BigDataVoyant: Automated Profiling of Large Geospatial Data

Pantelis Mitropoulos
Geospatial Enabling Technologies
Greece
pmitropoulos@getmap.gr

Kostas Patroumpas
Athena Research Center
Greece
kpatro@athenarc.gr

Dimitrios Skoutas
Athena Research Center
Greece
dskoutas@athenarc.gr

Thodoris Vakkas
Geospatial Enabling Technologies
Greece
tvakkas@getmap.gr

Spiros Athanasiou
Athena Research Center
Greece
spathan@athenarc.gr

ABSTRACT

We envisage an open, extensible, and scalable data profiling framework over various types of *geospatial* data, including vector, raster and multidimensional assets. In this paper, we outline our work in progress regarding the design and implementation of *BigDataVoyant*, a software platform for profiling big geospatial data. This software is able to ingest data in various spatial formats and reference systems. Its main goal is to extract and visualize a large variety of metadata and descriptions about data quality and characteristics both in an interactive as well as in a fully automated manner. We suggest a processing flow for such profiling and discuss a preliminary, yet comprehensive list of metadata items already supported by the open-source software prototype we are implementing. Finally, we outline open issues and extensions of the proposed framework to broaden its usefulness and strengthen its appeal to the geospatial data community.

1 INTRODUCTION

As the availability, the volume, and the variety of data from different sources grows, it is crucial for stakeholders to assess its relevance and suitability for a given type of analyses, applications or services. *Data profiling* comprises a collection of operations and processes for *extracting metadata* from a given dataset and thus facilitating decision making on its potential utilization. Such metadata may involve schema information, statistics, samples, or other informative summaries over the data, thus offering extensive and objective indicators for assessing datasets in terms of business value, fitness for purpose, and quality. In addition, a variety of *visualizations* (tables, charts, graphs, timelines, etc.) may be applied against this metadata to convey the significance and interpret the value of the underlying data.

In proprietary data catalogues, organizational data lakes, or scientific repositories, each containing numerous and perhaps heterogeneous data assets, offering users with search capabilities against rich collections of their extracted metadata can greatly facilitate *data discovery*. Exploring the schema, semantics, and actual contents of open data repositories through such profiles can also evaluate their usefulness in *data integration* tasks. For data traded in *marketplaces*, potential consumers can examine such profiles (especially any quality indicators) on the available datasets, compare them, and then determine whether to purchase and which one(s) to choose.

This is all the more important for *geospatial* data, represented not only as vectors, but also in raster or multi-dimensional formats. In the context of the *OpertusMundi* project[1], we build a trusted, robust, and scalable pan-European geospatial data marketplace. Metadata extracted from such datasets can indicate their coverage, timeliness, consistency, and completeness, along with other domain-specific properties (topology, scale, resolution, etc.) crucial for the entire lifecycle of spatial asset provision, discovery, sharing, purchasing, and use.

Data profiling as a means of exploring, analyzing, and interpreting big data assets has attracted a lot of interest over the past decade from researchers and practitioners alike. Challenges and important use cases have been presented in recent surveys [1, 9], which however focus on relational data, overlooking the specific requirements of geospatial data. Among the main challenges recognized is the ability to handle input data from heterogeneous sources and to deal with the computational complexity and scalability of profiling functionalities. Interpreting the output profiles is also challenging, since it typically requires domain expertise and may depend on the use case (e.g., exploration, integration, cleansing). Another recent survey [3] focuses on methods and systems that enable automatic indexing and interactive searching over large collections of datasets that fit users' needs. Discovering *dependencies* between attributes in relational data has also led to interesting techniques. For instance, algorithms for simultaneously identifying unique column combinations, inclusion and functional dependencies [6], as well as order dependencies [4] have been proposed. Prototype systems have also been developed for data profiling. *Data Civilizer* [5] extracts profiling signatures and creates a metadata graph to facilitate discovery of joinable datasets or those relevant to user tasks. The *GOODS* platform [8] can also infer provenance metadata and annotation tags to enable efficient and scalable discovery of datasets. Moreover, modern platforms for data science tasks like Kaggle[2] or digital marketplaces like Dawex[3] include data profiling mechanisms, as well as keyword-based search for data discovery.

Admittedly, the aforementioned generic schemes lack inherent capabilities for *geospatial data profiling*. Yet, GIS software platforms already support Exploratory Spatial Data Analysis (ESDA). ESDA employs spatial mining and analysis tools [2, 10] that allow users to visualize spatial distributions, identify outliers, discover patterns like clusters or hot spots, etc. Such tools are widely used in full-fledged GIS platforms like ArcGIS[4] and QGIS[5] or

[1] https://www.opertusmundi.eu/
[2] https://www.kaggle.com/
[3] https://www.dawex.com/en/
[4] https://www.esri.com/arcgis/
[5] https://qgis.org/

geospatially-aware DBMS like PostGIS[6] or Oracle Spatial and Graph[7]. Although powerful in capabilities, data profiling in GIS is not streamlined, but may involve manual execution of a series of operations. Sometimes, a step-by-step user intervention is necessary: invoke a given functionality (e.g., heatmap creation), specify parameters, map rendering options, etc. The cost of purchasing a software license may also be a hindrance, as well as usage or programming skills.

In contrast, we propose *BigDataVoyant*, an open-source, interactive, modular, and extensible framework specifically tailored for in-depth profiling of geospatial data. This platform aims to extract metadata from different types of spatial assets (vector, raster, multidimensional data) and allow data scientists to control its *degree of automation*. In its interactive, step-by-step mode, the analyst configures each operation, inspects its results, and optionally invokes it again with revised settings for advanced, detailed data exploration. In a fully automated mode, when triggered by inexperienced users or involved in a broader processing workflow (e.g., pipeline), a default or a user-preconfigured parametrization may be applied to run profiling as a batch job. In either mode, metadata items become progressively available and can be graphically visualized, thus expediting spatial data exploration. With BigDataVoyant, we aim to support efficient and robust profiling of large geodatasets with modern processing schemes (data partitioning, distributed processing) and provide it as a service with a RESTful API. Overall, we deem that the rich and extensible collection of automated metadata generated by BigDataVoyant will offer a powerful and comprehensive means of assessing quantity, quality, and variety in spatial data assets for mission-critical applications and services.

2 PROFILING FRAMEWORK

As illustrated in Figure 1, we envisage a data profiling framework that can handle large geospatial datasets of different types. We propose to automatically extract metadata not only from vector or raster data assets available in a variety of sources and formats (files, DBMS, or WFS services), but also from multidimensional scientific data having spatial reference (like meteorological, hydrological, or other sensor measurements). The *profiler engine* is the core processing component. Using an extensible set of software libraries and APIs, it can ingest a geospatial dataset and apply a set of processing tasks. Each task computes metadata according to a configuration specified by the analyst or automatically determined by the system, concerning the target metadata and parameter settings for the task. The resulting *metadata* include statistics in JSON format, geospatial features in Well-Known Text or Binary (WKT/WKB) representations, maps as PNG images, etc. All extracted metadata is stored in a *repository* to be used for dataset search and reporting. This metadata are also interactively visualized in a *dashboard* as lists, graphs, charts, and maps of various types, uncovering latent patterns, trends, and even issues (e.g., outliers, inconsistent or missing values) with the data. Employing a *human-in-the-loop* paradigm, the data scientist can steer or fine-tune the execution by choosing specific metadata items of interest and adjusting their parameters in order to delve into more detailed inspection of data characteristics.

Next, we present a non-exhaustive list of metadata items that can be automatically extracted from several types of geospatial data. We also discuss how such metadata can be visualized as lists, maps, charts, etc.

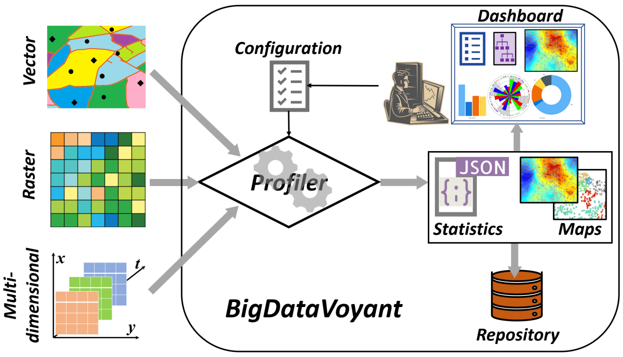


Figure 1: Processing flow for automated geodata profiling.

Table 1: Automated metadata computed over vector data

Metadata	*Description*	*Scope*
Native CRS	Coordinate reference system (EPSG)	Geometry
Spatial extent	Rectilinear MBR covering all features	Geometry
Feature count	Number of rows (records)	Dataset
Convex Hull	Convex polygon enclosing all features	Geometry
Concave Hull	Concave polygon enclosing all features	Geometry
Attribute names	List of column names of all attributes	Thematic
Attribute types	List of data types of all attributes	Thematic
Cardinality	Count of *NOT NULL* values per attribute	Thematic
Value pattern	Semantic domain of values	Thematic
Value distribution	Frequency histogram of attribute values	Thematic
N-tiles	Values dividing data into N equal parts	Thematic
Distinct values	List of categorical values	Thematic
Frequent values	top-k most frequent categorical values	Thematic
Attr. dependencies	Key, functional, conditional dependencies	Thematic
Heatmap	Colormap with varying intensity	Geometry
Clusters, outliers	Spatial clusters of features (e.g. POIs)	Geometry
Samples	Data portion(s) of limited size/extent	Dataset

2.1 Profiling of Vector Data Assets

Vector datasets may include thematic attributes, whereas geometry is typically stored in WKT, WKB, or BLOB. Profiling of vector data can be applied in different fashions to automatically compute metadata regarding the entire data asset, its spatial attribute (geometry), or its thematic attributes, as listed in Table 1. Next, we briefly outline each category of such metadata.

2.1.1 Entire dataset. This examines the vector data as a whole. For instance, it may involve computation of simple numerical values (e.g., feature count) or extraction of data *samples*, either for the entire area or a user-specified one, e.g. to allow the analyst or potential customer to examine data quality for an area of their interest before using or purchasing an asset.

2.1.2 Spatial attributes. These are automatically computed against a geometry attribute, usually projected according to a known Coordinate Reference System (CRS)[8]. The *spatial extent* is captured by the Minimum Bounding Rectangle (MBR, BBOX) of vectors, but is often insufficient or misleading due to the "dead space" induced by its rectangular shape. Thus, the convex and concave hulls of the geometries may provide further insight, allowing users to illustrate and inspect them together with the

[6] https://postgis.net/

[7] https://www.oracle.com/database/technologies/spatialandgraph.html

[8] Typically defined according to the EPSG registry: https://www.epsg.org

Table 2: Automated metadata computed over raster data

Metadata	*Description*	*Scope*
Native CRS	Coordinate reference system (EPSG)	Dataset
Spatial extent	Rectangle with the image bounds	Raster
Resolution	Pixel size (e.g., in meters) per axis	Raster
Width, Height	Number of pixels per axis	Raster
Bands	Count of the available bands	Raster
Value distribution	Histogram(s) of raster values	Band
Band statistics	Min, max, avg, stdev, etc. of raster values	Band
COG	Boolean: is Cloud Optimized GeoTIFF?	Raster
Pixel (bit) depth	Integer denoting the range of values stored	Band
NoData	Special value(s) denoting NoData pixels	Band
Samples	Custom portion(s) by user-specified box	Raster

Table 3: Automated metadata over multidimensional data

Metadata	*Description*	*Scope*
Native CRS	Coordinate reference system (EPSG)	Dataset
Dimension count	Number of dimensions in the dataset	Dataset
Dimension info	Name, length, etc. of each dimension	Dataset
Variable count	Number of variables in the dataset	Dataset
Variable info	Name, type, shape, attributes per variable	Dataset
Spatial extent	Range of values per spatial dimension	Variable
Temporal range	Timespan of stored values per variable	Variable
Value distribution	Histogram of values per variable	Variable
NoData	Special value(s) denoting NoData	Variable
Samples	Custom portion(s) by user preferences	Dataset

MBR and better assess the spatial coverage of a given asset. Examining the *spatial data distribution* may be even more informative. Features like clustering (e.g., using DBSCAN [7]) or heatmaps, computed over the geometries can reveal concentration, density and other spatial patterns [10] in the underlying features, e.g., hotpots for certain POI categories like bars or shops.

2.1.3 Thematic Attributes. For this part, relational data profiling can be applied [1]. A list of the available attributes and their data types (e.g., string, integer, double, date, timestamp, etc.) can illustrate the schema. Further, for each thematic attribute, the number or percentage of missing values, and its *cardinality* (i.e., the number of distinct values) can be computed. When depicted in a histogram or a chart, they can illustrate the quality of data across all attributes and possibly even indicate correlations amongst them. Classification of the *semantic domain* for certain attributes (e.g., phone numbers, web pages, names, etc.) may be also examined. This requires knowledge of specific patterns that can be progressively collected from the corpus of data assets handled by the profiler.

For a particular thematic attribute, extra processing can provide the *distribution* of its values, depicted as frequency histograms (equi-width, equi-depth, V-optimal, etc.). For numerical attributes, N-tiles (e.g., quartiles) may be also provided. For a categorical attribute (e.g., POI categories or road classifications), lists of the *distinct* or *most frequent values* may further convey its representativeness.

Discovery of *attribute dependencies*, such as key or functional dependencies, concerns multi-attribute profiling [6]. Such features may indicate that values in one set of columns functionally determine the value of another column, e.g., specific POI categories like pharmacies always include phone numbers for emergency calls. Other dependencies may be conditional, e.g. street names are unique for each city or postal code.

2.2 Profiling of Raster Data Assets

Raster data includes digital aerial photographs, satellite imagery, scanned maps, etc., typically organized in one or more *bands*. For example, a color RGB image has three bands (Red, Green, Blue), a digital elevation model (DEM) just one holding elevation values, whereas a multispectral image may have many bands. Table 2 lists some automated metadata that may be extracted from rasters.

Certain metadata items are similar to those proposed for vector data (e.g., spatial extent, native CRS, sampling). However, since rasters are tessellations of a 2-d surface into cells (pixels), raster profiling tools require different functionality from that applied against vectors. Besides, some information may not be always possible to calculate. For instance, native CRS for a TIFF image may be unknown if its accompanying .tfw file does not include it in its header. Spatial extent (MBR, BBOX) may be deduced from the original raster without reprojection. If the native CRS is known, raster resolution can be accompanied with units of measurement.

Raster-specific metadata concerns resolution, image size, as well as whether the imagery is COG (i.e., Cloud Optimized GeoTIFF). Information about *NoData* values is also important; this is typically a specially designated value (e.g., None, -999) to indicate missing information and could differ per band.

Finally, *band-related* metadata concern the number of bands, the data type/depth of each one, and statistics (min, max, avg, stdev, etc.) of values per band. A default histogram of cell values per band can give a more detailed insight about raster quality.

2.3 Profiling of Multidimensional Data Assets

Multidimensional data is typically used to store environmental, climate, or meteorological information with measurements for multiple *variables*, with NetCDF[9] its most common representative. *Dimensions* (each with a name and a length) are used to represent real physical dimensions, such as time, latitude, longitude, or elevation, and define the structure of data. *Variables* (each specified with a name, a type, a shape, and often accompanied by their units) hold multi-dimensional arrays of values of the same type. Profiling of such data assets involves some automatically extracted metadata that are common with vector and raster data, but due to the inherent complexity in this data format, it requires specialized handling. Table 3 lists automated metadata extractable from multidimensional data assets, with some of them concerning the entire dataset (e.g. native CRS) while others are computed over individual variables.

Histograms (one per variable) may be created in order to visualize the *distribution* of the values. In addition, special attributes (such as “missing_value” and “_FillValue” in netCDF files) may be used to identify missing or undefined data; these values should be treated like the *NoData* ones in raster layers. It may also be possible to extract metadata features depending on the type of the variables in the dataset. For example, considering the time dimension, identify the time granularity of data per variable, any gaps in measurements, etc. Such information is valuable for potential users in order to assess the coverage, timespan, and completeness per variable of interest in such multidimensional data. Finally, extraction of ad-hoc *samples* according to user preferences, such as a rectangular area of interest or an interval on a given dimension (e.g., a time period), should be also possible.

[9] https://www.unidata.ucar.edu/software/netcdf/

3 IMPLEMENTATION STATUS & OUTLOOK

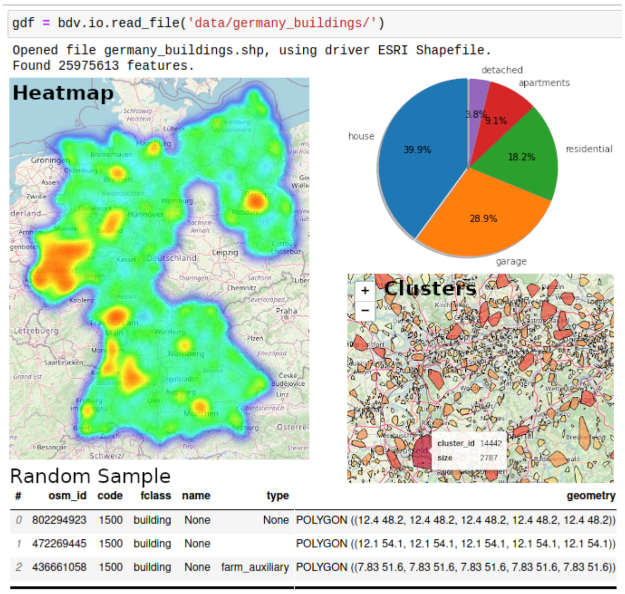
In the context of the OpertusMundi project, we are developing an open-source prototype for BigDataVoyant[10]. Current functionality mostly concerns its profiler engine, which leverages various Python libraries to offer an API for automated metadata extraction and manipulation. Python offers a robust, user-friendly environment for data scientists, and its ecosystem includes many popular libraries for geospatial data processing and visualization. Existing profiling tools[11] in Python are mostly based on Pandas[12] and unfortunately do not support geospatial data. We considered extending them, but Pandas frames do not scale for large datasets. Besides, its spatial extension GeoPandas can only handle vectors and its performance over complex geometries is slow.

As scalability is a major concern in BigDataVoyant, we opted for another solution by extending the memory-efficient Vaex library[13] with spatial capabilities. Presently, we actually support all types of 2-dimensional vector geometries, like (multi)points, (multi)linestrings, (multi)polygons, collections, etc. according to OGC specifications[14]. At a later stage, we plan to handle vector data with $d > 2$ dimensions, e.g., road segments with linear referencing. If necessary, we will also consider extra metadata features for particular applications, e.g., examine whether a road network is connected and can support routing. On top of this GeoVaex extension[15] we have been building for vector data, a beta version of our profiler can automatically extract most of the metadata in Table 1 for moderately large datasets. Further, we employ GDAL/OGR[16] to support raster data profiling (Table 2) for a wide range of file formats, and netCDF4 Python module[17] for multidimensional data profiling (Table 3).

Parameters like number of clusters or histogram buckets, weight and radius for heatmaps, etc., are currently configured manually. We will examine whether this could be automated in a data-driven manner according to data characteristics (e.g., size, spatial extent). Besides, some methods are applicable on specific geometry types (e.g., concave hulls on points only), so such data conversions are handled in the background without user intervention.

Of course, scalability against very large spatial datasets is the main challenge. Although the libraries employed in profiling can generally scale to big datasets, computation of certain metadata requires particular handling, especially where some kind of spatial aggregation is involved. For instance, special algorithms are needed to treat alpha shapes in the whole dataset, compute heatmaps or clusters, etc., when data partitioning is applied to boost computation efficiency. Initial tests of our approach employing GeoVaex verify its strong potential in terms of scalability and efficiency. Figure 2 exemplifies a subset of metadata automatically computed over a large vector dataset extracted from OpenStreetMap with more than 25 million polygons in Germany, and visualized in a dashboard. Once the software reaches a stable level, we plan an extensive empirical study against real-world geodatasets of various types and formats.

Data cleansing and detection of outliers is also challenging, mostly regarding the spatial aspect (geometries, pixels) in large data collections. We intend to use clustering results and seclude features that stand out, accompanied by an '*outlier confidence weight*', which will be apparently affected by the parameter values set in the algorithm. We deem that this will offer data scientists and stakeholders a precious tool to identify possible errors in the data, refine their analyses, and improve quality in their products.



```
gdf = bdv.io.read_file('data/germany_buildings/')
```

```
Opened file germany_buildings.shp, using driver ESRI Shapefile.
Found 25975613 features.
```

Random Sample

#	osm_id	code	fclass	name	type	geometry
0	802294923	1500	building	None	None	POLYGON ((12.4 48.2, 12.4 48.2, 12.4 48.2, 12.4 48.2, 12.4 48.2))
1	472269445	1500	building	None		POLYGON ((12.1 54.1, 12.1 54.1, 12.1 54.1, 12.1 54.1, 12.1 54.1))
2	436661058	1500	building	None	farm_auxiliary	POLYGON ((7.83 51.6, 7.83 51.6, 7.83 51.6, 7.83 51.6, 7.83 51.6))

Figure 2: Visualization of automated metadata profiling a vector dataset of 25 million polygons (buildings in Germany) extracted from OpenStreetMap.

Finally, this spatial data profiling is extensible thanks to the modular design of BigDataVoyant. Although our proposed collection of automated metadata is already rich and covers diverse types of geospatial data assets, extra features may be added based on feedback from stakeholders in the geospatial value chain and open source developers. Complementing them with intuitive, interactive visualizations will further improve the cognitive interpretation and reveal latent aspects in the geospatial data.

ACKNOWLEDGMENTS

This work was partially funded by the the European Union's H2020 project OpertusMundi (870228).

REFERENCES

[1] Z. Abedjan, L. Golab, and F. Naumann. Profiling relational data: a survey. *VLDB Journal*, 24(4):557–581, 2015.

[2] L. Anselin, Y. W. Kim, and I. Syabri. Web-based analytical tools for the exploration of spatial data. *J. Geogr. Syst.*, 6(2):197–218, 2004.

[3] A. Chapman, E. Simperl, L. Koesten, G. Konstantinidis, L. D. Ibáñez, E. Kacprzak, and P. Groth. Dataset search: a survey. *VLDB Journal*, 29(1):251–272, 2020.

[4] C. Consonni, P. Sottovia, A. Montresor, and Y. Velegrakis. Discovering order dependencies through order compatibility. In *EDBT*, pages 409–420, 2019.

[5] D. Deng, R. C. Fernandez, Z. Abedjan, S. Wang, M. Stonebraker, A. K. Elmagarmid, I. F. Ilyas, S. Madden, M. Ouzzani, and N. Tang. The data civilizer system. In *CIDR*, 2017.

[6] J. Ehrlich, M. Roick, L. Schulze, J. Zwiener, T. Papenbrock, and F. Naumann. Holistic data profiling: Simultaneous discovery of various metadata. In *EDBT*, pages 305–316, 2016.

[7] M. Ester, H. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, pages 226–231, 1996.

[8] A. Y. Halevy, F. Korn, N. F. Noy, C. Olston, N. Polyzotis, S. Roy, and S. E. Whang. Goods: Organizing Google's datasets. In *SIGMOD*, pages 795–806, 2016.

[9] F. Naumann. Data profiling revisited. *SIGMOD Record*, 42(4):40–49, 2013.

[10] S. Shekhar, M. R. Evans, J. M. Kang, and P. Mohan. Identifying patterns in spatial information: A survey of methods. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, 1(3):193–214, 2011.

[10] Publicly available at https://github.com/OpertusMundi/BigDataVoyant

[11] E.g., https://github.com/pandas-profiling/

[12] https://pandas.pydata.org/

[13] https://vaex.readthedocs.io

[14] https://www.ogc.org/standards/sfa

[15] https://github.com/OpertusMundi/geovaex

[16] https://gdal.org/

[17] https://unidata.github.io/netcdf4-python/netCDF4/index.html