International Journal of Geo-Information

MDPI

Article

Knowledge Discovery Web Service for Spatial Data Infrastructures

Morteza Omidipoor [1,†], Ara Toomanian [1,‡], Najmeh Neysani Samany [1] and Ali Mansourian [2,*,‡]

[1] Department of GIS and Remote Sensing, Faculty of Geography, University of Tehran, Tehran 1417853933, Iran; omidipoor@ut.ac.ir (M.O.); a.toomanian@ut.ac.ir (A.T.); nneysani@ut.ac.ir (N.N.S.)
[2] Department of Physical Geography and Ecosystem Science, Lund University, Box 117, SE-223 62 Lund, Sweden
* Correspondence: ali.mansourian@nateko.lu.se; Tel.: +46-46-222-1733
† The work was done when Morteza Omidipoor was a PhD student at University of Tehran.
‡ These authors contributed equally to this work.

Abstract: The size, volume, variety, and velocity of geospatial data collected by geo-sensors, people, and organizations are increasing rapidly. Spatial Data Infrastructures (SDIs) are ongoing to facilitate the sharing of stored data in a distributed and homogeneous environment. Extracting high-level information and knowledge from such datasets to support decision making undoubtedly requires a relatively sophisticated methodology to achieve the desired results. A variety of spatial data mining techniques have been developed to extract knowledge from spatial data, which work well on centralized systems. However, applying them to distributed data in SDI to extract knowledge has remained a challenge. This paper proposes a creative solution, based on distributed computing and geospatial web service technologies for knowledge extraction in an SDI environment. The proposed approach is called Knowledge Discovery Web Service (KDWS), which can be used as a layer on top of SDIs to provide spatial data users and decision makers with the possibility of extracting knowledge from massive heterogeneous spatial data in SDIs. By proposing and testing a system architecture for KDWS, this study contributes to perform spatial data mining techniques as a service-oriented framework on top of SDIs for knowledge discovery. We implemented and tested spatial clustering, classification, and association rule mining in an interoperable environment. In addition to interface implementation, a prototype web-based system was designed for extracting knowledge from real geodemographic data in the city of Tehran. The proposed solution allows a dynamic, easier, and much faster procedure to extract knowledge from spatial data.

Keywords: spatial data mining; knowledge discovery web service; Hadoop; spatial data infrastructures

Citation: Omidipoor, M.; Toomanian, A.; Neysani Samany, N.; Mansourian, A. Knowledge Discovery Web Service for Spatial Data Infrastructures. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 12. https://doi.org/10.3390/ijgi10010012

Received: 25 November 2020
Accepted: 28 December 2020
Published: 31 December 2020

1. Introduction

The availability of Spatial Data Infrastructures (SDIs) and interoperable services provide an opportunity to establish a society that is empowered by data-driven innovation. Currently, more than 150 thousand datasets are available just in the INSPIRE infrastructure [1]. Recent developments in technologies such as smartphones and geo-sensors, in addition to paradigms such as Volunteered Geographic Information (VGI), citizen-centric data, geo-crowdsourcing information, and open-source communities increase the availability of data in our society [2–4]. In this context, the European Union (EU) encourages people, businesses, and organizations to keep and publish their data openly and freely for making better decisions in various domains. In light of the benefits, there is an urgent need for addressing the knowledge extraction problem from these voluminous data [5–7]. Within this ideal situation, spatial data remain critically significant for many businesses, and applications [8].

There has been an increasing interest in Spatial Data Mining (SDM) methods [9,10]. SDM is the process of discovering interesting and previously unknown but potentially

useful insight from large geospatial data. Recently, considerable literature has focused on methods, algorithms, tools, and frameworks related to SDM. Mining patterns [11,12], classification [13], outlier detection [14], clustering [15], regression [16], association and prediction [17,18] are the most widely used techniques that have been applied for extracting knowledge from spatial data.

Despite significant efforts in SDM techniques and efficient algorithms, the critical challenge is the distributed nature of spatial data. In a distributed environment, relevant data often reside in separate physical machines. This means that to perform SDM methods, all the required data typically need to be traditionally collected in a data repository [6]. However, this is a time-consuming process that requires reliable, scalable, interoperable, and distributed big data processing frameworks.

Although numerous Spatial Web Services (SWSs) have been developed for the collection, storage, updating, mapping, and processing of geospatial data, far too little attention has been paid to the SDM methods using standard Service-oriented architecture (SOA). Extracting knowledge from distributed geospatial data based on SWSs provides a foundation for distributed SDM in an SDI environment. SWS is a collection of software components designed based on SOA to support interoperable machine-to-machine interaction for managing spatial data over a network. The general idea for spatial knowledge extraction is to use interoperable SWS on top of a big spatial data platform. Fortunately, with the advent of modern big data frameworks, performance restrictions of traditional processing systems have been improved. Apache Hadoop and Spark are two frameworks that can integrate with web services for distributed knowledge extraction from spatial data [19].

To extract knowledge from distributed spatial data in the SDIs, in this paper, a solution based on SWS is proposed. We call our solution Knowledge Discovery Web Service (KDWS), which can be used as a layer on top of the SDIs to provide spatial data users and decision-makers with the possibility of extracting knowledge from massive heterogeneous spatial data in SDIs. By proposing and utilizing a system architecture for KDWS, this study contributes to performing spatial data mining techniques as a service-oriented framework on top of SDIs. It provides the opportunity to focus on what we typically want from the data instead of focusing on how to run SDM algorithms.

The rest of the study is organized as follows: First, the background dimensions of the research including big data processing frameworks, SDM, and SWS are addressed (See Section 2). In Section 3, components employed for the proposed solution are described. In Section 4, the implementation steps of the proposed framework are outlined. Finally, in Section 5, properties of the developed framework are discussed, and conclusions are provided.

2. Background

The growing availability of spatial data from different sources offers great possibilities for discovering valuable knowledge. Although SDI is well suited for distributed data-driven processes and sharing spatial data, it is not yet adapted for knowledge extraction in an interoperable environment [20]. It has been confirmed that retrieving knowledge from the massive, heterogeneous, and distributed spatial data requires a unique software foundation and architecture ecosystem [19]. However, some restrictions and difficulties such as conventional data storage, computing technologies, heterogeneity, and interoperability concerns of spatial data led to a delay in the development of such an architecture ecosystem. A more comprehensive description of these challenges can be found in the study conducted by [21].

Recent studies (See [19,22,23]) have showed that distributed and parallel processing frameworks make it possible to meet performance requirements for handling large-scale big spatial data. In this context, Apache Hadoop, an open-source framework for reliable, scalable, and distributed computing has emerged as powerful platforms adapted to tackle the big data challenges (https://hadoop.apache.org/). Hadoop is capable of storing and managing large volumes of data efficiently, across clusters of computers by using simple

programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. The main concept of the framework is isolated into two parts, the Hadoop Distributed File System (HDFS) for storing data and the MapReduce programming model for the process of data usually stored on HDFS [24]. Apache Spark, an in-memory distributed computing is another framework that provides a novel data abstraction called Resilient Distributed Datasets (RDDs). The RDDs are collections of objects partitioned across a cluster of machines (https://spark.apache.org/). To date, several studies have investigated that Hadoop and Spark frameworks provide high-performance computing for retrieving patterns and knowledge from a massive volume of spatial data. Park, Ko, and Song (2019) [25] proposed a method to ingest big spatial data using a parallel technique in a cluster environment. S. Li et al. (2016) [26] introduced an HDFS-based framework with native support for spatio-temporal data types and operations named ST-Hadoop. Apache Sedona (Formerly GeoSpark) is another cluster-computing framework for processing large-scale spatial data (http://sedona.apache.org/). Sedona extends Apache Spark and SparkSQL with a set of Spatial RDDs and SpatialSQL that efficiently load, process, and analyze large-scale spatial data across machines [27]. It provides functionalities including spatial data partitioning, spatial indexing, and spatial join with several APIs, which allow users to read heterogeneous spatial objects from various data formats. GeoMesa is also a Spark-based open-source suite that enables large-scale geospatial querying and analytics on distributed computing systems (https://www.geomesa.org/). It provides NoSQL databases for massive storage of point, line, and polygon data. Through GeoServer, it also facilitates integration with a wide range of existing OGC standards such as Web Map Service (WMS) and Web Feature Service (WFS).

Advances in SOA have provided a bright vision for managing distributed and heterogeneous spatial data [28–32]. SOA is defined as software architecture style, concept, or paradigm that involves principles such as loose coupling, reusability, interoperability, scalability, agility, flexibility, and technology-independent by using standardized and modular components called as service [33,34]. Generally, the functionality and behavior of service design are implemented based on interface specification. Data types, operations, transport protocol binding, and the network location of service are the most important characteristics that describe interface specification. To date, to better manage geospatial data based on SOA, a collection of standards has been developed by OGC. These standards, known as OGC Web Services (OWS), can be categorized into different groups such as data models, encoding, searching, storing, processing, mapping, and publishing geospatial data [28,29,32]. In addition, a set of SWS has been developed every year, being reviewed by OGC working groups. It can be expected that in the coming years, SWS will consider various aspects of spatial data management, and the SOA paradigm will be used increasingly in geospatial science, systems, and societies [20,35,36]. In the review of mash-up GIS services, Chow (2011) [35] highlighted the paradigm shift of GIS "from an isolated architecture to an interoperable framework, from a standalone solution to a distributed approach, from individual proprietary data formats to open specification exchange of data, from a desktop platform to an Internet environment". Based on previous studies and the recent trend in technologies, it seems that a paradigm shift will happen from GI-systems to GI-Services. GI-Services may also become more important than GI-systems in academic, industrial, and business communities in recent years (Figure 1).

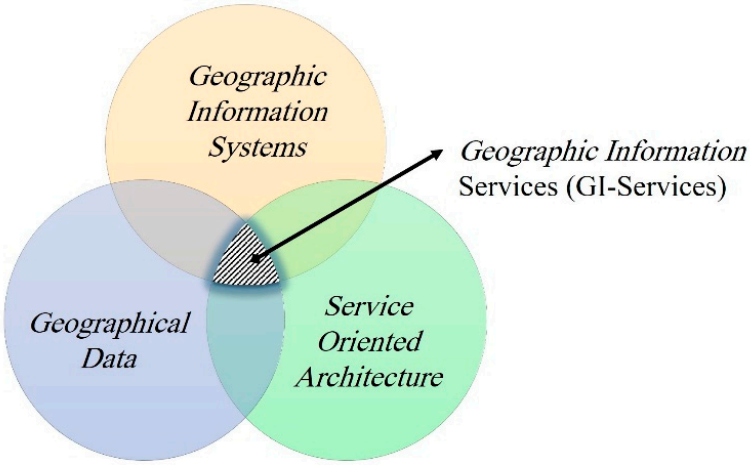

Figure 1. A paradigm shift from Geographic Information (GI)-systems to GI-Services.

Web service-based spatial knowledge discovery can be used for handling interoperability issues of spatial knowledge discovery in an SDI environment. A number of studies [37–42] have proposed the use of SOA for data mining, Machine Learning (ML), Business Intelligence (BI), and On-Line Analytical Processing (OLAP). Zorrilla and García-Saiz (2013) [42] described an SOA-based software architecture to extract useful and novel knowledge using data mining techniques in order to obtain patterns that can be used in the decision-making process. The architecture was presented in 5 layers including the data layer, enterprise component, services, business process, and presentation layer. The service layer is offered as a web service, which makes it easily accessible from any client application. Its main characteristic is that it is based on the use of templates that answer certain previously defined questions. These templates gather the tasks of the Knowledge Discovery in Databases (KDD) process to be carried out on the data set, which is sent by the end-user. Medvedev et al. (2017) [43] implemented a number of data mining methods as WS and used them in a web-based data mining tool. The main idea of the research is to provide scientific workflows to extract useful patterns from large data sets based on the service component. Here, the scientific workflows allow composing the convenient model of the data mining process covering a number of different methods. By using Apache Hadoop, Kusumakumari, Sherigar, Chandran, and Patil (2017) [44] propose a time-efficient algorithm for mining frequent item sets in real-time streaming data. They observe that the Hadoop works well for mining frequently occurring patterns. They concluded that using the same proposed method in a MapReduce framework significantly lowers the execution time taken. Omidipoor et al. (2018) [20] demonstrated that a set of Web service-based SDM methods is necessary to respond to the GIS community requirements. To facilitate distributed SDM methods, they (See [20]) proposed a general Spatial Knowledge Infrastructure (SKI) in which SWS has a pivotal role in their study.

Although extensive research has been carried out on integrating DM techniques based on SOA, far too little attention has been paid to the use of SDM methods as an interoperable WS procedure. It is now well established from a variety of studies that in recent years, significant progress in parallel and distributed GIS systems occurred. The question that can be addressed is how to extract useful knowledge from heterogeneous and distributed spatial data using interoperable and standard services. As [21] mentioned, heterogeneity requires interoperability and standards among the data processing tools. A new approach is, therefore, needed for spatial knowledge discovery that tackles interoperability challenges.

3. The Proposed Solution

The proposed solution integrates the capabilities of SOA and SDM techniques in an interoperable and parallel computing engine to facilitate the knowledge extraction process from SDIs. Figure 2 presents the general architecture of the proposed solution. The

architecture contains four major layers to provide desirable functionalities and capabilities: (1) the SDI layer is responsible for the integration of distributed and heterogeneous data from different SDIs, using modern big data storage technologies; (2) a knowledge discovery engine layer supports high-performance spatial data mining techniques across clusters of computers named processing workers; (3,4) seamless and interoperable interaction between clients and bottom layers provided by the KDWS. In the following sections, a detailed description of the architecture components is described.

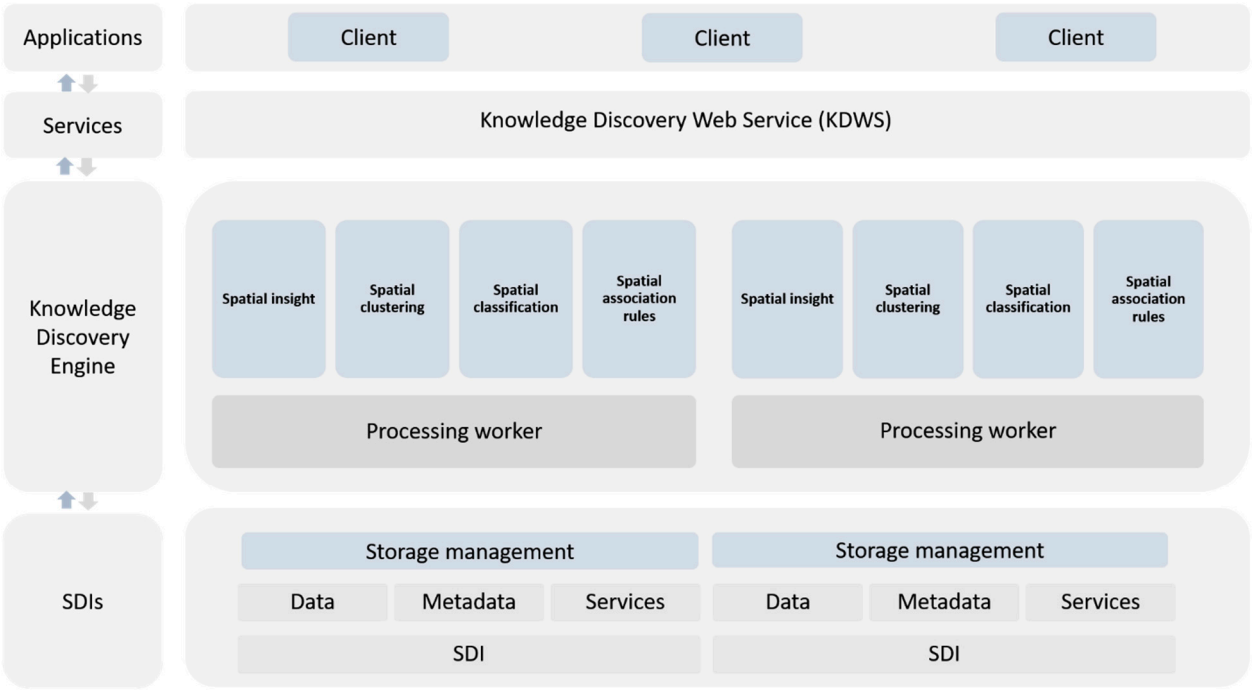

Figure 2. Proposed framework for extracting knowledge from spatial data.

3.1. SDIs Layer Component

The SDIs layer component is the core component for spatial data integration. The component allows the integration of data from different SDIs to extract, transform, and load (ETL) in a modern big data storage system. Specifically, the component is responsible for loading various sources of data into an HDFS cluster. HDFS is designed to support extremely large files (terabytes), and it is well satisfied for applications that follow "write-once-read-many" semantics and require that these are "read" to be satisfied at streaming speeds. This is consistent with the goals of data mining in SDIs. An HDFS cluster primarily consists of a NameNode that manages the file system metadata and DataNodes that store the actual data (master/slave architecture). Internally, a file is split into one or more blocks, and these blocks are stored in a set of DataNodes. To provide reliability, it maintains each file as a sequence of blocks. The NameNode and DataNodes have built-in web servers that make it easy to monitor the status of the cluster. A comprehensive description of HDFS architecture can be found in Hadoop HDFS architecture available on the official website.

In the proposed solution, the mentioned process is managed by a storage management utility. It delivers a wide range of spatial data formats including XML-based, JSONbased, CSV, and other traditional vector formats into an HDFS. Currently, it is possible to handle efficiently massive amounts of data by using in-memory ETL tools. It should be noted that metadata-driven spatial ETL can be used for the integration of SDI data sources in HDFS by utilizing OGC Web services.

3.2. Knowledge Discovery Engine

The knowledge discovery engine is a core-processing component for extracting knowledge from a big spatial data platform. To prevent low time latency, the layer uses parallel or cluster computing solutions. This layer also allows users to read the heterogeneous spatial objects from various data formats and running parallel spatial processing tasks. In this regard, innovative open-source parallel computing frameworks such as GeoSpark can be used. The main components of GeoSpark provide a set of spatial RDDs (SRDDs), which is a read-only collection of data that can be partitioned across a subset of Spark cluster machines. In the proposed solution, each partition assigns to a processing worker. Although data mining techniques are very diverse, the most common and most widely used techniques are reviewed based on [6,9,10] and [45] are implemented in the proposed solution. Techniques or algorithms considered in the knowledge discovery engine are described in the following sections. In addition to general data mining methods, efforts have been made for implementing spatial explicit data mining methods.

3.2.1. Spatial Clustering

Spatial clustering algorithms are among the most widely used groups of SDM methods. The core idea of spatial clustering is to summarize geospatial objects into K categories such that the intra-cluster and between-cluster similarity is maximized and minimized, respectively [14,46,47]. Several partitioning, hierarchical and density-based clustering algorithms are supported by the knowledge discovery engine component (See Table 1). Since these algorithms only deal with attribute similarity, the spatial explicit clustering method developed by [46] has been implemented. In addition to grouping geospatial objects, hot and cold spots as well as a spatial outlier can be identified by using this method.

Table 1. Clustering algorithms supported by the knowledge discovery engine component.

Algorithm	Use Case
Anselin cluster and outlier [46]	Spatial explicit cluster with outlier expected
K-Means [48]	General-purpose
Birch [49]	The data volume is large
Spectral [50]	Few clusters expected
Ward [51]	Many clusters expected
OPTICS [52]	Many clusters and connectivity constraints expected
DBSCAN [53]	Density estimation

3.2.2. Spatial Classification

Spatial classification is a data mining method used for finding a model that describes and assigns a class label for geographic objects based on spatial relationships [54,55]. Depending on the purpose, context, and data availability, this process can be done using either supervised or unsupervised algorithms [56]. Supervised classification performs in two steps, learning and testing [57,58]. In the first step, some portions of the dataset are selected as training datasets, and then different algorithms are used to build a classifier. In the testing step, the classifier is used to predict class labels and test the model. A rule derived from a set of training data can be evaluated using two criteria: coverage and accuracy. Although in most algorithms, spatial relationships are not considered, some researchers have also proposed algorithms to consider spatial relationships [54,55]. Classification algorithms are supported by the knowledge discovery engine component presented in Table 2.

Table 2. Classification algorithms supported by the knowledge discovery engine component.

Algorithm	Use Case
K-nearest neighbors [58]	A non-parametric method, the decision boundary is very irregular
Decision trees [55]	Handling both numerical and categorical data
Support vector machines [56]	Binary classification problems
Random forests [59]	High computational complexity
Logistic regression [56]	Input variables have a Gaussian distribution

3.2.3. Spatial Association Rule Mining

Spatial association rule mining is among the most interesting methods used in the SDM [9,18,26]. This technique is a kind of mining process that is used for discovering useful relations, associations, or patterns that are not explicitly stored in spatial databases. A spatial association rule can be defined as X→Y, in which the representation of X and Y shows the predicate set [17,18]. There are two main steps to finding useful rules. First, all combinations of geographic objects that occur with a specified minimum frequency should be found. Then, based on recognized frequent item sets, rules are evaluated. The Apriori is the most widely used association rule-mining algorithm for finding frequent item sets. Since the number of possible identified rules is usually too high, a threshold value is given by the user to determine the importance of identified spatial association rules (strong rules). There are three common methods used to evaluate the importance of identified rules: support, confidence, and lift [9,17].

3.3. Services Layer

To make a seamless and interoperable interaction between end-users and the knowledge extraction engine component, a web service called KDWS was developed. From an operational point of view, a client delegates a series of parameters to his/her request and assigns an SDM task to the knowledge extraction engine. Then, the knowledge extraction engine presents the extracted spatial knowledge to the user based on the parameters received by the user. In this context, spatial knowledge refers to the interoperable output of SDM techniques, provided by the KDWS.

The most important concepts used in the KDWS are service, interface, and operation. Service is a set of interfaces provided by an entity. Different services have different functions that are independent and separable. For example, different services, such as retrieving, mapping, and processing, are considered as separate services. In order to execute a process (here, knowledge extraction from spatial data), the method of referring requests to an object (here, the server means) is specified by an interface. In the simplest case, the KDWS interface describes the name of the operation, list of parameters, and the values allowed to extract knowledge from a database. In connection with the KDWS, various operations are expected, which are defined by the service interface. The KDWS service interface depicted in Figure 3 is based on Unified Modeling Language (UML).

The class diagram shows that KDWS inherits GetCapabilities operation from the OWS interface and adds three operations named GetInsight, GetSpatialClusters, GetSpatialClassification, and GetSpatialAssociationRules. It should be noted that OGC /ISO standards related to UML notation have been used to define interface specifications of KDWS. Given the wide variety of SDM techniques, it is very difficult, if not impossible, to create a single schema that can provide different structures and parameters as an interface. Therefore, each data mining algorithm is defined as a separate interface. Although the proposed service operations are different from other OGC services, such as Web Processing Service (WPS), Web Coverage Service (WCS), and WFS, there are many similarities with common interfaces. Therefore, since OWSs use a standard called Common Implementation Specification in shared implementations, this specification is used in this study [60]. Request operation in the KDWS is implemented based on OGC 05-008 standard. Therefore, the request can be made based on an HTTP GET, KVP coding, or based on HTTP POST. In

the service, according to parameters specified by the user's request, interoperable and standard output is provided.

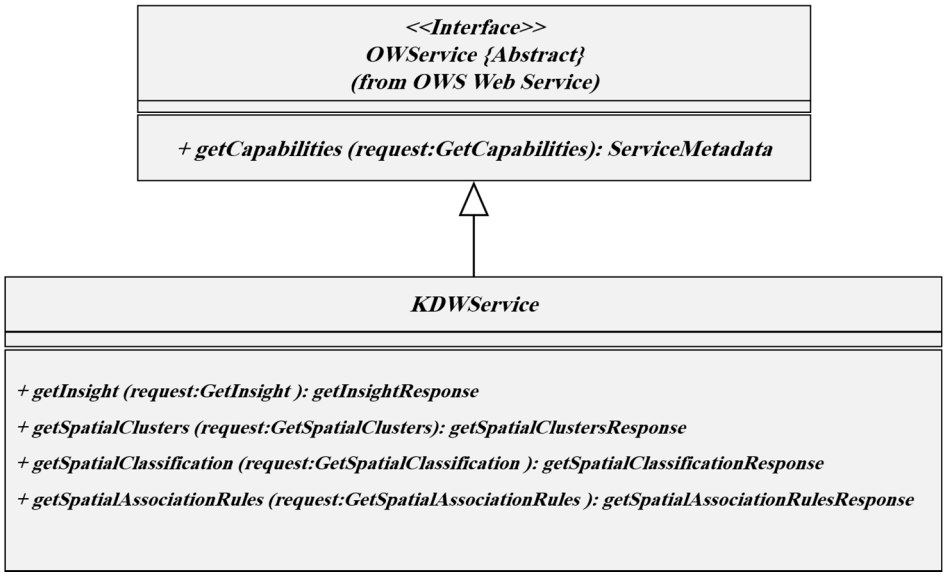

Figure 3. Class diagram of the Knowledge Discovery Web Service (KDWS).

To extract knowledge from spatial data, different operations are supported in the KDWS described in the following sections.

3.3.1. GetCapabilities Operation

In order to understand how to use the KDWS, a series of metadata information in the form of an interoperable format (i.e., XML document) is required. The document not only contains valid KDWS requests, but it also refers to the service providers, and other service components, such as operations descriptions, supported SDM algorithms, parameters, access levels, header information, and available dataset. While a web server receives a request from the client, this metadata information is sent to the client in an interoperable format. Figure 4 shows the GetCapabilities UML class diagram.

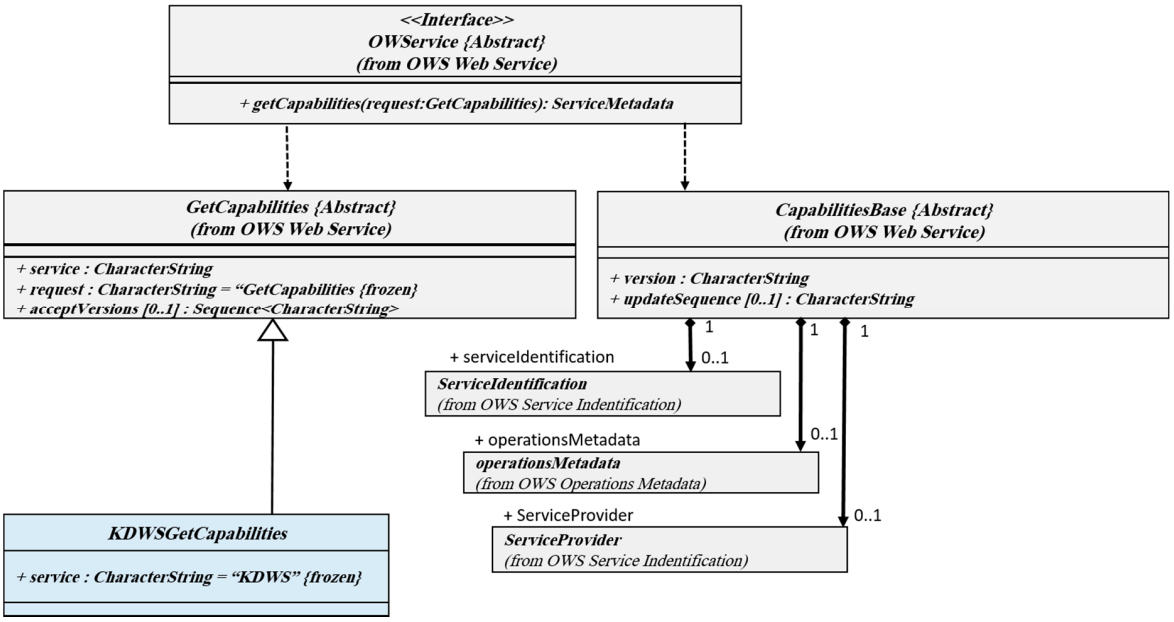

Figure 4. Interface of the GetCapabilities operation.

As shown in the above interface, in addition to inheriting OGC-based classes, the KDWS metadata has added new classes to describe and provide metadata related to GetCapabilities operation.

3.3.2. GetInsight Operation

The GetInsight operation provides an overview of geographic data. Here, insight can be summarization, distribution, and the relation between variables. Generally, the client wants to have a summary of the geographic data in different formats such as image, JSON, and XML. Quantifiers such as mean, median, as well as measures of variability are the most important benchmarks that are emphasized in the operation. Data distribution and communication between variables are also expected in the GetInsight operation. Summarization, relations, and distribution are the most important insight types that have been implemented in the GetInsight operation. Figure 5 shows the GetInsight operation class diagram.

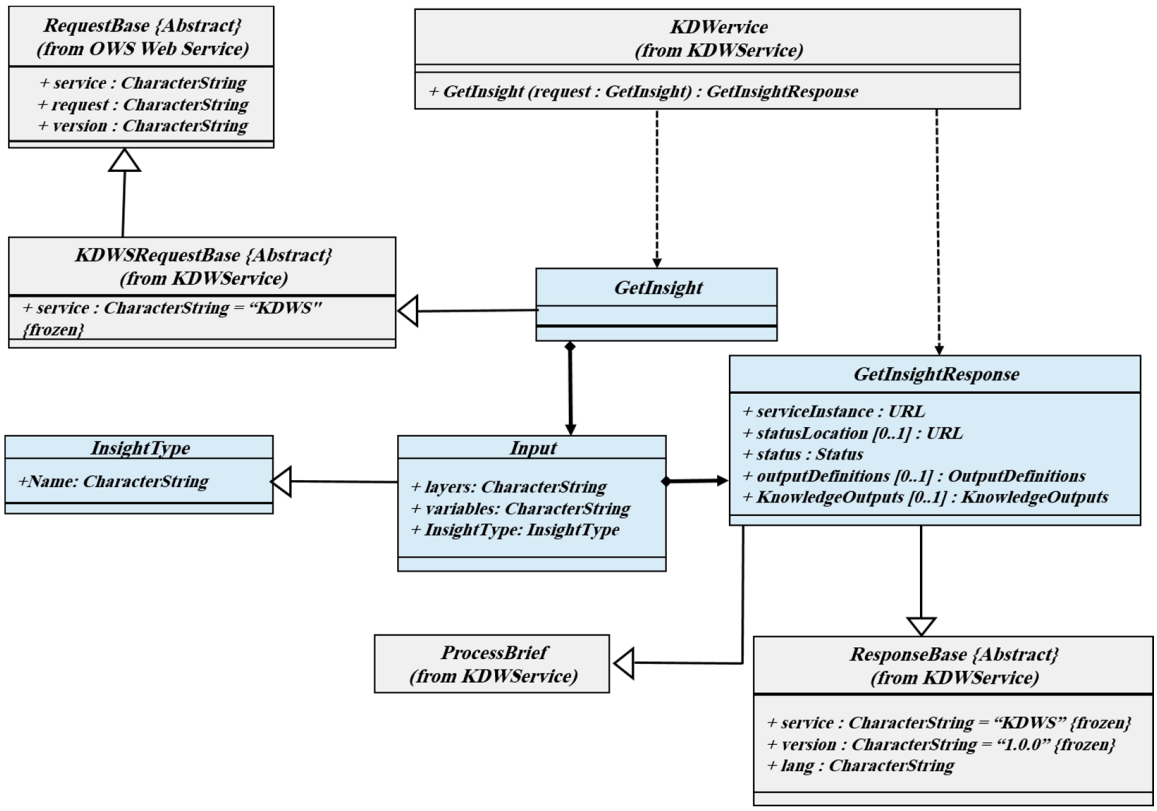

Figure 5. Interface of the GetInsight operation.

3.3.3. GetSpatialClusters Operation

In the GetSpatialClusters, the client chooses the algorithm, data set, related variables, and response format. In the GetSpatialClusters request, the output of the algorithm is presented in JSON format by default, so it can be reused and shared. It is possible to obtain the output as XML or GML. The GetSpatialClusters operation class diagram is depicted in Figure 6.

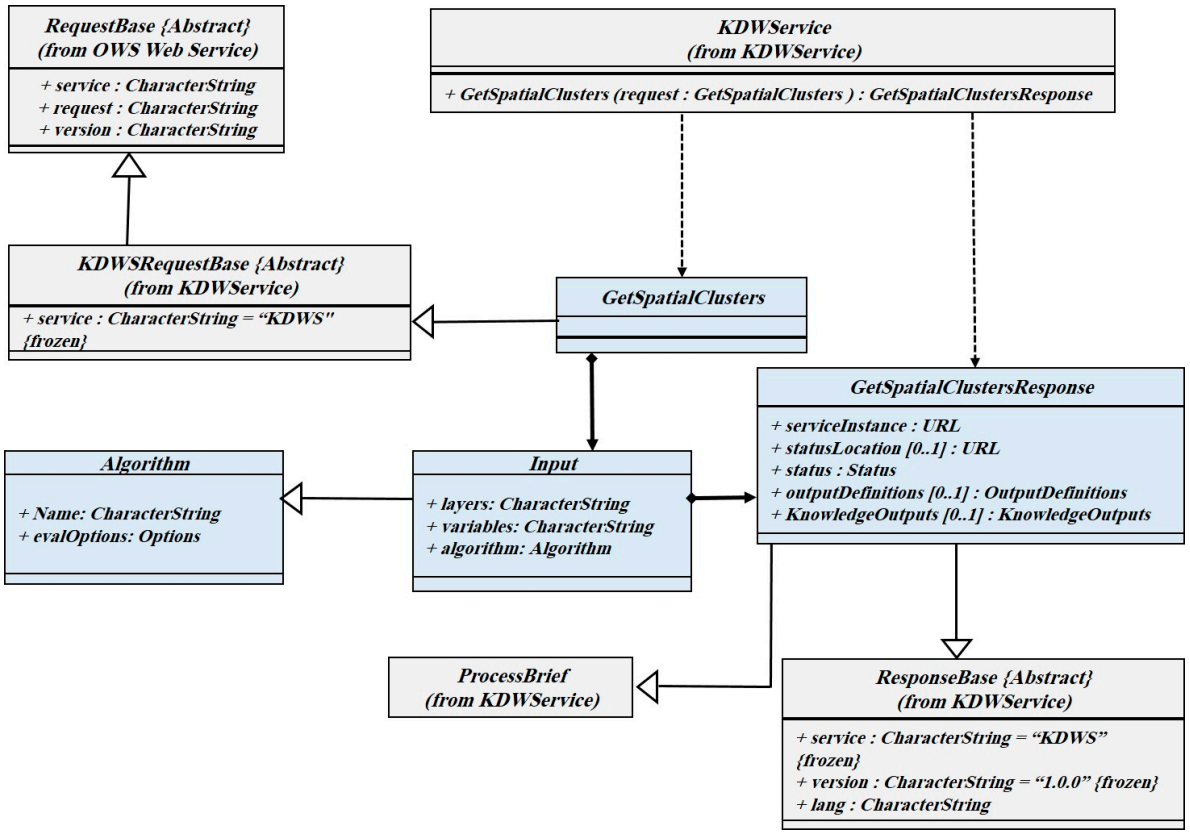

Figure 6. The interface of the GetSpatialClusters operation.

3.3.4. GetSpatialClassification Operation

Implementing spatial classification algorithms in the form of web services is more complex than other SDM algorithms. The reason should be founded in the separate training and testing steps. To handle this complexity in the KDWS service interface, it is possible to consider a part of the data set for testing and learning of the model. For example, 20 percent of the dataset can be randomly considered as test data, and the model can be calculated after the execution of the algorithm. The number of mandatory and optional parameters varies depending on the algorithm used. Unsupervised algorithms in the GetSpatialClassification operation are also supported. The GetSpatialClassification operation class diagram is depicted in Figure 7.

3.3.5. GetSpatialAssociationRules Operation

Among the most interesting operations supported in the KDWS is GetSpatialAssociationRules. It identifies hidden rules in a set of spatial datasets. Usually, many of the identified rules may not be very important, so the proposed service provides the extracted rules with two important evaluation criteria: support and confidence. Therefore, in all extracted rules, the value and criterion of acceptance of association rules will be defined. Like mentioned operations, the output of the knowledge extracted in the above request can be received in a different interoperable format, so it can be reused and shared. Figure 8 shows a class diagram of the GetSpatialAssociationRules operation.

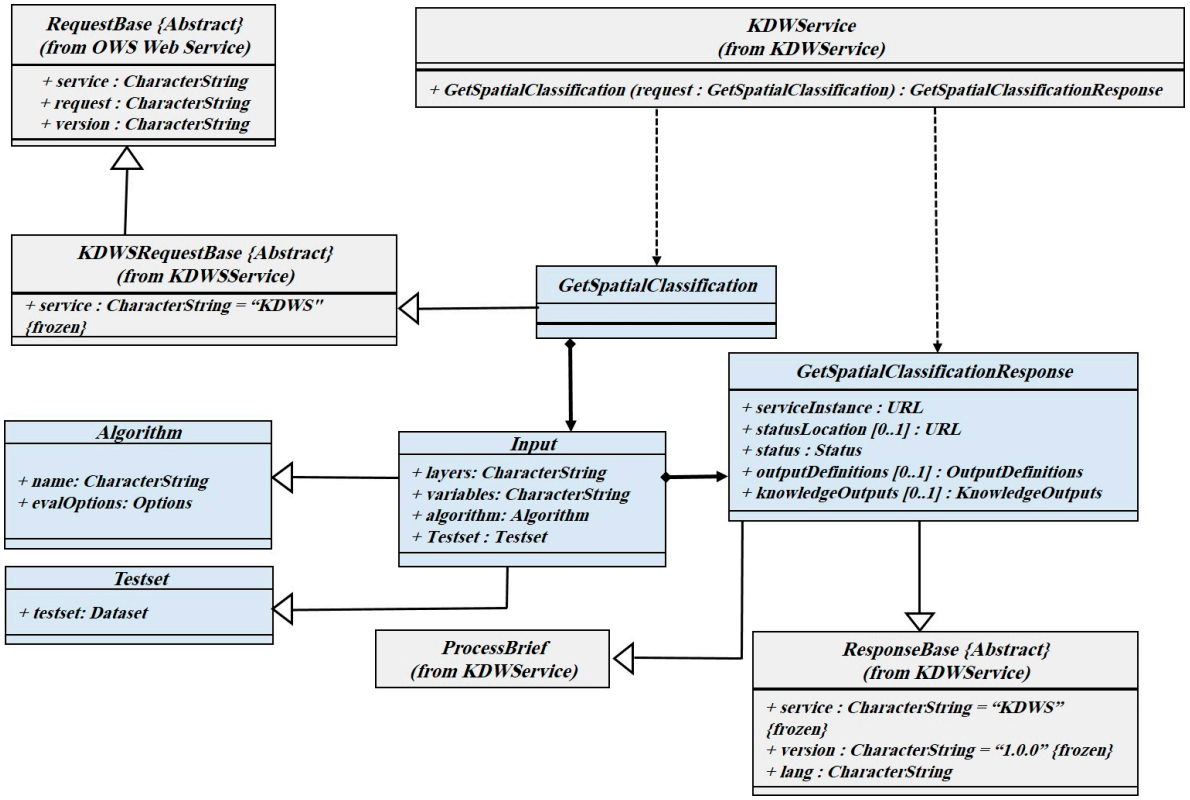

Figure 7. The interface of the GetSpatialClassification operation.

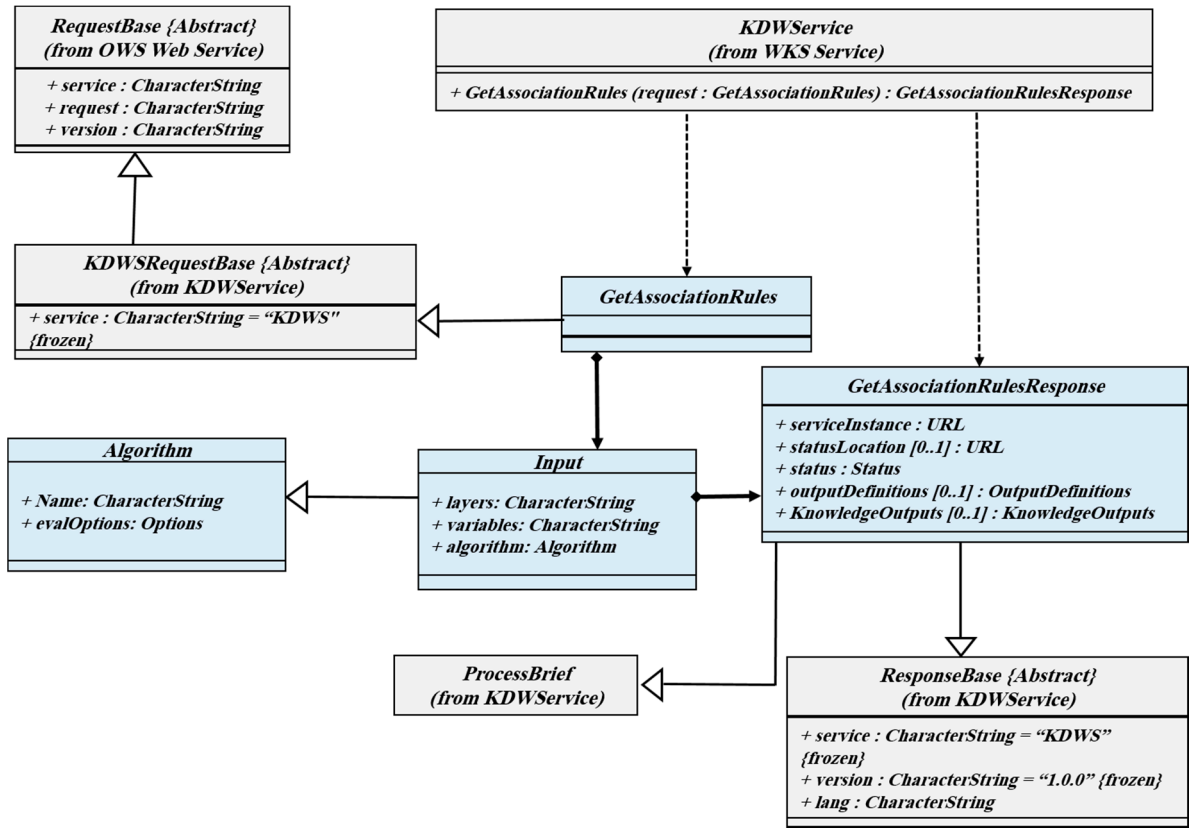

Figure 8. The class diagram of the GetAssociationRules operation.

3.4. Applications Layer

This is the topmost layer of the proposed architecture. The utilization of KDWS provides usable and interoperable knowledge that can be used in various applications. Generally, many organizations, markets, and industry decisions require spatial knowledge; therefore, the proposed solution can provide a soft spatial knowledge infrastructure in an interoperable and participatory environment. For example, some organizations could participate together to share knowledge extracted by their KDWS services, and generate a collaborative environment during spatial decision-making processes. In addition to exchange service outputs, some organizations may also share their service itself with others. In this situation, relevant organizations can use service capabilities according to their needs. In such cases, a service registry can provide a description of the KDWS and available spatial data and algorithms. Then, spatial knowledge discovery algorithms are utilized as an executable workflow. Finally, a KDWS service chain can then be created to bind the extracted spatial knowledge to generate high-level knowledge.

4. Implementation and Results

To examine the capabilities of the framework, a scenario is defined and the capabilities of the proposed solution are determined. In the scenario, information related to geodemographic data published as standard web services in different local SDIs. The goal is to extract useful knowledge from these datasets through the proposed solution. In this scenario, spatial knowledge refers to the output of SDM techniques, provided by the KDWS.

A case-study approach was adopted to gain a detailed understanding of the proposed framework. The dataset used is related to three districts of Tehran, Iran (Districts: 3, 6, and 11). These regions are representative of different quality of life levels, which are located in the northern, central, and southern parts of the city (See Figure 9). From socio-economic vision, different social classes live in the city, including the affluent class (upper) in the northern part, semi affluent in the central parts (middle), and deprived class (lower) in the southern parts.

Figure 9 shows the location of the selected districts, and population density at the census blocks. These blocks contain 3184 units. Socio-economic and geodemographic indicators related to these blocks are stored in a distributed network, published by different authorities. The indicators are shown in Table 3.

Table 3. The most important geodemographic indicators used for testing of the proposed methodology.

Indicator	Description
Employment ratio	The ratio of employed people to the active population
Income	The average income of individuals from different job groups in each block.
Literacy	A combination of three parameters: literate over 6 years old, students and university graduates
Welfare amenities	Having basic services (water, electricity, gas and telephone), personal car and computer
Single and divorced	Combination of divorce rate and percentage of people never married (more than 50 years)
Population aging	The ratio of people over 65 years to the total population
Youth population	The ratio of age groups from zero to 14 years to the total population
Dependency ratio	The population of 0-14 and >65 age groups to a population of 15 to 65 years
Gender ratio	Number of men for every 100 females

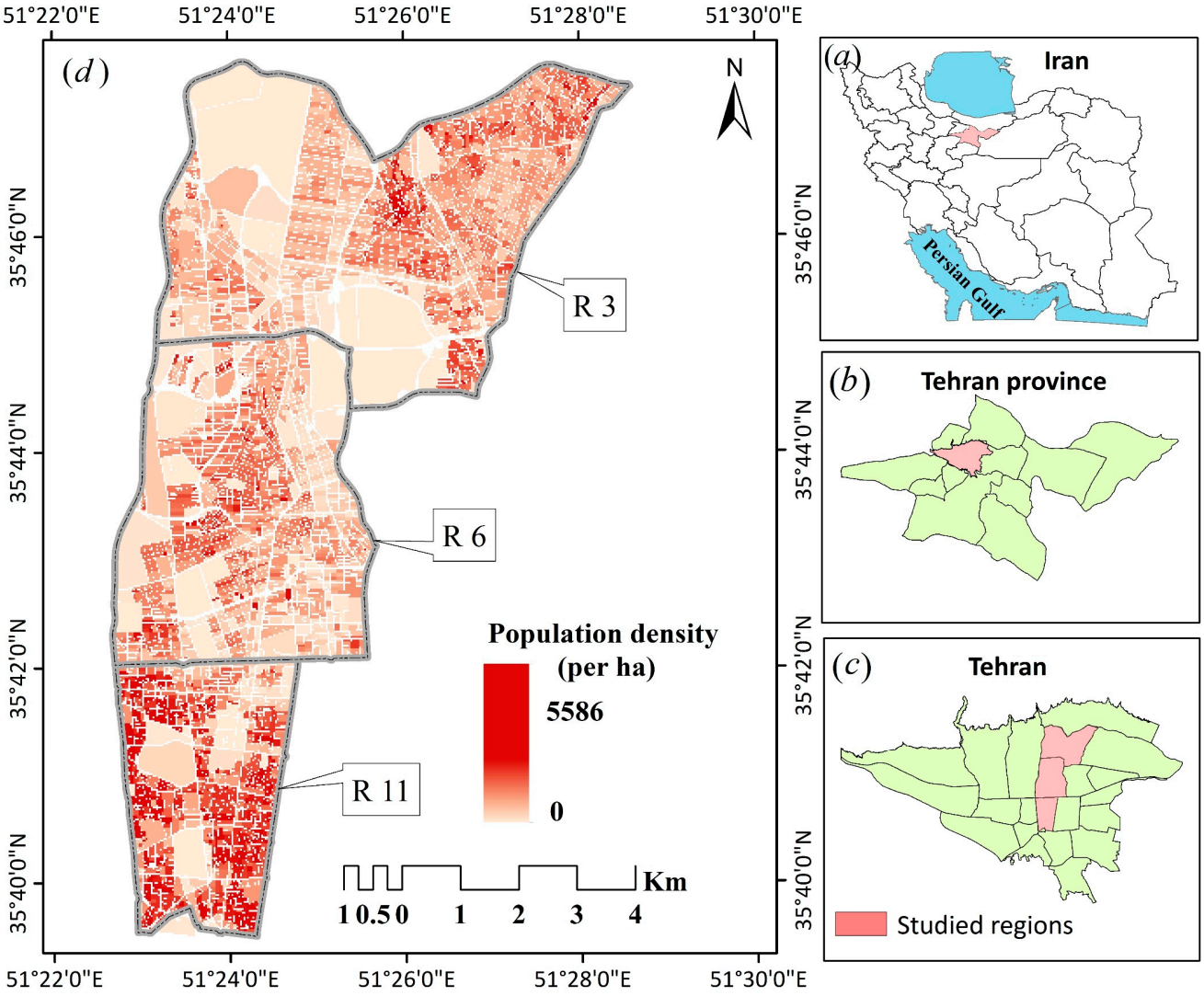

Figure 9. Location of the study area. (**a**) Iran, (**b**) Tehran province, (**c**) Tehran city, and (**d**) Districts 3, 6 and 11.

4.1. Implementation Workflow

This section describes the implementation steps for extracting knowledge from distributed spatial data. As shown in Figure 10, the implementation workflow contains three major phases as follows:

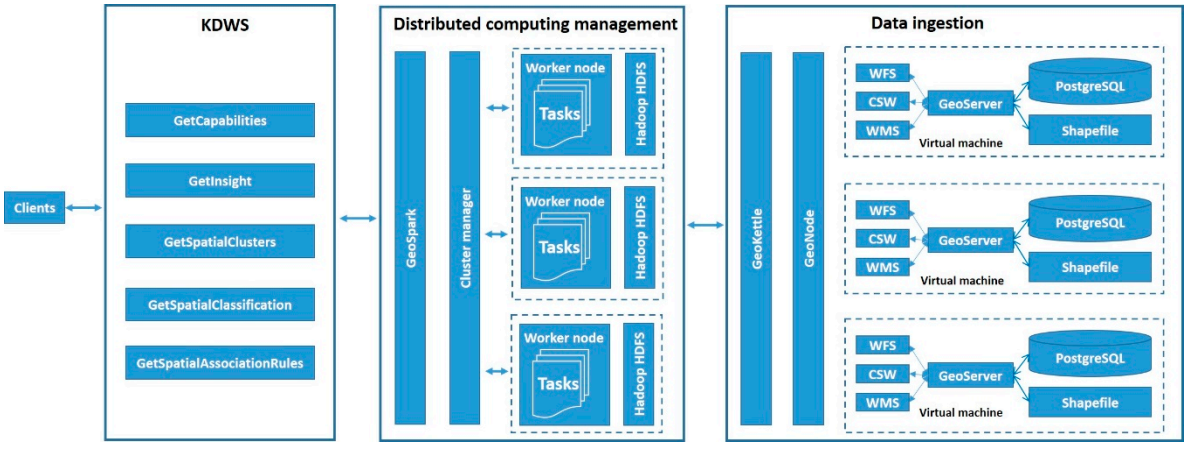

Figure 10. The implementation workflow of the proposed solution.

4.1.1. Data Ingestion

In this phase, spatial data from different SDIs are loaded and integrated into the HDFS-based data storage systems. Here, it is very important to understand how data, metadata, and geospatial services available in different SDIs are organized in the solution. To simulate such an environment, Oracle VirtualBox, a cross-platform virtualization software (https://www.virtualbox.org/), was used for creating three virtualized machines. For each machine, resource allocation, database server (here, PostgreSQL), GIS server (here, Geoserver), and network-related settings are configured. GeoNode (http://geonode.org/), a geospatial content management system, was used for easy-to-use managing data, metadata, and publishing different OGC services including WMS, WFS, and Catalogue Service for the Web (CSW). After publishing data as OGC services, the data should be stored in the desired HDFS provided by the Hadoop framework. By using GeoKettle, a metadata-driven spatial ETL tool geodemographic data stored in the desired HDFS.

4.1.2. Distributed Computing Management

In this stage, all required data are properly stored in the form of local HDFS files. To efficiently utilize the capabilities of parallel processing, it is necessary to partition data across separate machines. Moreover, the required processing for knowledge extraction should break down into different computing units (worker node). There are some challenges in partitioning and parallel processing management. Unfortunately, native Hadoop storage utilities split data into multiple partitions without properly considering spatial characteristics (i.e., data types, spatial indexes, and geometrical operations). For partitioning necessary data based on spatial characteristics, GeoSpark (http://sedona.apache.org/), a cluster computing system for processing large-scale spatial data, was used. It extends the RDD, the core data structure in Apache Spark, to accommodate big geospatial data in a distinct cluster. Data in SRDDs are partitioned according to the spatial data distribution, and nearby spatial objects are very likely to be put into the same partition. Managing partitioned data location (the corresponding path of data) and different computation (tasks) across multiple machines is another challenge. For carefully keeping track of where the data are stored across the distributed HDFS, a utility named "Cluster management" was used. By using a mapper function, it integrates distributed competitions on the nodes where the partitioned data are located.

Python libraries such as PyClustering [61], PySAL [62], and Scikit-Learn [63] are used for implementing spatial data mining tasks. Additionally, The GeoPandas, an open-source python project that supports geospatial data types (https://geopandas.org/), was used for the implementation of classification algorithms. In addition to the mentioned libraries, a pure Python implementation was used for spatial association rule mining and other functionalities.

4.1.3. KDWS

Seamless and interoperable interaction between clients and distributed knowledge discovery tasks are provided by the KDWS. The capabilities achieved by the KDWS were mentioned before (See Section 3.3).

The Django web development framework was used in the implemented process of the prototype web framework (https://www.djangoproject.com/). This allows the integration of system modules based on a loosely coupled approach [64].

As shown in Figure 11, different capabilities of the proposed solution are implemented in a web-based system prototype. These capabilities are achieved by the KDWS operations mentioned before.

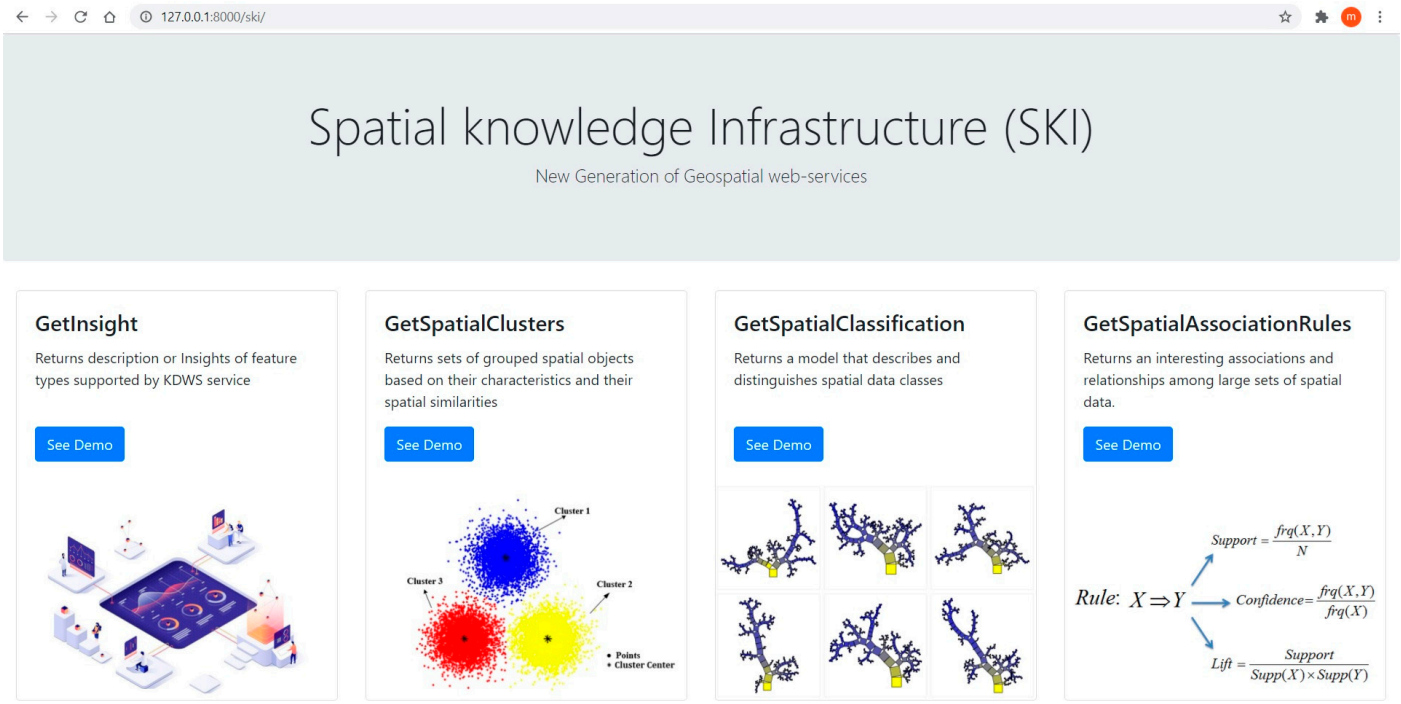

Figure 11. A prototype system developed to use the KDWS.

4.2. Results

The HTTP request–response protocol allows clients to communicate with the implemented prototype. The client submits an HTTP request message to the server based on HTTP GET or HTTP POST. A series of metadata information contains valid KDWS requests/response parameters provided by the GetCapabilities operation (See Section 3.3.1). Based on required knowledge discovery tasks, the client chooses the algorithm, data set, related variables, and response format. In the following, a sample KVP request is shown.

http://127.0.0.1:8000/KDWS?service=KDWS&version=1.0.0&request=getSpatialclusters&algorithm=k-means&dataset=geodemographic&variables=literacy,income&outputformat=application/json.

Results of GetSpatialClusters for the data set used in the study area are shown in Figure 9. As shown in Figure 12, homogeneous statistical blocks of the study area based on different variables including income, employment, divorce rate, and literacy rate were identified as different spatial clusters. In the figure, green and blue regions are clusters as well as red, and purple is the outliers.

Figure 13 shows the results of the GetSpatialClassification operation applied to the geodemographic data of the study area. The aim is to classify statistical blocks into separate classes based on social and economic indicators. Due to the lack of training data, the K-NN classification algorithm was used. As a result, the most similar spatial features were separated from other features in the form of classes and are presented in Figure 13.

By applying the GetAssociationRules operation, the relationships between the various variables are presented as a set of association rules. The following examples are such rules for which the support and confidence are provided for evaluation.

Rule 1: Literacy rate='(0.96-1]' Computer and Car='(0.71-0.85]' Family size='(2.9-3.5]' Dependency burden='(0.2-0.4]' 398 ==> Annual income (M Iranian Rial)='(153 -171)' 376 <supp:(0.125) conf:(0.94)>

Rule 2: Divorce rate='(0-0.052]' Annual income (M Iranian Rial) ='(153-171]' Computer and Car='(0.72-0.86]' Dependency burden='(0.2-0.4]' 478 ==> Literacy rate='(0.96-inf)' 435 <supp:(0.15) conf:(0.91)>

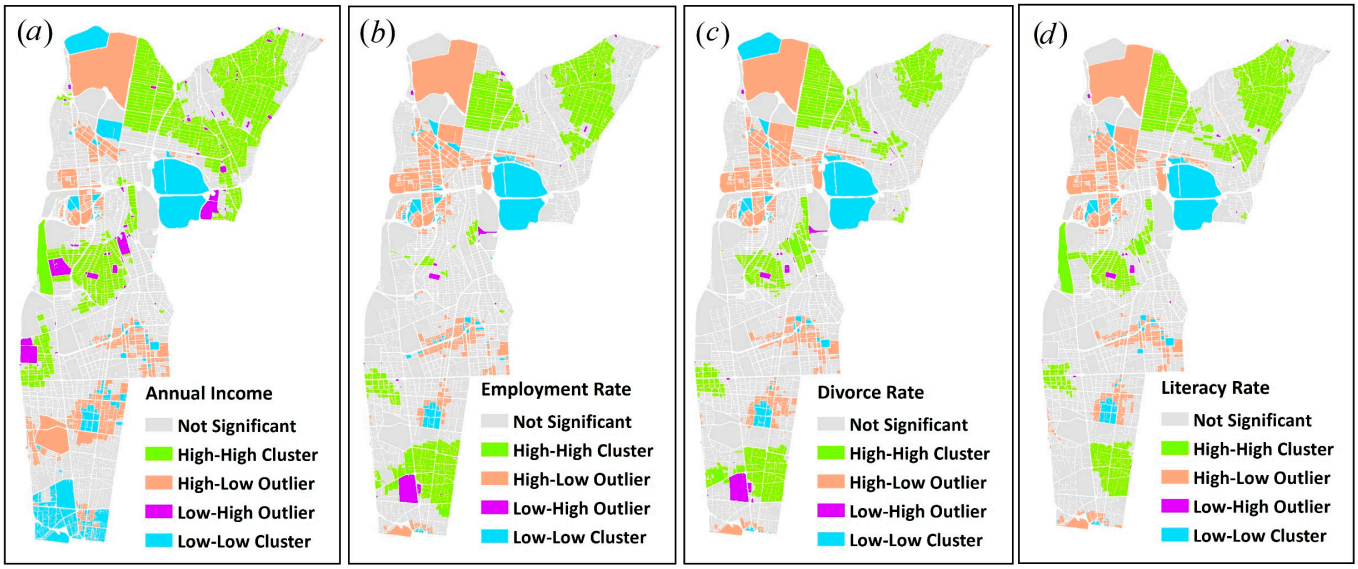

Figure 12. An example of the GetSpatialClusters operation output applied to geodemographic data of Tehran: (**a**) annual income, (**b**) employment rate, (**c**) divorce rate, and (**d**) literacy rate.

In the first rule, if the literacy rate is more than 96%, 71–85% of households have a computer and personal car, the family size is between 2.9 and 3.5, and the dependency burden is between 0.2–0.4, then the average annual income of households living in these blocks will be between 153 and 171 million Iranian Rials. The support and confidence values for this rule were 0.125 and 0.94, respectively.

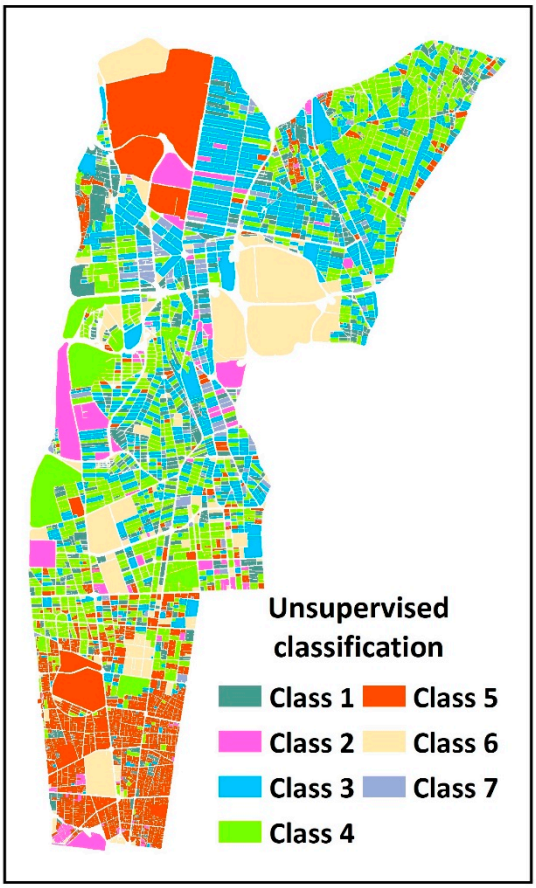

Figure 13. The output of GetSpatialClassification operations applied to study area.

Figure 14 shows the side effects of the components (conditions) in the second rule, respectively. According to this rule, census blocks with less than 5.2% divorce rate and annual income between 153 and 171 million Iranian Rials and households having a computer and personal car between 71 and 85% and burden of responsibility between 0.2 and 0.4 have people with a literacy rate of more than 96%.

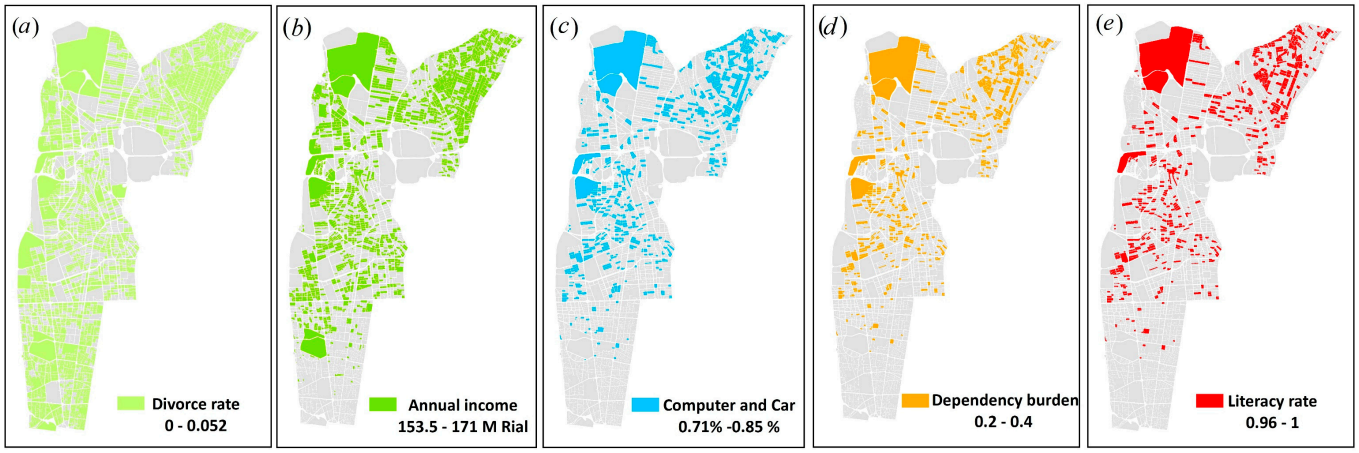

Figure 14. The output of GetSpatialAssociationRules operation applies to geodemographic data. (**a**) Divorce rate, (**b**) annual income, (**c**) computer and car, (**d**) dependency burden, and (**e**) literacy rate.

5. Discussion and Conclusions

The purpose of the current study was to propose a framework for extracting knowledge from distributed spatial data on top of SDIs. The framework integrates SOA and SDM techniques to enable spatial knowledge extraction processes. In this context, we introduced an architecture that contains four major layers. The data layer is used for the integration of distributed and heterogeneous spatial data. The knowledge discovery engine layer supports high-performance spatial data mining techniques across clusters of computers, and the KDWS layer provides interoperability for applications. Accordingly, a web service is implemented that supports SDM techniques in modern data storage and a parallel computing platform named KDWS. The KDWS interfaces are implemented based on interoperability standards that support the most important SDM techniques include spatial clustering, classification, and association rule mining. In addition to the interface implementation, the procedures of this study were executed for extracting useful knowledge from part of the Tehran geodemographic data. Integrating SDM techniques based on SOA and distributed computing provides high performance and interoperable spatial web services that can be used in many applications.

The findings should make an important contribution to geospatial web service, SDI, and geographic knowledge discovery (GKD) fields. Compared to traditional SDM procedures, the proposed solution allows a dynamic, easier, and much faster practice to perform the SDM technique. Due to the interoperable components of the proposed framework, this method is particularly useful in spatial knowledge sharing. This means that a decision-maker can use a combination of KDWS operations to answer unstructured questions, also hoping for service orchestration or composition to obtain more valuable knowledge. The findings of this study suggest that web service-based SDM techniques can be used for geographic knowledge discovery.

Although the proposed framework provides advantages for GIS societies, it also includes certain limitations. In this study, loading SDIs data to HDFS applied by ETL tools as well as multiple manual coding and scripts was used. However, the repetition of such manual operations is a time-consuming process, especially when hundreds or thousands

of clusters exist. In this regard, the automation of such manual operations is an essential mechanism. Automation data integration and ETL process through workflows or a user-friendly graphical interface can solve this problem. Moreover, geographic data especially in the environmental field are inherently stored in the raster data model, but this type of data model is neglected in the developed service. Strategies to enhance the performance of the methodology should be involved in future studies. The proposed solution can be extended to support other spatial mining techniques. Additionally, while this study focuses more on the feasibility of extracting knowledge from SDIs using interoperable services, the speed and performance of this kind of service could be tackled when the voluminous dataset is used (gigabytes or terabytes of a spatial dataset). More broadly, research is also needed to determine the semantic problems of spatial data in the proposed solution. To integrate heterogeneous and distributed spatial data in different contexts, the framework can be by considered geo-ontologies to describe semantic relations of big spatial data.

In future work, utilization of the proposed solution from different visions includes knowledge extraction, sharing, and the composition can be implemented or evaluated in a geoportal. A greater focus on procedures to integrate or the composition of various KDWS services could produce interesting findings, which could be very important in the future. The question raised by this study is how to combine web service-based SDM techniques and gain more important knowledge (knowledge about knowledge). Another possible area of future research would be to investigate a web service-based SDM solution for a raster-based data model.

Author Contributions: Conceptualization, Morteza Omidipoor, Ara Toomanian, and Ali Mansourian; Funding acquisi-tion, Ali Mansourian; Investigation, Morteza Omidipoor, Ara Toomanian; Methodology, Morteza Omidipoor, Ali Mansourian, and Ara Toomanian; Project administration, Ara Toomanian and Ali Mansourian; Software, Morteza Omidipoor; Supervision, Ara Toomanian, Ali Mansourian and Najmeh Neysani Samany; Validation, Ara Toomanian; Writing—original draft, Morteza Omi-dipoor; Writing—review and editing, Ali Mansourian, Ara Toomanian, and Najmeh Neysani Samany. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding", however, the authors were supported by the Envi-ronmental Management in the Middle East (EMME) project, funded by European Union to Ali Mansourian, project number: 598189.

Institutional Review Board Statement: The study did not require ethical approval.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kotsev, A.; Minghini, M.; Tomas, R.; Cetl, V.; Lutz, M. From Spatial Data Infrastructures to Data Spaces—A Technological Perspective on the Evolution of European SDIs. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 176. [CrossRef]
2. Andrachuk, M.; Marschke, M.; Hings, C.; Armitage, D. Smartphone technologies supporting community-based environmental monitoring and implementation: A systematic scoping review. *Biol. Conserv.* **2019**, *237*, 430–442. [CrossRef]
3. Brovelli, M.A.; Minghini, M.; Zamboni, G. Public participation in GIS via mobile applications. *ISPRS J. Photogramm. Remote Sens.* **2016**, *114*, 306–315. [CrossRef]
4. Kankanamge, N.; Yigitcanlar, T.; Goonetilleke, A.; Kamruzzaman, M. Can volunteer crowdsourcing reduce disaster risk? A systematic review of the literature. *Int. J. Disaster Risk Reduct.* **2019**, *35*, 101097. [CrossRef]
5. Li, D.; Wang, S.; Yuan, H.; Li, D. Software and applications of spatial data mining. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2016**, *6*, 84–114. [CrossRef]
6. Miller, H.J.; Han, J. *Geographic Data Mining and Knowledge Discovery*; CRC Press: Boca Raton, FL, USA, 2014.
7. Ristoski, P.; Paulheim, H. Semantic Web in data mining and knowledge discovery: A comprehensive survey. *J. Web Semant.* **2016**, *36*, 1–22. [CrossRef]
8. Pashova, L.; Bandrova, T. A brief overview of current status of European spatial data infrastructures—Relevant developments and perspectives for Bulgaria. *Geo-Spat. Inf. Sci.* **2017**, *20*, 97–108. [CrossRef]
9. Gervone, G.; Lin, J.; Waters, N. *Data Mining for Geoinformatics: Methods and Applications*; Springer: New York, NY, USA, 2014.

10. Perumal, M.; Velumani, B.; Sadhasivam, A.; Ramaswamy, K. Spatial Data Mining Approaches for GIS–A Brief Review. In *Emerging ICT for Bridging the Future-Proceedings of the 49th Annual Convention of the Computer Society of India CSI*; AISC: Chicago, IL, USA, 2015; Volume 2. [CrossRef]
11. Shirowzhan, S.; Lim, S.; Trinder, J.; Li, H.; Sepasgozar, S. Data mining for recognition of spatial distribution patterns of building heights using airborne lidar data. *Adv. Eng. Inform.* **2020**, *43*, 101033. [CrossRef]
12. Thach, N.N.; Ngo, D.B.-T.; Xuan-Canh, P.; Hong-Thi, N.; Thi, B.H.; Nhat-Duc, H.; Dieu, T.B. Spatial pattern assessment of tropical forest fire danger at Thuan Chau area (Vietnam) using GIS-based advanced machine learning algorithms: A comparative study. *Ecol. Inform.* **2018**, *46*, 74–85. [CrossRef]
13. Georganos, S.; Grippa, T.; Niang Gadiaga, A.; Linard, C.; Lennert, M.; Vanhuysse, S.; Kalogirou, S. Geographical random forests: A spatial extension of the random forest algorithm to address spatial heterogeneity in remote sensing and population modelling. *Geocarto Int.* **2019**, 1–16. [CrossRef]
14. Ernst, M.; Haesbroeck, G. Comparison of local outlier detection techniques in spatial multivariate data. *Data Min. Knowl. Discov.* **2017**, *31*, 371–399. [CrossRef]
15. Unternährer, J.; Moret, S.; Joost, S.; Maréchal, F. Spatial clustering for district heating integration in urban energy systems: Application to geothermal energy. *Appl. Energy* **2017**, *190*, 749–763. [CrossRef]
16. Blachowski, J. Application of GIS spatial regression methods in assessment of land subsidence in complicated mining conditions: Case study of the Walbrzych coal mine (SW Poland). *Nat. Hazards* **2016**, *84*, 997–1014. [CrossRef]
17. Jayababu, Y.; Varma, G.; Govardhan, A. Incremental topological spatial association rule mining and clustering from geographical datasets using probabilistic approach. *J. King Saud Univ. Comput. Inf. Sci.* **2018**, *30*, 510–523. [CrossRef]
18. Kumar, R.; Jha, S.; Mittal, M.; Goyal, L.M. Spatial data analysis using association rule mining in distributed environments: A privacy prospect. *Spat. Inf. Res.* **2018**, *26*, 629–638. [CrossRef]
19. Alkathiri, M.; Jhummarwala, A.; Potdar, M. Multi-dimensional geospatial data mining in a distributed environment using MapReduce. *J. Big Data* **2019**, *6*, 82. [CrossRef]
20. Omidipoor, M.; Toomanian, A.; Samani, N.N. Towards Spatial Knowledge Infrastructure (SKI): Technological Understanding. In Proceedings of the 21st AGILE International Conference on Geographic Information Science, Lund, Sweden, 12–15 June 2018. Available online: https://www.semanticscholar.org/paper/Towards-Spatial-Knowledge-Infrastructure-(-SKI-)-%3A-Omidipoor/823c974fbdf149e8412d0ae5fe692ef1584bdaf2 (accessed on 28 December 2020).
21. Li, Z.; Gui, Z.; Hofer, B.; Li, Y.; Scheider, S.; Shekhar, S. Geospatial information processing technologies. In *Manual of Digital Earth*; Springer: Singapore, 2020; pp. 191–227.
22. Jo, J.; Lee, K.-W. High-performance geospatial big data processing system based on MapReduce. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 399. [CrossRef]
23. Yao, X.; Mokbel, M.F.; Alarabi, L.; Eldawy, A.; Yang, J.; Yun, W.; Zhu, D. Spatial coding-based approach for partitioning big spatial data in Hadoop. *Comput. Geosci.* **2017**, *106*, 60–67. [CrossRef]
24. Alarabi, L.; Mokbel, M.F.; Musleh, M. St-hadoop: A mapreduce framework for spatio-temporal data. *GeoInformatica* **2018**, *22*, 785–813. [CrossRef]
25. Park, S.; Ko, D.; Song, S. Parallel Insertion and Indexing Method for Large Amount of Spatiotemporal Data Using Dynamic Multilevel Grid Technique. *Appl. Sci.* **2019**, *9*, 4261. [CrossRef]
26. Li, S.; Dragicevic, S.; Castro, F.A.; Sester, M.; Winter, S.; Coltekin, A.; Stein, A. Geospatial big data handling theory and methods: A review and research challenges. *ISPRS J. Photogramm. Remote Sens.* **2016**, *115*, 119–133. [CrossRef]
27. Yu, J.; Zhang, Z.; Sarwat, M. Spatial data management in apache spark: The geospark perspective and beyond. *GeoInformatica* **2019**, *23*, 37–78. [CrossRef]
28. Wagemann, J.; Clements, O.; Marco Figuera, R.; Rossi, A.P.; Mantovani, S. Geospatial web services pave new ways for server-based on-demand access and processing of Big Earth Data. *Int. J. Digit. Earth* **2018**, *11*, 7–25. [CrossRef]
29. Yue, P. *Semantic Web-Based Intelligent Geospatial Web Services*; Springer: New York, NY, USA, 2013.
30. Yue, P.; Di, L.; Yang, W.; Yu, G.; Zhao, P.; Gong, J. Semantic Web Services-based process planning for earth science applications. *Int. J. Geogr. Inf. Sci.* **2009**, *23*, 1139–1163. [CrossRef]
31. Zhang, F.; Chen, M.; Ames, D.P.; Shen, C.; Yue, S.; Wen, Y.; Lü, G. Design and development of a service-oriented wrapper system for sharing and reusing distributed geoanalysis models on the web. *Environ. Model. Softw.* **2019**, *111*, 498–509. [CrossRef]
32. Zhao, P. *Geospatial Web Services: Advances in Information Interoperability: Advances in Information Interoperability*; IGI Global: Hershay, PA, USA, 2010.
33. Chaves, J.T.F.; de Freitas, S.A.A. A Systematic Literature Review for Service-Oriented Architecture and Agile Development. In Proceedings of the International Conference on Computational Science and Its Applications, Saint Petersburg, Russia, 29 June 2019. [CrossRef]
34. Niknejad, N.; Ismail, W.; Ghani, I.; Nazari, B.; Bahari, M. Understanding Service-Oriented Architecture (SOA): A systematic literature review and directions for further investigation. *Inf. Syst.* **2020**. [CrossRef]
35. Chow, T.E. Geography 2.0: A mashup perspective. In *Advances in Web-based GIS, Mapping Services and Applications*; CRC Press: Boca Raton, FL, USA, 2011; pp. 15–36.
36. Li, S.; Dragicevic, S.; Veenendaal, B. *Advances in Web-Based GIS, Mapping Services and Applications*; CRC Press: Boca Raton, FL, USA, 2011.

37. Loreti, D.; Lippi, M.; Torroni, P. Parallelizing Machine Learning as a service for the end-user. *Future Gener. Comput. Syst.* **2020**, *105*, 275–286. [CrossRef]
38. Ribeiro, M.; Grolinger, K.; Capretz, M.A. Mlaas: Machine Learning as a Service. In Proceedings of the 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), Miami, FL, USA, 9–11 December 2015. Available online: https://ieeexplore.ieee.org/document/7424435 (accessed on 28 December 2020).
39. Sun, Z.; Zou, H.; Strang, K. Big data analytics as a service for business intelligence. In Proceedings of the Conference on e-Business, e-Services and e-Society, Delft, The Netherlands, 13–15 October 2015. Available online: https://link.springer.com/chapter/10.1007/978-3-319-25013-7_16 (accessed on 28 December 2020).
40. Wehrle, P.; Miquel, M.; Tchounikine, A. A Grid Services-Oriented Architecture for Efficient Operation of Distributed Data Warehouses on Globus. In Proceedings of the 21st International Conference on Advanced Information Networking and Applications (AINA'07), Niagara Falls, ON, Canada, 21–23 May 2007. Available online: https://www.semanticscholar.org/paper/OLAP-query-processing-for-partitioned-data-Bellatreche-Karlapalem/4719af2994bb45fd9dfd687eebaa2b829b9ab474 (accessed on 28 December 2020).
41. Wu, L.; Barash, G.; Bartolini, C. A Service-Oriented Architecture for Business Intelligence. In Proceedings of the IEEE International Conference on Service-Oriented Computing and Applications (SOCA'07), Newport Beach, CA, USA, 19–20 June 2007. Available online: https://dl.acm.org/doi/10.1109/SOCA.2007.6 (accessed on 28 December 2020).
42. Zorrilla, M.; García-Saiz, D. A service oriented architecture to provide data mining services for non-expert data miners. *Decis. Support Syst.* **2013**, *55*, 399–411. [CrossRef]
43. Medvedev, V.; Kurasova, O.; Bernatavičienė, J.; Treigys, P.; Marcinkevičius, V.; Dzemyda, G. A new web-based solution for modelling data mining processes. *Simul. Model. Pract. Theory* **2017**, *76*, 34–46. [CrossRef]
44. Kusumakumari, V.; Sherigar, D.; Chandran, R.; Patil, N. Frequent pattern mining on stream data using Hadoop CanTree-GTree. *Procedia Comput. Sci.* **2017**, *115*, 266–273. [CrossRef]
45. Golmohammadi, J.; Xie, Y.; Gupta, J.; Li, Y.; Cai, J.; Detor, S.; Shekhar, S. An Introduction to Spatial Data Mining. 2018. Available online: https://conservancy.umn.edu/handle/11299/216029 (accessed on 28 December 2020).
46. Anselin, L. Local indicators of spatial association—LISA. *Geogr. Anal.* **1995**, *27*, 93–115. [CrossRef]
47. Duan, L.; Xu, L.; Guo, F.; Lee, J.; Yan, B. A local-density based spatial clustering algorithm with noise. *Inf. Syst.* **2007**, *32*, 978–986. [CrossRef]
48. Arthur, D.; Vassilvitskii, S. *K-Means++: The Advantages of Careful Seeding*; Stanford University: Stanford, CA, USA, 2006.
49. Zhang, T.; Ramakrishnan, R.; Livny, M. BIRCH: An efficient data clustering method for very large databases. *ACM Sigmod Rec.* **1996**, *25*, 103–114. [CrossRef]
50. Dhillon, I.S.; Guan, Y.; Kulis, B. Kernel K-Means: Spectral Clustering and Normalized Cuts. In Proceedings of the tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 22–25 August 2004. [CrossRef]
51. Murtagh, F.; Legendre, P. Ward's hierarchical agglomerative clustering method: Which algorithms implement Ward's criterion? *J. Classif.* **2009**, *31*, 274–295. [CrossRef]
52. Ankerst, M.; Breunig, M.M.; Kriegel, H.-P.; Sander, J. OPTICS: Ordering points to identify the clustering structure. *ACM Sigmod Rec.* **1999**, *28*, 49–60. [CrossRef]
53. Ester, M.; Kriegel, H.-P.; Sander, J.; Xu, X. *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*; KDD: Stanford, CA, USA, August 1996; Volume 96, pp. 226–231.
54. Frank, R.; Ester, M.; Knobbe, A. A Multi-Relational Approach to Spatial Classification. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 28 June–1 July 2019; KDD: Stanford, CA, USA. [CrossRef]
55. Koperski, K.; Han, J.; Stefanovic, N. An Efficient Two-Step Method for Classification of Spatial Data. In Proceedings of the International Symposium on Spatial Data Handling (SDH'98), Vancouver, BC, Canada, 11–15 July 1998. Available online: https://www.semanticscholar.org/paper/An-Efficient-Two-Step-Method-for-Classification-of-Koperski-Han/c9e10cf4006690e6f3a3c05a151515d0c5a8ca6d (accessed on 28 December 2020).
56. Fan, R.-E.; Chang, K.-W.; Hsieh, C.-J.; Wang, X.-R.; Lin, C.-J. LIBLINEAR: A library for large linear classification. *J. Mach. Learn. Res.* **2008**, *9*, 1871–1874.
57. Breiman, L.; Friedman, J.H.; Olshen, R.A.; Stone, C.J. *Classification and Regression Trees*; CRC Press: Boca Raton, FL, USA, 1984; Volume 432, pp. 151–166.
58. Goldberger, J.; Hinton, G.E.; Roweis, S.T.; Salakhutdinov, R.R. Neighbourhood components analysis. *Adv. Neural Inf. Process. Syst.* **2004**, *17*, 513–520.
59. Geurts, P.; Ernst, D.; Wehenkel, L. Extremely randomized trees. *Mach. Learn.* **2006**, *63*, 3–42. [CrossRef]
60. Whiteside, A. *OGC Implementation Specification 06-121r3: OGC Web Services Common Specification*; Open Geospatial Consortium: Wayland, MA, USA, 2007.
61. Novikov, A. PyClustering: Data mining library. *J. Open Source Softw.* **2019**, *4*, 1230. [CrossRef]
62. Rey, S.J.; Anselin, L. PySAL: A Python library of spatial analytical methods. In *Handbook of Applied Spatial Analysis*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 175–193.

63. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Dubourg, V. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
64. Omidipoor, M.; Jelokhani-Niaraki, M.; Moeinmehr, A.; Sadeghi-Niaraki, A.; Choi, S.-M. A GIS-based decision support system for facilitating participatory urban renewal process. *Land Use Policy* **2019**, *88*, 104150. [CrossRef]