations are defined as inverse properties. For example, as shown in Figure 5, Alberta is an instance of *Province* class. It is inside Western Canada and east of British Columbia. It is close to Saskatchewan and British Columbia, and it overlaps with the North Saskatchewan River. Two cities, Calgary and Edmonton, are inside the province.

Each instance of the *SpatialData* class connects to the instance of the *GeographicalRegion* through the property of ‘aboutWhere’ and to the instance of spatial format through property of ‘format’. Figure 6 shows that a dataset called

'abpopdb' is available in Access database format. It contains population data for the province of 'Alberta'.

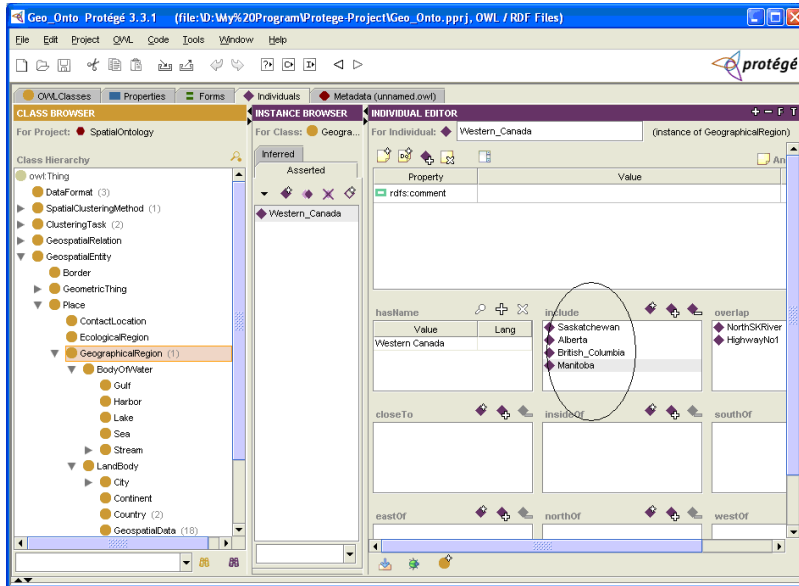


Figure 4. Western Canada is an instance of GeographicalRegion

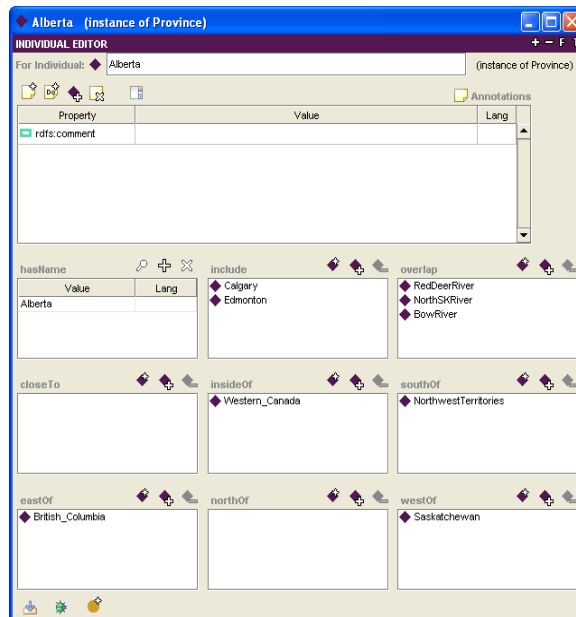


Figure 5. Alberta is part of Western Canada

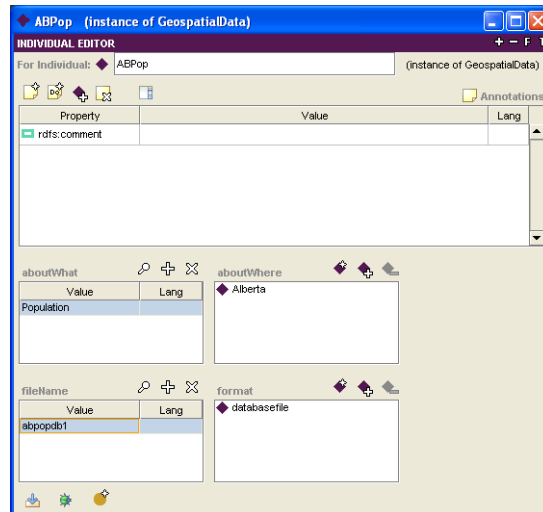


Figure 6. abpopdb is an instance of SpatialData

To reason in Protégé, we use a JessTab plug-in [15], which integrates Protégé with Jess, a fast rule engine and scripting environment. For the user's goal given above, we create a Jess query (or rule) as follows:

```
(defrule query1
  (object(is-a ClusteringMethod)(hasName ?method_name)
    (forGeneralPurpose "Yes"))
  (object (is-a GeographicalRegion) (OBJECT ?gr1)
    (hasName "Western Canada") )
  (object (is-a Province) (OBJECT ?pr) (inside ?gr1))
  (object (is-a GeospatialData) (hasName ?name)(
    aboutWhat "population")(aboutWhere ?pr))
=> (printout t ""?method_name" can be used on "?name", which is a dataset about
populations in western Canada" crlf))
```

The result of running the Jess query indicates that four databases, i.e., abpopdb1, bcpopdb1, mnpopdb1, and skpopdb1, could be used as datasets for the clustering on the populations of western Canada. Five available clustering methods, including STING[10], K-means[11], DBSCAN[4], CLARANS[6], and AUTOCLUST[3], can be used to accomplish this general-purpose clustering task. Obtaining this type of result is a simple form of applying reasoning to the ontology.

Currently we do not have a spatial data viewer component which can easily integrate with Protégé. Figure 7 shows the clustering result when we pick DBSCAN as the clustering method and use ArcGIS to deploy the clustering result on the map. The resulting clusters are matched with the locations of major cities or geographical area in the ontology, and then we can explain the clustering results. As shown in Figure 7, each cluster is represented by the cities or geographical areas (if the cluster includes more than one major cities) and the number of points in the clustering.

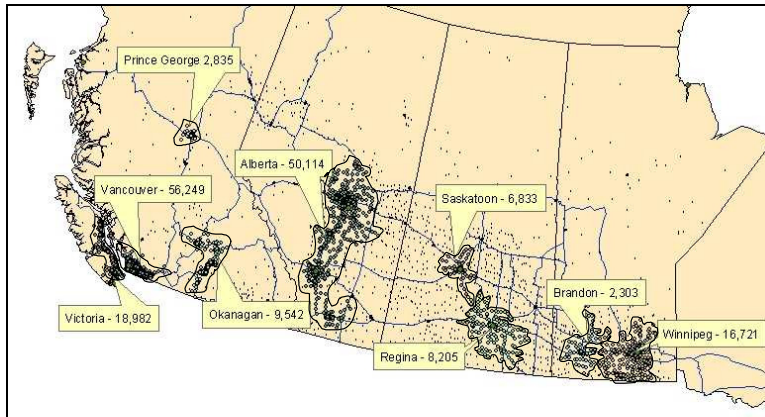


Figure 7. Result of Clustering Population Counts of Western Canada

5. Conclusions

In this paper, we presented GEO_CLUSTER, a framework for ontology-based geospatial clustering. In the framework, geospatial clustering can be conducted with the support of a geospatial clustering ontology. The ontology can play an important role in organizing information related to the process of clustering.

This paper focused on building the geospatial clustering ontology and simple reasoning on it. The existing framework needs to be extended with regard to its capabilities and its flexibility. Currently, a more sophisticated spatial query handler and spatial data viewer generator are under development.

References:

- [1] Agrawal, R., Gehrke, J., Gunopulos, D., and Raghavan, P.: Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications, *SIGMOD Record*, 27(2) (1998) 94-105
- [2] Berners-Lee, T., Hendler, J., and Lassila, O.: The Semantic Web. *Scientific American* 284(5) (2001) 34-43
- [3] Estivill-Castro, V., and Lee, I. J.: AUTOCLUST+: Automatic Clustering of Point-Data Sets in the Presence of Obstacles. In: *Proc. of Intl. Workshop on Temporal, Spatial and Spatio-Temporal Data Mining*, Lyon, France (2000) 133-146
- [4] Ester, M., Kriegel, H., Sander, J., and Xu, X.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, *Proc. of 2nd KDD*, Portland (1996) 226-231
- [5] Gruber, T. R.: A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2) (1993) 199-220
- [6] Han, J., Lakshmanan, L. V. S., and Ng, R. T.: Constraint-Based Multidimensional Data Mining. *Computer* 32(8) (1999) 46-50
- [7] Ng, R., and Han, J.: Efficient and Effective Clustering Method for Spatial Data Mining, *Proc. of Int'l Conf. on Very Large Data Bases*, Santiago, Chile (1994) 144-155

- [8] Sund, R.: Utilisation of administrative registers using scientific knowledge discovery. *Intelligent Data Analysis*, 7(6) (2003) 501-519
- [9] Tung, A. K. H., Han, J., Lakshmanan, L. V. S., and Ng, R. T.: Constraint-Based Clustering in Large Databases. In *Proc. 2001 Intl. Conf. on Database Theory*, London, U.K. (2001) 405-419
- [10] Wang, W., Yang, J., and Muntz, R.: STING: A Statistical Information Grid Approach to Spatial Data Mining, *Proc. of 23rd VLDB*, Athens, Greece, (1997) 186-195
- [11] Witten, I.H., and Frank, E.: *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann (2000)
- [12] <http://protege.stanford.edu/index.html>
- [13] http://www.dmtispatial.com/geocoding_software.html
- [14] <http://protege.stanford.edu/plugins/owl/index.html>
- [15] <http://www.ida.liu.se/~her/JessTab/>