International Journal of
Geo-Information

Article

Geographic Information Retrieval Method for Geography Mark-Up Language Data

Caili Fang [1,2] and Shuliang Zhang [1,3,*]

[1] Key Laboratory of Virtual Geographic Environment of Ministry of Education, Nanjing Normal University, Nanjing 210023, China; fangleheart@henu.edu.cn
[2] College of Computer Information Engineering Henan University, Kaifeng 475001, China
[3] Jiangsu Center for Collaborative Innovation in Geographical Information Resource Development and Application, Nanjing 210023, China
[*] Correspondence: zhangshuliang@njnu.edu.cn

Received: 27 November 2017; Accepted: 7 March 2018; Published: 9 March 2018

Abstract: Geography Mark-up Language (GML) is the geographic information coding specification based on the Extensible Markup Language (XML) technology, which was developed by the Open GIS Consortium (OGC). GML expresses spatial and non-spatial attributes of geographic objects. Retrievals for traditional XML and geographic information have some limitations with respect to GML data, such as mismatching of the retrieval model, a single search form, and low retrieval quality. Based on analysis of the attributes, spatial relations, and structural features of GML data, this paper takes GML data elements as retrieval units and summarizes the GML retrieval mode. Then, the GML retrieval mode is constructed and formalized. On this basis, the GML Geographic Information Retrieval (GML_GIR) model is presented. The method implements the construction of a comprehensive index and the relative ordering of retrieval results by means of Lucene, an open-source full-text retrieval framework, and its components. For different features of GML data, corresponding relevance calculations are proposed. This study designs several different retrieval forms for GML data and simplifies the process of user information acquisitions. It provides reference methods for exploring geographical information retrieval based on semi-structured data represented by GML. Experimental results showed the efficiency and accuracy of the retrieval method.

Keywords: GML data features; GML_GIR model; retrieval mode; comprehensive index construction; relevance calculations

1. Introduction

With the implementation of the Geography Markup Language (GML) International Standard (ISO 19136-2007), the popularity of Web Feature Services (WFS) has increased. Accordingly, the GML data generated by artificial or spatial services have surfaced on the Internet and have been extended to many data specifications, such as the Keyhole Markup Language (KML), CityGML, and WFS. They are used for many kinds of applications such as indoor navigation [1], location-based services, and other aspects. The data mainly exist in two forms for practical applications. The first is document-type GML data stored locally in the form of text, such as the Aeronautical Information Exchange Model (AIXM) [2] based on GML data, the urban 3D modeling based CityGML [3], etc.; the second is service-type GML data stored on the web server in the form of data services, such as WFS and Web Feature Gazetteer Services (WFS-G). These data specifications are derived from GML/Extensible Markup Language (XML) and have complex nested relationships between elements. The data includes rich text, spatial information, and significant structural information. They have typical semi-structural features, and then GML or similar to GML data has been generated. These spatial data exist in the form of GML documents. With the advent of GIS big data, these resources have increased dramatically and been more complex. In the process of geographic information retrieval (GIR), the complexity of geospatial data is also a problem [4]. Existing retrieval methods have made great achievements in the field of GIR. However, for these resources, there is a lack of methods and models suitable for GML data; we need to provide new methods for GML data, and further studies are essential.

GML is the specific application of XML in the field of geographic information. XML information retrieval has made great achievements. The retrieval methods can be expanded in terms of keywords, structures, and query languages. Retrieval based on keywords involves many aspects, such as the fuzzy expression of keywords [5], inference of user intention based on keywords semantics [6–8], keywords retrieval among multiple documents [9], and other aspects. These retrievals can return relatively complete XML fragments that align with users' intent; their study may be more reasonable if they considered the correlation ranking. Retrievals based on structures usually start with the path definitions of XML, such as the path constraints [10,11], clustering analysis of different structural documents [12], and retrievals based on the XML document's structure and contents [13], all of which can provide rich retrieval functions. Their results could be more convincing if they included the relationships between types of elements, nodes, and locations. Retrievals based on query languages, such as Xpath and XQuery, integrate XML and its extension language into the XML retrieval system. These methods further enrich the forms of XML retrieval; unfortunately, only users who understand the query language and structural information of XML documents can use the retrieval system. For attribute information retrieval of GML data, although these methods provide methodological references, XML lacks the spatial information of GML data, and XML query methods and theories cannot completely solve GML data spatial retrieval.

Compared to XML, retrievals for GML geographic data have added methods for spatial data and spatial relations. At present, the GML retrieval includes SQL-based queries and extended XML queries. Corcoles et al. [14] proposed a GML spatial query language based on SQL. It represents the parent–child relationship of XML elements through nodes. Boucelma et al. [15] proposed the GML spatial query method based on XQuery. They used the Java Topology Suite (JTS), an API for processing geographic data, to parse the basic geometry object model. The approach would be more helpful if it avoided conversions between text and spatial formats. Jesus et al. [16] used XPath to process the semantic structure of GML documents. Accordingly, XPath was converted into SQL statements and solved the query of city maps. Finally, for visualization, the query was also transformed into the KML format.

From the perspective of data compression, Savary et al. [17] optimized the GML spatial queries and enhanced its efficiency. Lan et al. [18] proposed GMLXQL and GQ, which are based on XQuery syntax, to add space-related models and operations. Guan et al. [19] expanded the XQuery data model and formal semantics, thereby increasing the query and spatial topology based on geometric elements. Their studies may be more easily accepted by more users if the complexity of the query language and operations was reduced. Moreover, in the real-estate cadaster field, Tong et al. [20] constructed GML application schema and built queries of simple objects and object spatial relations for cadastral data, making the GML application scope wider.

By combining traditional text information retrieval, the project SPIRIT [21] used space-text indexing, which combines the location of the document with the text index. Thus, it can determine the geographical location of the document site by including the spatial index. It then combines the geographical grid with the text and sorts it with an inverted list. This approach is essentially a text index; however, for geographical information, a more complex retrieval is needed. Cai et al. [22] developed a GeoVSM system based on geographical information and text retrieval. Each document of a corpus is indexed in a geographic coordinate space and word item space. It is then queried in the two spaces and the results are fused. Buscaldi et al. [23] studied the diversification of queries in GIR. The method adds an extended index, whereby each place name contains its geographical scope, such as Europe, whose extension index is the United Kingdom, Germany, and other European countries. It then refines the query scope for improved clarity.

In summary, there are some problems with GML data retrieval. First, query methods based on SQL and XQuery extension are not suitable for GML data retrieval. For extended query methods based on SQL, there is a notably different structure between the GML data and structured data. For extended query methods based on XQuery, they only support the string type. Furthermore, for GIR, the query syntax becomes complex. These methods need to construct complex query expressions, and only the professional personnel could operate them; for ordinary users, the query application is difficult and requires high costs, so we need to design a method that can reduce the complexity [24] of retrieval and reflect the features of GML data retrieval. The second problem pertains to the lack of a GML retrieval model. Although XML document retrieval methods can perform spatial information retrieval, they do not consider the relationship between GML data attributes and structural features. The searching mode is simple (for example, syntax complexity), and the efficiency is lower than the corresponding text retrieval. A mature model is needed to support GML data retrieval. Finally, existing methods do not support GML and similar GML data retrieval. At present, research on GIR mainly focuses on a specific data format, or structured geographic information data for semi-structural geospatial data, all of which are described in the form of tag pairs, such as GML, KML, SVG, WFS-G, and CityGML. Moreover, other research focuses on text retrieval methods and lacks a methodology to support the two existing types of GML data.

To address the above limitations, this paper presents the GML Geographic Information Retrieval (GML_GIR) model and retrieval method for GML data. It realizes the diversification of search styles, returns more relevant retrieval results, and enriches the contents and forms of GIR.

2. GML Document Data and Retrieval Mode Construction

2.1. Analysis of GML Document Data

The GML specification consists of three parts: the core schema, the application schema, and the instance document. The core schema is defined by OGC, which provides a basic framework for describing geographic objects and defining basic types and elements. However, it does not define specific elements of the real world, such as roads and political boundaries. The application schema is defined by users. To express the real world, according to the core schema, it creates an application schema of a specific area (for example, roads) to define a set of geographical objects in this area. The instance document is formed by instantiating the application schema document.

For example, Road.xsd of a road element (Figure 1a) includes a content model, wherein the structure of the road element instance, LRDL, is organized. The LRDL element type describes many spatial and non-spatial attributes. In the form of tagged pairs, these attributes define the data structure of the road element. Non-spatial attributes include the road name, identification, width, material, and so on, as well as numeric attributes (OBJECTID and WIDTH) and text attributes (MATRL and NAME). Spatial attributes directly use the pre-defined type of GML geometry schema, such as acurveProperty and multiCurveProperty. LRDLType, inherited from the element schema, illustrates the application of the road data to the element model. In the instance document, Road.gml (Figure 1b), the instance data starts from the tag FeatureCollection. The tag boundedBy describes the spatial scope of the element, and each element is organized in the form of a featureMember tag pair, forming a collection of elements that embody the nested form. Corresponding to the application schema, elements in the instance data contain all the attributes described in the application schema. Each attribute has a definite value included in attribute tags, e.g., the WIDTH value is 24, and the posList attribute value of the LingString object is the coordinate sequence.

According to the GML application schema and the instance document, GML encapsulates the spatial information and its attributes. The differences between GML and traditional data characteristics are as follows:

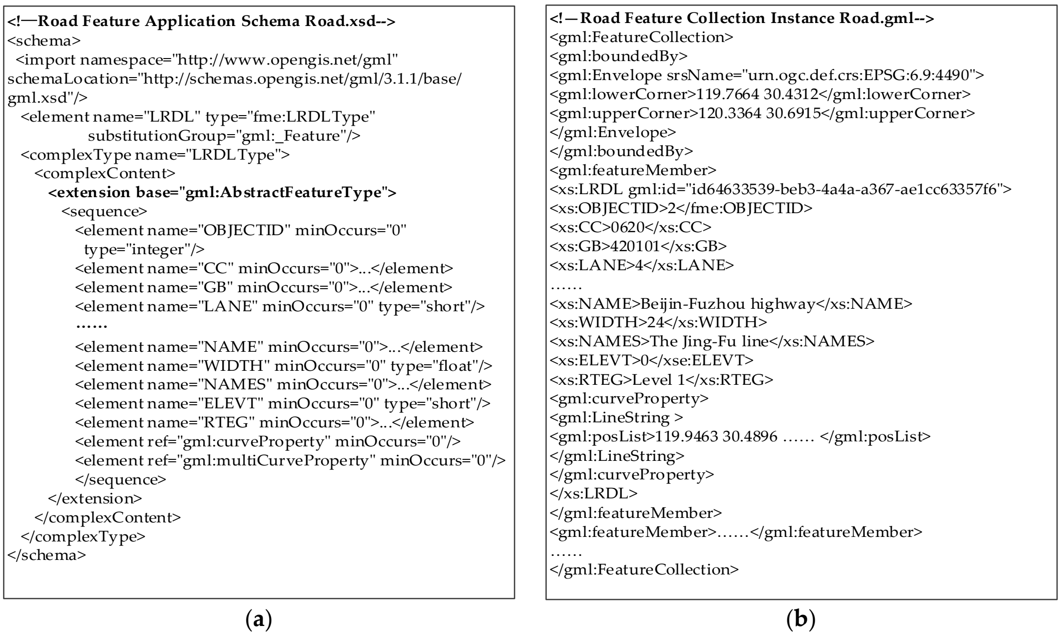
1. GML document instance data are usually composed of one or more feature collections. The feature collections contain a series of member elements, and elements are expressed by the

attributes that include spatial and non-spatial attributes. Elements can be specific physical objects, such as roads and rivers.

2. Non-spatial attribute features are an important semantic to identify GML data and express them in text, such as the road name and grade. Their types are diverse and expressed as texts, numbers, and dates.

3. The topological relationships among GML elements can be described by mapping topological primitives (node, edge, face, and toposolid) to geometric primitives (point, line, polygon, and solid).

4. The GML document has an obvious structural feature. From Figure 1, GML data are described in tag forms. By parsing the GML application schema, the tree structure of GML document data can be established. Thus, the GML data structure can be divided into internal and external features of the elements. The internal features refer to the number, name, type, and location of the attribute node; they can also be called element structures. The external features are represented by the location of the element nodes in the GML document tree and the relationship between the nodes. They are usually referred to as path structures, with the path expressions of element nodes. For example, searching "gml:FeatureCollection/gml:featureMember/fme:YL" expresses the YL location and relationship to other nodes.



```
<!—Road Feature Application Schema Road.xsd-->
<schema>
 <import namespace="http://www.opengis.net/gml"
schemaLocation="http://schemas.opengis.net/gml/3.1.1/base/
gml.xsd"/>
  <element name="LRDL" type="fme:LRDLType"
            substitutionGroup="gml:_Feature"/>
  <complexType name="LRDLType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence>
          <element name="OBJECTID" minOccurs="0"
           type="integer"/>
          <element name="CC" minOccurs="0">...</element>
          <element name="GB" minOccurs="0">...</element>
          <element name="LANE" minOccurs="0" type="short"/>
          ……
          <element name="NAME" minOccurs="0">...</element>
          <element name="WIDTH" minOccurs="0" type="float"/>
          <element name="NAMES" minOccurs="0">...</element>
          <element name="ELEVT" minOccurs="0" type="short"/>
          <element name="RTEG" minOccurs="0">...</element>
          <element ref="gml:curveProperty" minOccurs="0"/>
          <element ref="gml:multiCurveProperty" minOccurs="0"/>
          </sequence>
      </extension>
    </complexContent>
  </complexType>
</schema>
```

(**a**)

```
<!—Road Feature Collection Instance Road.gml-->
<gml:FeatureCollection>
<gml:boundedBy>
<gml:Envelope srsName="urn.ogc.def.crs:EPSG:6.9:4490">
<gml:lowerCorner>119.7664 30.4312</gml:lowerCorner>
<gml:upperCorner>120.3364 30.6915</gml:upperCorner>
</gml:Envelope>
</gml:boundedBy>
<gml:featureMember>
<xs:LRDL gml:id="id64633539-beb3-4a4a-a367-ae1cc63357f6">
<xs:OBJECTID>2</fme:OBJECTID>
<xs:CC>0620</xs:CC>
<xs:GB>420101</xs:GB>
<xs:LANE>4</xs:LANE>
……
<xs:NAME>Beijin-Fuzhou highway</xs:NAME>
<xs:WIDTH>24</xs:WIDTH>
<xs:NAMES>The Jing-Fu line</xs:NAMES>
<xs:ELEVT>0</xse:ELEVT>
<xs:RTEG>Level 1</xs:RTEG>
<gml:curveProperty>
<gml:LineString >
<gml:posList>119.9463 30.4896 …… </gml:posList>
</gml:LineString>
</gml:curveProperty>
</xs:LRDL>
</gml:featureMember>
<gml:featureMember>……</gml:featureMember>
……
</gml:FeatureCollection>
```

(**b**)

Figure 1. (**a**) Road application schema file; (**b**) road instance data document.

2.2. GML Data Feature and Retrieval Mode Construction

According to Section 2.1, analysis of GML retrieval should follow the GML data features. It can be considered in three ways.

1. GML non-spatial attributes is an important aspect of GIR. Retrieval forms usually contain the attributes text retrieval and attributes number range retrieval. Attribute types are generally text, numbers, and dates. Thus, attribute retrieval can be studied qualitatively and quantitatively, as shown in Figure 2. For example, we need some retrievals such as "gas stations" or "scenic spots above grade 4A".

2. GML spatial feature retrieval is mostly related to practical applications. For example, searching "farmland to the north of a town", which expresses the spatial relation retrieval for complex area elements. For metrical relationships, "near" and "around" are qualitative expressions, and "within 50 m" and "2 km east" are quantitative expressions. Because the qualitative measurement relationship

relates to the concept of spatial cognition and spatial scale, this paper only considers the qualitative measurement relation.

3. As described in Section 2.1, the retrieval of GML data structural features can be divided into element structures and path structures, such as searching road elements, according to GML path information, to determine the path expressions: /Root/FeatureCollection/RoadFeatureMember. All sub-elements of the element collection can be searched. Through the above, we can generalize the model of GML data features as shown in Figure 2.

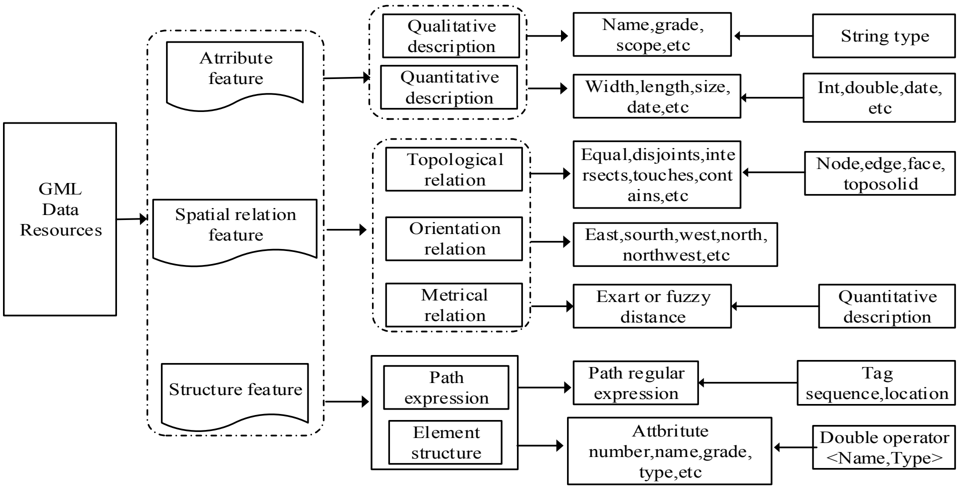


Figure 2. Model of GML data features.

With attributes, spatial features, and structural features, GML retrieval should integrate GML feature items into an index system, where GML feature items are used as index feature items. Therefore, based on the features of GML data and retrieval requirements, we summarize the common types of retrieval modes.

The model of GML data features provides the basis for the retrieval model and method based on GML data features. Similarly, GML geographical information retrieval is classified into three types: attribute-based retrieval, spatial-feature-based retrieval, and structure-based retrieval. These three types can be combined to form a richer retrieval mode. Examples of GML data retrieval mode are shown in Table 1.

Table 1. Examples GML data retrieval mode.

Retrieval Features	Classification of Retrieval Modes	Examples
Attribute features	Text attributes	Name: supermarkets
	Number attributes	Length: the road of less than 3 km
Spatial features + others	Topology + attributes	Name: gas stations Spatial constraint: in Wukang town
	Topology + measurement + attributes	Name: restaurants Spatial constraint: a park east 100 m
	Topology + directions + attributes	Name: highways Width: more than 20 m Spatial constraint: Ning-Hang high speed rail east and intersecting with it
	Topology + measurement + composite attributes	Name: hotels Star level: above 3-star Spatial constraint: the park east and within 2 km
Structure features	Element structures	Searching linear features that include NAME, WIDTH attributes: <field>NAME,string</field> <field>WIDTH,integer</field>
	Path structures	Searching POI elements: /ROOT/FeatureCollection/POI/FeatureMember/*

Unlike traditional keyword-based GIR, as shown in Table 1, GML data retrieval offers retrieval mode diversity. The diversity changes the current status of inputting contents, which not only supports the geographical element name as retrieval conditions, but also increases the retrieval support for geographical relations and GML data structure features.

2.3. Extraction of GML Information Retrieval Granularity and Definition

GML geographic objects are interrelated and nested, which is a key issue in extracting GML data information and dividing the index granularity in GML information retrieval. GML document instance data are typically composed of one or more feature collections. These collections include a series of element members or another feature collection by the "featuremember attribute" or "featuremembers attribute" tag. Elements are described by attributes. GML element attributes include geometric and non-spatial attributes. If the element collection is taken as a basic extraction unit, it is equivalent to taking the element collection contents of each DML instance document and its sub-element with its contents as a whole object. Then it can be operated as a whole; however, the object already includes many complete elements. Thus, for retrieval and index construction, the retrieval granularity is too large and users will not obtain the required information. If we take the spatial and non-spatial attributes of elements to be extraction units, they are only partly descriptive of the elements and do not indicate the complete contents. Therefore, the retrieval results will have no real meaning for users. Therefore, an appropriate granularity should be in the middle of the two. The GML element is taken as a basic unit of extracting a GML instance document fragment. It uses the element contents and attributes as a complete, inseparable object, constructs the index, and returns the retrieval results.

3. GML_GIR Retrieval Model

GML data has the above three kinds of features. Attribute information comprises text, numbers, dates, and other types of information. Spatial information involves many spatial relations, such as geometric, topological, and spatial. Structural information consists of the element structure and the element paths. Therefore, how to combine the information with various features and how to describe GML data retrieval in a complete way is a key issue for GML information retrieval. To retrieve contents that are both text-related and geography-related, the GML_GIR retrieval model is proposed. The basic idea of the model is to take the GML data element as a unit to extract the GML information and design the constraints of query conditions. A comprehensive evaluation algorithm based on attributes, spatial relations, and structure correlations is established. It determines the correlation between the retrieval conditions and the retrieval objects. Furthermore, retrieval results are sorted by a relevance score.

The GML_GIR model can be expressed by the quadruples M, which includes the GML data resources, retrieval conditions, calculation methods, and retrieval results.

Definition 1. $M = \langle D, Q, F, R \rangle$**:** *D is the GML data resources set; d is the retrieval data feature of GML data; and Q is the set of retrieval conditions that corresponds to the retrieval mode, and the feature items of retrieval conditions are represented as q. In addition, F is the calculation method that calculates the relation between q and d, and R is the retrieval results.*

D: <Elements, Features of elements' attributes, Element spatial features, Element structure features>

Q: <Attribute text conditions, Attribute numerical conditions, Spatial relation conditions, Structure conditions>

R: < G, S>, G is the result set, including GML geographical element fragments; S is the relevance score set; and the retrieval results are sorted by the relevance scores.

F: It can be represented as follows:

$$Sim\_GML(q,d) = \omega_1\ Sim\_Pr(q,d) + \omega_2\ Sim\_St(q,d)\ + \omega_3\ Sim\_Sp(q,d), \tag{1}$$

where $Sim\_Pr(q,d)$ is the attribute relevance, $Sim\_St(q,d)$ is the structure relevance, $Sim\_Sp(q,d)$ is the spatial relevance, ω_1, ω_2, and ω_3 are the weight factors. ω is a measure of the importance of the three factors to the result of queries. Ren et al. [25] and Cardoso et al. [26] all presented two similar indicators (text and geographic) for GIR, and the weights are set to be equal. They have proved the weight setting is more suitable for these indictors. Based on the two references, and for GIR, the spatial feature is relatively important. We thus suppose ω_3 is 0.5, and ω_1 and ω_2 are both 0.25.

The relevance calculation expresses the degree of matching between the objects of retrieval conditions and the target objects. It is quantitatively represented by the similarity calculation method between the retrieval feature items and the indexing feature items.

3.1. Attribute Relevance Calculation

As an import part of GML element data, the types of attributes are divided into simple and complex attributes. The simple attribute is usually a basic data type, such as text, numbers (int, float, etc.), or dates. The complex attribute itself is usually a GML object. Regardless of whether it is a simple or complex attribute, the attributes can be decomposed into key-value pairs, which are composed of basic data types, by means of GML application schemas. Therefore, the process for the GML data attributes information is equivalent to the process for basic data types. There are three types, as outlined below.

1. For the retrieval of the textual type, it adopts a full-text retrieval method that uses a spatial vector model to calculate the textual relevance [27].

2. For numeric and date attributes, due to the date being converted into numerical data, the date type is essentially the same as it is for numerical attributes. The retrieval of numerical attributes is generally based on numerical range retrieval, for example, retrieving highways with a length from 100 km to 500 km.

3. For retrieval of both textual and numerical attributes, normally the numerical condition is used as the retrieval constraint. The next correlation calculation can be performed such that only the numerical condition is satisfied. This method also has its drawbacks. For example, although the range of values is very close, it does not accurately meet the data constraints. When results are returned, the method will filter the close results, which eventually leads to fewer results. Therefore, considering that the retrieval importance levels of textual and numerical attributes are equal, this paper presents the comprehensive attribute correlation calculation method based on textual and numerical attributes. The attributes relevance calculation formula is as follows:

$$Sim\_Pr(q,d) = (Sim\_T(q,d) + Sim\_N(q,d))/2, \tag{2}$$

where $Sim\_T(q,d)$ is the textual attribute relevance and adopts a full-text retrieval method by taking the vector spatial model to calculate the attribute text relevance [27]. $Sim\_N(q,d)$ is the numerical attribute relevance. In retrieval conditions, the numerical retrieval usually includes exact and range types, such as "4A scenic spots" and "the length of a road is greater than 20 km". The formula is as follows:

$$Sim\_N(q,d) = \begin{cases} 1 & p_n \in q_n \\ \frac{1}{1+|p_n - q_n|} & p_n \notin q_n \end{cases}, \tag{3}$$

where p_n is the value of numerical feature and q_n is the numerical constraint conditions of retrieval conditions. If the retrieval conditions of the objects are satisfied, the relevance value is one; otherwise, it decreases according to the degree of deviation. Thus, during the retrieval, it will not filter more objects with close values.

3.2. Spatial Relation Relevance Calculation

Spatial relations are the spatial characteristic relationships existing among geographic entities, such as topological relations, metric relations, and orientation relations, which are the basis of spatial data analysis, reasoning, and application. These relations between geographical elements make them mutually interrelate, mutually influence, and mutually restrict. In GML retrieval, the single retrieval of topological relations or metric relations may filter out some retrieval results, even if they meet the requirements. Therefore, more results conforming to retrieval conditions would be obtained by establishing the correlation operating of multiple relations. The spatial relation relevance is the matching relation between the spatial relation constraints and target geographic objects in retrieval conditions. The matching degree (or values) can be expressed by a quantitative calculation method. According to the spatial features of the GML data, retrieval mode, and information on the geographical relation, this study uses basic factors of spatial relations to reflect the geographical relevance. The formula is designed as follows:

$$Sim\_Sp(q,d) = w(q,d)\sum_{i=1}^{n} Sim\_B(q,d)/n, \tag{4}$$

where $Sim\_B(q,d)$ is the spatial relation basic factor, and n is the number of spatial-relation basic factors involved in the retrieval conditions. where $Sim\_B(q,d)$ is a spatial relations basic factor, including topological, directional, and metrical relations. We filtered the target objects through $Sim\_B(q,d)$. If retrieval conditions include many kinds of spatial relation constraints, then the relevance calculation uses their average value, and $Sim\_B(q,d)$ is defined as follows:

$$Sim\_B(q,d) = \begin{cases} Sim\_Top(q,d) \\ Sim\_Dir(q,d) \\ Sim\_Dis(q,d) \end{cases}, \tag{5}$$

where $Sim\_Top(q,d)$, $Sim\_Dir(q,d)$, and $Sim\_Dis(q,d)$ are the topological, directional, and metrical relation relevance calculations, respectively. Each value range is [0,1]. $w(q,d)$ is the spatial proximity factor, which is regarded as a distance relation parameter. The closer the distance, the closer the spatial connection. Accordingly, for geographic information retrieval, if an object met any of the same spatial relationship constraints, users generally select a searching target that ranges from near to distant. Therefore, based on Tobler's Fist Law, near things are more related than distant things; this paper defines a measure of spatial proximity. The retrieval target with the closer distance can obtain a higher spatial proximity value in the calculation. For example, in the case of a consistent direction, the closer the distance is, the more similar it is, which is also the case with the topological and metrical relations. Thus, $w(q,d)$ is:

$$w(q,d) = 1 - \left(\frac{D(q,d)}{\max(D(q,d_i)) + 1}\right), \tag{6}$$

where $D(q,d)$ is the distance parameter between geographical objects and indexing retrieval objects in retrieval conditions, and adopts the Euclidean distance. In addition, $\max(D(q,d_i))$ is the maximum distance between geographical objects in retrieval conditions and searching target objects.

1. Topological relevance calculation ($Sim\_Top(q,d)$)

Topology is the basic spatial relationship for expressing geographical spatial structure. Bruns et al. [28] proposed the conceptual of neighborhood graphs of topological relations. By calculating the distance between two topological relations, the neighborhood difference matrix of the topological relations between the two regions is obtained. Based on the matrix, it uses quantitative topological relations to calculate the relevance of topological relations. In the quantitative approach, the matrix defines the difference between topological concepts. However, there is a great difference between these topological concepts; for example, between "disjoint", "contain", and "contained", the difference is four; the difference of topological concepts between “intersects” and “covered by” is two, and so on.

Referencing the difference matrix and combining the relations between topological concepts of the retrieval mode of GML information, we designed a similar matrix of topological relation concepts in GML information retrieval. The similar matrix modifies the matrix of [28]. For example, it defines the similarity value of the topological relation of the difference distance of four as zero. Considering the actual application scenario, for the topological relation similarity of the difference distance from one to three, it was grouped and categorized, reducing the difference distance. For example, it defined the similarity as 0.5 for topological relations of the difference distance from one to three; it defined the similarity as 1 for the difference distance of zero.

Using this approach, the conceptual similar matrix of topological relations was designed and fully considers the relationship between topological relations and their applicability, as shown in Table 2. For example, in retrieving “the roads with Bei-Fu road intersecting, “the retrieval condition is ”intersects”. The topological relation between the roads, which is the closest to the possible matching results, and the “Bei-Fu” road is shown in the third row in Table 2.

Table 2. Conceptual similar matrix of topological relations.

Target Object Retrieval Condition	Disjoints	Touches	Intersects	Equals	Contains	Within
Disjoints	1	0.5	0.5	0.5	0	0
Touches		1	0.5	0.25	0.25	0.25
Intersects			1	0.5	0.25	0.25
Equals				1	0.5	0.5
Contains					1	0
Within						1

According to the concept similar matrix of topological relations, the topological relations similar function is:

$$Sim\_Top(q,d) = \begin{cases} 1 & Topological\ relations\ are\ consistent \\ \alpha & Topological\ relations\ are\ similar \\ 0 & Topological\ relations\ are\ inconsistent \end{cases}, \tag{7}$$

where α is the concept similarity value of topological relations, which is defined by the matrix.

2. Direction relevance calculation ($Sim\_Dir(q,d)$)

The relevance of direction relations is a quantitative method to measure the similarity between the target objects and the reference objects in the spatial direction. In relevance calculations of direction relations, the direction concept similar matrix is the typical method that is used. Goyal et al. [29] proposed the concept of the spatial direction distance, which uses conceptual grids to divide different spatial directions. They translated the target direction into the moving shortest distance of another direction, which is described by the conceptual nearest-neighbor distance. However, the computational granularity is much greater and cannot accurately express the continuous change of direction. We propose the method of direction angle offsets to calculate the spatial direction relations.

As shown in Figure 3, when calculating the direction relevance of target object A and reference object B, we can calculate the angle between vectors RA and RB. A smaller angle indicates higher similarity, and an angle of more than 90 degrees is not relevant. The same holds true for north and east–north, and so on. We only need to calculate the angle between the connecting line of the target and the reference object with the retrieval direction. This method can calculate the relevance of continuous change of different directions and significantly improve the filtering ability of the retrieval results, thereby ensuring the accuracy of the retrieval results.

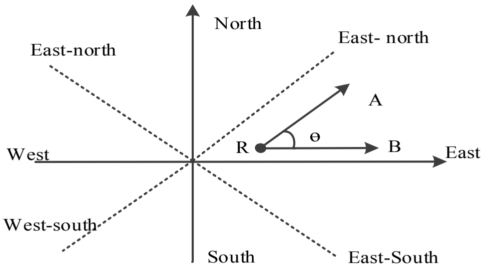


Figure 3. Direction relations diagram.

According to the method of the direction angle offset, the relevance function of spatial directions is defined as follows:

$$Sim\_Dir(q,d) = \begin{cases} 0, & \theta \geq 90^\circ \\ \cos\theta = |\overrightarrow{RA}\ \overrightarrow{RB}| & \theta < 90^\circ \end{cases}, \tag{8}$$

where $\cos\theta$ is the value of the relevance of the direction relations, which is decided by the cosine value of θ. Here, θ is the offset angle between retrieval object A and constraint direction $\overrightarrow{RB}$ in searching conditions of reference object R.

3. Metric relation relevance calculation ($Sim\_Dis(q,d)$)

When GML geographic information is retrieved based on the distance relationship, taking the distance parameter as the influence factor of the spatial relations relevance, usually the farther the distance is, the smaller the relevance. For example, one case is searching for a "hospital within 1 km from the park". Here, the nearer the target is, the more fully it meets the retrieval conditions. Therefore, it takes the reciprocal of the distance to calculate the relevance. The formula is:

$$Sim\_Dis(q,d) = \frac{1}{1 + ed\left(S_q, L_d\right)}, \tag{9}$$

where $ed\left(S_q, L_d\right)$ is the Euclidean distance between the target objects and reference objects. The larger $ed\left(S_q, L_d\right)$ is, the smaller the relevance between the target objects and reference objects.

Another case is the retrieval within the distance range between the target objects and reference objects, such as searching for "bus stops within 1 to 2 km from the subway station". For this retrieval, if the target objects are within the distance range, they meet the retrieval requirement and the relevance value is one. If they are not within the range, the greater the range distance is, the smaller the relevance. The formula is:

$$Sim\_Dis'(q,d) = \begin{cases} 1, & D \in Q \\ \frac{1}{1+|D-Q|}, & D \notin Q \end{cases}, \tag{10}$$

where D is the Euclidean distance between the target and reference objects, and Q is the distance range value set by the retrieval conditions. When D meets Q, the relevance value is one; otherwise, the greater the deviation from the retrieval range, the smaller the correlation.

3.3. Structure Relevance Calculation

As described in Section 2.2, GML data structures include the element structure and path structure. These are more specific features compared to traditional geographical data. By restricting the structure, retrieval methods of GML information can retrieve the data of the corresponding category, a problem that is often encountered in retrieval. Combining the element structure and the path structure, the calculation of the structure relevance is as follows:

$$Sim{-}St(q,d) = (Sim\_Fs(q,d) + Sim\_P(q,d))/2, \tag{11}$$

where $Sim\_Fs(q,d)$ and $Sim\_P(q,d)$ represent the element structure relevance and the path structure relevance, Respectively.

1. Element structure relevance calculation:

For inner attributes of element structures, the contents that the element structure express are the name and type of elements. They are described by 2-tuple < Name,Type>. Many element attributes consist of a 2-tuple collection. The computational process is relevance matching, where the matching objects are the element structure information of the index database and the retrieval conditions. The function is as follows:

$$Sim\_Fs(q,d) = \frac{\sum_{i=1}^{Len(q)} S(q(i))}{\max(Len(q), Len(d))}, \tag{12}$$

where $Len(*)$ is the 2-tuple group number of attribute names and types of retrieval conditions, and $Len(d)$ is the 2-tuple group number of attribute names and types of target elements. $S(q(i))$ is the matching degree of the i-th 2-tuple group in retrieval conditions. If the name and type are both matched, the matching value is 1; if only the name is matching, the value is 0.5; all others are 0. The matching value range is [0,1].

2. Path structure relevance calculation:

The path structure can be abstracted as a tag sequence model; a tag sequence expresses a path from the GML root node to a leaf node. For example, the path of a DOM document tree, wherein a GML element node resides, is /FeatureCollection/featureMember/SFCP. The path structure expresses the path tag sequence as (FeatureCollection, featureMember, SFCP). The structure relevance is calculated using the location information of the common sequence tags [30]. However, when the path structures are matched, the label (sub-sequence) location of a sequence would be a factor that influences the path matching. Therefore, the function of structural relevance is as follows:

$$Sim\_P(q,d) = \omega\ St(p_i, q_j) + (1-\omega)\ Sp(p_i, q_j), \tag{13}$$

where p_i and q_j are the tag sequences of the path structures, $p_i = (t_1, t_2, \ldots t_m)$, $q_j = (g_1, g_2, \ldots g_m)$, ω shows the different importance of the two parts in the similarity of path structure. $St(p_i, q_j)$ is the similarity of the tag sequences between retrieval conditions and target objects. The function is defined as:

$$St(p_i, q_j) = \frac{Len(C(p_i, q_j))}{\max(Len(p_i), Len(q_j))}, \tag{14}$$

where $C(p_i, q_j)$ is the common tags sequence of p_i and q_j, and $Len(p_i)$ is the tag sequence numbers of p_i. The location similarity is defined as:

$$Sp(p_i, q_j) = \frac{1}{1 + \sum_{k=1}^{Len(C(p_i,q_j))} |md(k)|}, \tag{15}$$

where $Len(C(p_i, q_j))$ is the length of $C(p_i, q_j)$. Suppose the k-th item of $C(p_i, q_j)$ is t_k, then, p_l is the longest tag sequence between p_i and q_j, and $md(k)$ is the location offset value of t_k relevance to p_l, and the range of location similarity is [0,1]. For example, $p_i = (a, b, c, d)$, $q_j = (a, b, d, e, c)$, suppose ω = 0.5, then, the path structure similarity is $Sim\_P(p_i, q_j) = 0.5 * (4/5) + 0.5 * 1/(1 + 0 + 0 + 2 + 1) = 0.525$.

4. Retrieval Implementation Based on GML_GIR

4.1. Process for GML Data Retrieval

Based on the GML_GIR model, we designed a GML geo-information retrieval system by means of Lucene, an open-source, full-text retrieval framework. The design is as follows: First, a GML instance document is parsed according to GML data application schema. Taking GML elements as the basic unit, data are obtained such as, GML data elements, spatial attributes, non-spatial attributes, and structural information. Second, with the help of Lucene and its related technology, an integrated index of GML data is generated, along with textual, numerical, spatial, and structural features. Third, by adding Lucene Spatial components to enable spatial retrieval, spatial information of GML data can be filtered according to the spatial relationship. Finally, based on the correlation scoring mechanism of the Lucene retrieval results and the GML_GIR retrieval model, the GML data retrieval results are sorted.

This paper uses Lucene and other open-source technology solutions to support the realization of the GML_GIR retrieval model. Its process is shown in Figure 4. GIR is mainly divided into index construction and relevance ranking of retrieval results. For retrieval of GML data, we take the GML element as the basic unit. This should extract element segments of GML data and analyze their features. It is combined with the interface of text and spatial indexes supported by Lucene. For the text index, it uses IKAnayler, an open-source Chinese segmentation component. For the spatial index, Lucene Spatial supports geometry types, such as points, multi-points, lines, multi-lines, and rectangles.

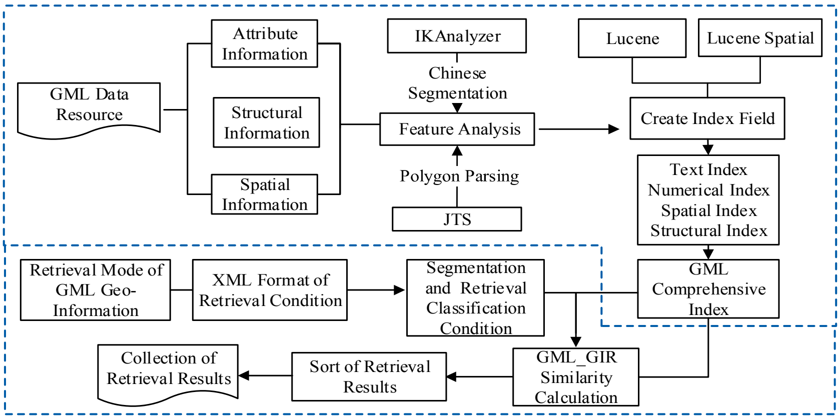


Figure 4. GML geo-information retrieval.

Complicated polygons (for example, polygons with holes or islands) rely on JTS to realize different objectives, such as the parsing, constructing geometry objects, computing complex spaces, and judging spatial relations. Then, the spatial index is created. For the structure index, it uses a textual index to realize construction of GML integrated indexes. The sorting of retrieval results is the key to obtain correct information. The retrieval conditions are expressed by XML formation; the retrieval feature items are extracted. Accordingly, the Lucene similarity scoring algorithm is improved. Finally, the final ranking of retrieval results is achieved by combining the similarity calculation method of the GML_GIR model.

4.2. Index Construction of GML Document Data

4.2.1. Index Model for GML

The index is the foundation for retrieval. Relying on the textual index model and spatial index components offered by Lucene, we designed an index model for GML data, as shown in Figure 5. This shows the contents of the model, such as attributes information, spatial information, and structure information. Using the Lucene index, we constructed indexes for GML element items and integrated them into the comprehensive index file. For the index construction for attributes text, such as the name, address, and other attribute fields, usually we need Chinese word segmentation and obtain textual items that are based on word units. In this paper, the textual index uses an inverted index. For numeric attributes, such as length and area, no word segmentation is required and so the indexes are directly constructed, then stored. For spatial information, first, coordinates are extracted; second, based on the forms of points, lines and areas, they are converted to the corresponding geometry; finally, with Lucene Spatial modules, spatial information is stored in index fields and spatial indexes are constructed by means of the structures and methods of the spatial module. For structural information, they are divided into label sequences for paths and element structures. In fact, they belong to the "text" type; therefore, the index construction is the same as for text.

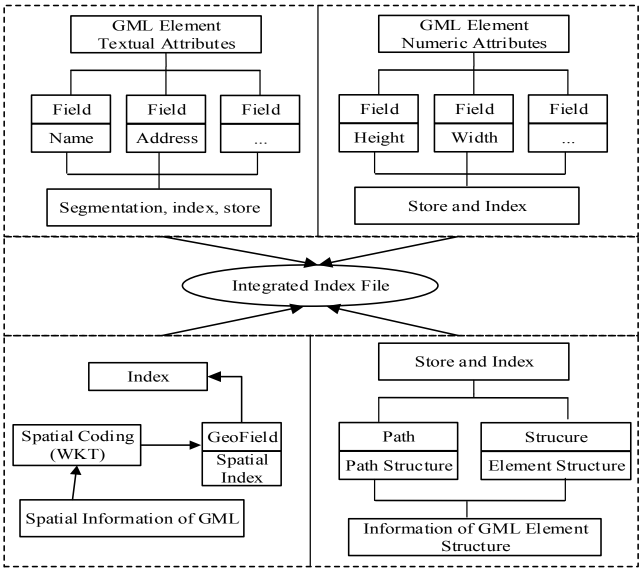


Figure 5. Index model for GML data.

4.2.2. Implementation for GML Index

With the components and API of Lucene, Lucene Spatial, and related technologies, GML indexes are constructed. Based on GML element features, the construction process can be described in three steps.

1. Constructing an Index of Attribute Information:

The index of attribute information includes textual and numeric indexes. To construct the textual index, we extract the attribute text data of geographical elements in GML documents, such as names and address. Each name is used as an index field, and the attribute value is defined as a value of the index field, which is shown in the form <Attribute Name, Attribute Value> as parameters to help construct the indexes. Then, the object of the textual index field is constructed and added to the GML document. The textual index needs the support of the Chinese word segmentation component. IKAnalyzer is the Chinese word segmentation component of Lucene; it provides a rich Chinese word segmentation interface, so this paper selected it to help construct the index.

The construction of numerical value is similar to that in the textual index; numerical attributes of elements should be exacted first in the form <Attribute Name, Attribute Value, Attribute Type> by taking the names of numerical attributes as index fields and selecting numerical index objects according to the numerical value type, such as IntField, LongField, and FloatField. Then, with NumbericUtils, the object of the numerical index field is constructed based on the triple form and added to the GML document.

2. Constructing the Spatial Index:

Spatial geometry information is extracted by parsing GML application schema, and then shown in the form <Geometry Type, Coordinate Values>. For basic geometric figures, such as points, lines, and rectangles of GML elements, they can be parsed directly by the Lucene Spatial module. For complicated polygons, the JtsShapeReadWrite interface is used by the supported spatial module, which is called WKTReader of JTS, to finish the parsing. After parsing geometric figures, index fields are created. Spatial index codes are obtained to complete the fields. This paper uses Geohash coding to encode the spatial index with the class GeohashPrefixTree; then, the spatial index object is constructed and added to the GML document.

3. Constructing the Structure Index:

The path structure and feature structure of GML data are essentially the same as for the text information. In this paper, we construct an index in the form of a text index and extract the structure information of GML elements. The path is stored in the form of a string and "/" is used to separate the path tag sequence. The element structure is stored in the form <Attribute Name, Attribute Type>. Taking the path structure string as a parameter, a "StringField" object of index fields is created. Then, also taking the element structure information as a parameter, structure index fields are created and added to the document objects. If required, the structure information is retrieved through the regular expression and wildcard search.

Based on the design and implementation of the GML index model, the process of constructing the index is given in the following (see Figure 6).

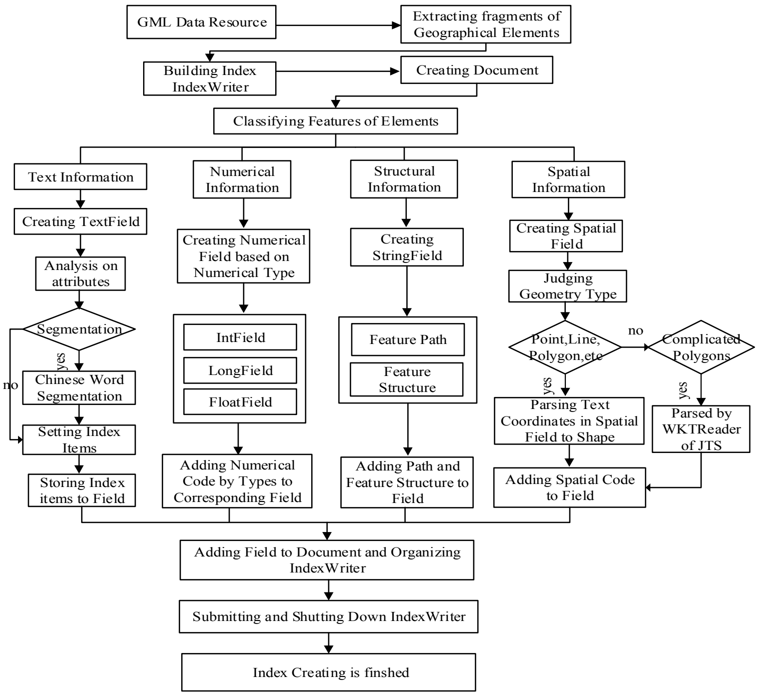

GML Data Resource
Extracting fragments of Geographical Elements
Building Index IndexWriter
Creating Document
Classifying Features of Elements
Text Information
Numerical Information
Structural Information
Spatial Information
Creating TextField
Analysis on attributes
Segmentation
yes
no
Chinese Word Segmentation
Setting Index Items
Storing Index items to Field
Creating Numerical Field based on Numerical Type
IntField
LongField
FloatField
Adding Numerical Code by Types to Corresponding Field
Creating StringField
Feature Path
Feature Structure
Adding Path and Feature Structure to Field
Creating Spatial Field
Judging Geometry Type
Point,Line, Polygon,etc
no
Complicated Polygons
yes
yes
Parsing Text Coordinates in Spatial Field to Shape
Parsed by WKTReader of JTS
Adding Spatial Code to Field
Adding Field to Document and Organizing IndexWriter
Submitting and Shutting Down IndexWriter
Index Creating is finshed

Figure 6. Process of constructing of GML index.

Taking the GML geographic element as a basic unit and using the Lucene framework, an index of DML data is constructed, where GML information is split, classified, reunited, and finally encapsulated as an index database in the documents, and the documents are submitted to Index Writer.

4.3. Correlation Ranking Model of Retrieval Results

The core module of Lucene provides a highly scalable scoring mechanism. In practical applications, users can rewrite or adjust the scoring mechanism based on practical requirements. The similarity class defines the abstract base class of the Lucene ranking mechanism. Thus, the ranking realization of any information retrieval model based on Lucene must be inherited from a similarity class and some related methods of computation should be extended.

This paper defines the GMLSimilarity class to implement the correlation ranking of the retrieval results for GML data. The GMLSimilarity class is inherited from the ClassicSimilarity Class, which integrates the method of the text vector space model (VSM) based on term frequency–inverse document frequency (TF-IDF). At the same time, it has a similarity normalization function. On the basis of these functions, in this paper, we added the method of calculating the attribute correlation, spatial relationship, and structural relationship, which successfully reduces duplicate work and quickly implements the sorting algorithm of retrieval results based on GML_GIR. Finally, the GMLSimilarity ranking module is added in the retrieval system by the setSimilarity method of IndexSearcher. The GMLSimilarity class mainly includes GMLSimScorer (the attribute of GML correlation ranking) and GMLSimWeight (the attribute of the GML correlation weight). The structure of the GMLSimilarity class is shown in Figure 7.

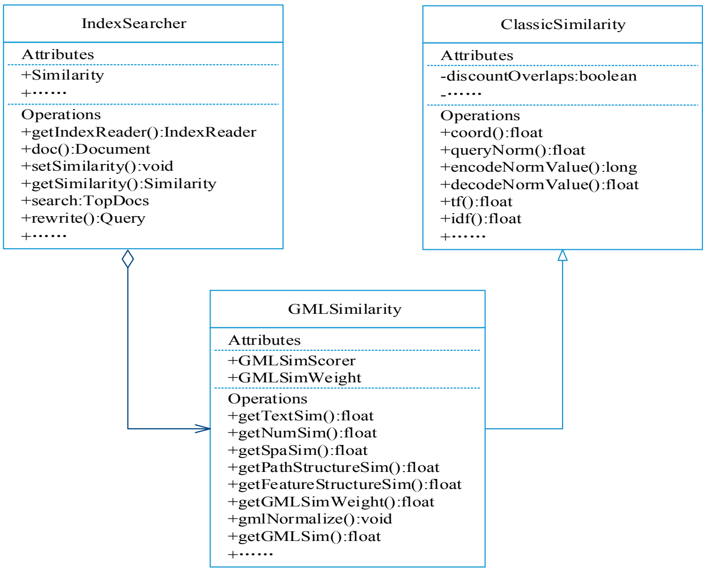


Figure 7. Structure of GMLSimilarity class.

As depicted in Figure 7, GMLSimilarity is inherited from ClassicSimilarity and regarded as one attribute of IndexSearcher in the correlation calculation of the search results. Based on the original algorithm and bonding with the GML_GIR retrieval model, the text calculation method is further encapsulated and extended, thereby adding the computing methods of attribute numbers, spatial relations, and structure relations. Meanwhile, some functions—for example, normalizing correlation—are overridden in the GMLSimilarity class. The new methods are shown in Table 3.

Table 3. Description of the main methods of GMLSimilarity.

Methods	Description
getTextSim()	Obtaining correlation of text, based on TF-IDF
getNumSim()	Obtaining correlation of numbers
getSpaSim()	Obtaining correlation of spatial relationships
getPathStructureSim()	Obtaining correlation of path structures
getFeatureStructureSim()	Obtaining correlation of element structures
getGMLSimWeight()	Overriding weight() to set weights of similarity
gmlNormalize()	Overriding normalize() to normalize correlation of geographical entities
getGMLSim()	Calculating scores of correlation of geographical entities in final retrieval results

5. Experiments and Results

5.1. Experimental Environment and Data Source

The experiment used the 1st Geographical Conditions Survey (GCS) data in Deqing County of Zhejiang Province. The study area is 937.96 square kilometers, which is from 119.76° E to 120.34° E and 30.43° N to 30.70° N. These data were applied to the system in the form of OGC WFS. There were

37 element layers in the first survey data, covering roads, buildings, waters, geographical units, etc. Fourteen layers with abundant data were chosen for the experiments, as shown in Table 4. Approximately 586 MB of GML data were obtained through the GetFeature interface. The development environment was Eclipse 4.4 with version 1.8.0 JDK. Rest Service was used as the retrieval service to implement the retrieval service invocation. It relied on JTS and Lucene Spatial components to implement the retrieval of spatial information.

Table 4. Information of WFS layers of experimental data.

Layer Label	Layer Name	Geometry Type	Number of Elements	Size of GML Document (KB)
SFCP	Construction (Point)	Point	2340	1070.5
SFCL	Construction (Line)	Line	568	548
SFCA	Construction (Polygon)	Polygon	12	1.37
LVLL	Country Road	Line	959	1597.44
LCTL	Urban Road	Line	364	254
LRDL	Highway	Line	1174	1556.48
LRRL	Railway	Line	7	26.1
HYDL	Water (Line)	Line	3471	4700.16
HYDA	Water (Polygon)	Polygon	15,902	62531
BUCP	Comprehensive Unit (Point)	Point	323	156
BOUP7	Administrative Village	Point	181	76.8
BUCA	Comprehensive Unit (Polygon)	Polygon	20	2.46
BOUA6	County District	Polygon	11	1146.88
LCA	Land Cover Classification Data	Polygon	112,334	525,312

5.2. Experimental Procedure

The WFS data service published from the national geographic condition survey data of one county was used as the experimental data. In the client, the GetFeature interface of the WFS data service was called through the HTTP protocol to obtain the GML data. These returned data functioned as objects of index constructing and retrieving. Based on the GML_GIR retrieval model and open-source retrieval framework of Lucene, we constructed an index for GML data, implemented the retrieval service, sorted the retrieval results, and performed the front-end display (see Figure 8).

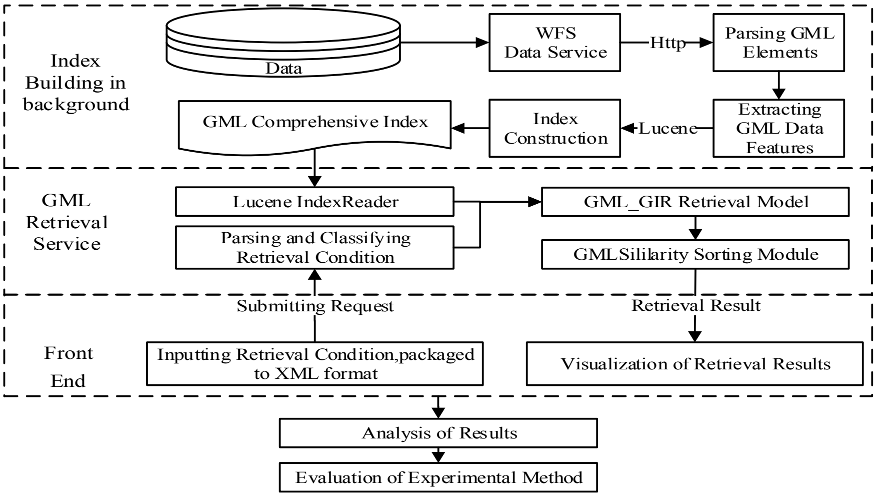

Index Building in background
Data
WFS Data Service
Http
Parsing GML Elements
GML Comprehensive Index
Index Construction
Lucene
Extracting GML Data Features
GML Retrieval Service
Lucene IndexReader
GML_GIR Retrieval Model
Parsing and Classifying Retrieval Condition
GMLSililarity Sorting Module
Submitting Request
Retrieval Result
Front End
Inputting Retrieval Condition,packaged to XML format
Visualization of Retrieval Results
Analysis of Results
Evaluation of Experimental Method

Figure 8. Experimental procedure.

To verify the feasibility and accuracy of the GML retrieval method, an analysis was performed on different aspects, such as efficiency of the index construction, the size of the index file, the query efficiency, and the query accuracy. Six retrieval examples were designed under four different circumstances:

1. Testing attribute retrieval (R1); only text or numbers were included.

2. A testing retrieval combination of simple spatial relationships and attributes (R2-R4).
3. A testing retrieval combination of kinds of spatial relationship factors and attributes (R5).
4. Testing retrieval structural features (R6).

Table 5 shows examples of retrieval. The retrieval for each document of different sizes was executed 10 times in the experiment, which obtained the average running time. The experiment verified the retrieval quality and efficiency.

Table 5. Examples of retrieval.

Numbers	Examples	Retrieval Type
R1	Scenic area above 4-A level	Text and numeric attributes
R2	Hospitals in Wukang	Topologic relation and attributes
R3	Restaurants within 500 km from Deqing People's Hospital	Metric relation and attribute
R4	Gas Station in southeast of Beijing-Fuzhou Highway	Position relation and attributes
R5	Gas Station 500 m in southeast of Beijing-Fuzhou Highway	Metric relation, position relation and attributes
R6	(All required type of elements in layer SFCP)	Structural relation

5.3. Efficiency of Index Construction

The index construction is the basis and premise of geographic information retrieval. The efficiency of the index construction and size of the index file should address the following items, as outlined below.

1. In this paper, we analyze the runtime and size of the index file with different sizes for GML data when constructing an index.

The land cover data were chosen as the experimental data in this group, which contained 112,334 elements, and the size of the GML documents exceeded 500 MB. These data were divided into many groups, of which the minimum was about 100 MB and the maximum was about 500 MB.

As shown in Figure 9, with the amount of GML data increasing, the index construction time and the size of the index file also increase. The time of the index construction is linearly related to the GML document size. The average 100 MB GML document required 688.93 s. Due to the time consumed in the network transmission for calling the WFS service to obtain the GML data, if the GML documents were local, much less time would be required to construct indexes for GML documents with the same amount of data. From the retrieval space, an increase in GML data has less effect on the size of the index files, which reflects the functions of compression and index optimization for the comprehensive index of Lucene with the large amount of data.

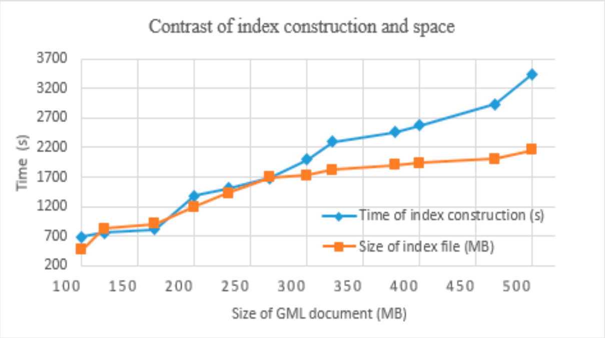
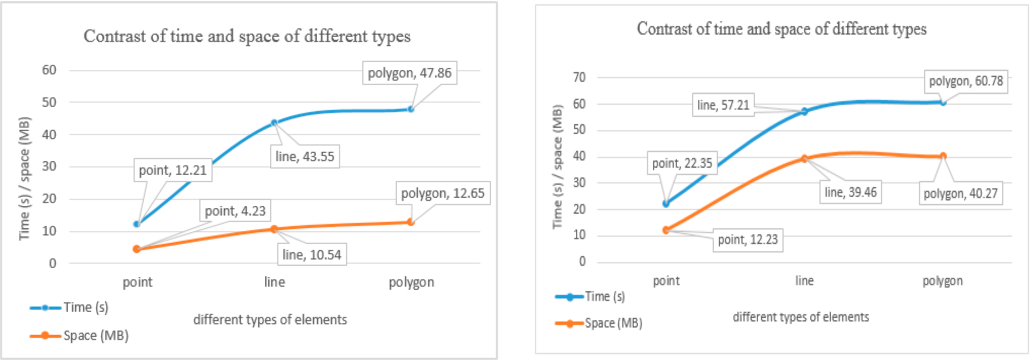


Figure 9. Time of index construction vs. space.

2. The GML data of point elements, line elements, and area elements were selected as research objects. Using the same GML document size, we constructed an index for these objects, and documented the runtime and the size of index files.

In Figure 10, under the same size of GML documents (Figure 10a or Figure 10b) and different types of elements, the construction time of point elements is the shortest, and its index file size is the smallest, while the construction time of area elements is the longest, and the index file size is the greatest. The experimental results are consistent with expectations. Compared to point elements, line elements and area elements are supposed to store more coordinate points. Thus, GML data increased, and there was much more time and space to store and parse the GML data. In addition, with more compressive spatial relations for line and area elements, the Lucene Spatial module required much more time and memory to construct the spatial index for the spatial information of geographical elements. Thus, the computing time was longer and the index file size was larger. However, the trend of space change of Figure 1b is slightly different from that of Figure 1a because most of the polygons are regular polygons, and thus the size of index files is very close to that of line elements (the same as time), and is still consistent with the above rules.



(**a**) (**b**)

Figure 10. (**a**) Time vs. space for GML elements of different type (Example 1); (**b**) time vs. space for GML elements of different type (Example 2).

5.4. Retrieval Effectiveness

The recall and precision ratio are effective indicators for analysis of retrieval quality. The recall ratio represents the ratio of the number of relevant results to all that meet the retrieval conditions. The precision ratio represents the ratio of the number of relevant results to all that are retrieved. These formulas are defined as in Equations (16) and (17):

$$Recall = \frac{C}{M} \tag{16}$$

$$Precision = \frac{C}{T}, \tag{17}$$

where C represents the correlation result numbers of retrieval, M represents all result numbers of meeting the retrieval conditions, and T represents the numbers of all retrieved results.

In order to verify the retrieval quality, this paper designs the retrieval experiment based on Oracle as a contrast experiment. Oracle has launched Oracle Spatial, a spatial query component, and Oracle Text, a full-text retrieval component, where the full index uses the Chinese_lexer analyzer. These components can construct spatial indexes and full-text indexes. Table 6 shows a comparison of the two methods.

According to the statistics of the retrieval results, as shown in Table 6, the proposed method proved the validity of the experiments. Compared to the method based on Oracle, in general, the R value of the proposed method is higher than that of the method based on Oracle; however, for P it is the opposite. There are two main reasons. First, although IkAnalyzer could provide a richer Chinese word segmentation interface, more useless words are divided, which affects the precision, while the

Chinese_lexer analyzer can identify the vast majority of Chinese words and improve the efficiency. Thus, for spatial retrieval, this paper uses compressive indexes and retrieval result ranking to obtain more relevant results; the method based on Oracle is usually performed by attribute retrieval and then space retrieval, in which some relevant results are filtered out. The results almost meet the retrieval requirements.

Table 6. Comparison of retrieval quality.

Retrieval Instances	**The Proposed Method**		**The Method Based on Oracle**	
	R	**P**	**R**	**P**
R1	95%	90%	90%	91%
R2	92%	90%	93%	90%
R3	97%	92%	83%	93%
R4	97%	95%	92%	90%
R5	95%	90%	89%	83%
R6	100%	92%	100%	93%

As shown in Figure 11, R and P represent the completeness and quality of the retrieval results, respectively. When the retrieval conditions are R2 and R5, respectively, changes in the curves are all at lower values, as shown in Table 5; the main retrieval type of R2 and R5 is topology relations and two direction and metric relations, respectively (viz. comprehensive relations). This is because topology calculations are often sensitive to spatial computing, and comprehensive relations retrieval is affected by multiple factors such as comprehensive index complexity and computational complexity; therefore, this is in line with the actual situation of the retrieval. In addition, for the blue line, there is a bold curve with a very flat trend that corresponds to R3 and R4, as shown in Table 5; all their retrieval types belong to spatial relations and the computational complexity is more similar, so R reflects the trend. In summary, comparison results reflect the effectiveness of the algorithm from different aspects.

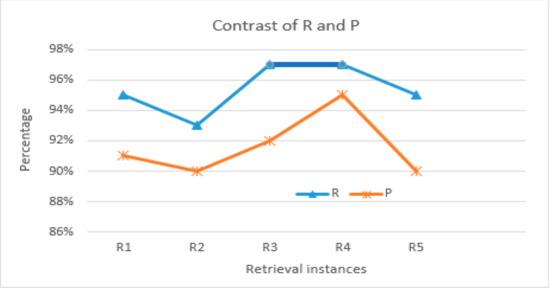


Figure 11. Comparison of R and P.

5.5. Retrieval Efficiency

The response time is an efficiency indicator defined for the GML retrieval service from accepting the request to returning the retrieval results. We obtained statistics on the response time based on different circumstances (R1, R2, R5, R6) with the same size of GML data, and the size of the GML files varied from 100 MB to 500 MB. The experimental instances are as follows: 1. attribute text retrieval: "name: station"; 2. attribute numeric retrieval: "width: 20–40 m"; 3. spatial relations retrieval: "highway within WuKang county"; 4. structural information retrieval: "/FeatureCollection/featureMember/SFCP"; 5. comprehensive retrieval: "stations within 1 km east of JingFu highway". Each type of retrieval was run 10 times and its average was obtained. Experimental results are shown in Figure 12.

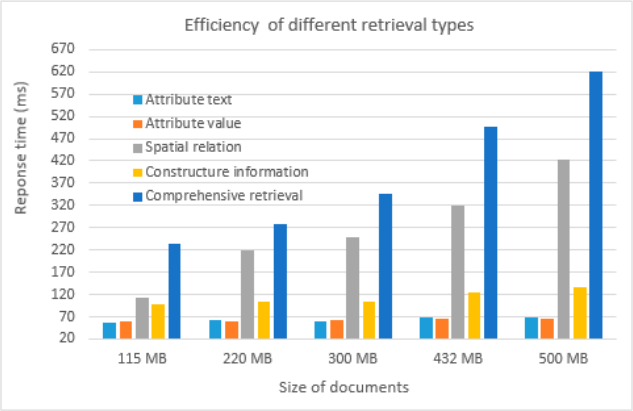
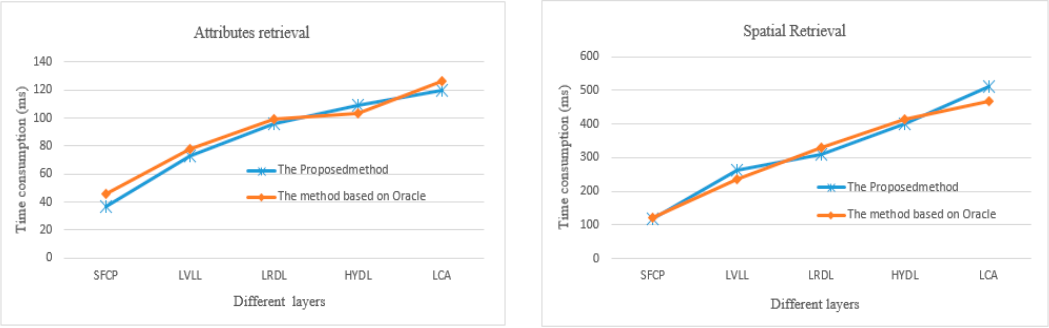


Figure 12. Efficiency of different retrieval types.

As presented in Figure 12, the efficiency of retrieving attribute text and numbers is higher than that of spatial and structural relationships with different sizes of data, which proves that the Lucene text search engine is suitable for GML data retrieval of attribute text and number. However, for spatial and structural retrieval for GML data, efficiency could be improved in terms of the index construction. The retrieval efficiency of simple features is higher than that of comprehensive retrieval, which aligns with the actual situation. With the increasing GML data and index data (especially the spatial index), it can be observed that the time for retrieving a single feature increases. Thus, the retrieval time shows a large growth trend. Nevertheless, the response time of comprehensive retrieval is less than 1 s, even for 500 MB of GML data, which can definitely meet user demand and thus shows the feasibility of comprehensive retrieval.

In order to verify the results of time consumption with the method based on Oracle, we selected four typical layers with rich elements that represent points, lines, and polygons (see Table 5). For each layer, attributes and spatial retrieval were selected for comparison. The average of the 10 run times is selected. As shown in Figure 13a, overall, the proposed method is better than the method based on Oracle. This indicates that the attribute relevance (see Section 3.1) is suitable for GML data. As shown in Figure 13b, for the two methods, in general, the trend line of run time is similar; however, for the LCA layer, the change is more obvious. For polygon entities, the accuracy is too low for the distance calculation with Lucene. The reason may be how long it takes to select the GeoHash string; it might be better to choose an R-tree index, just like the method based on Oracle.

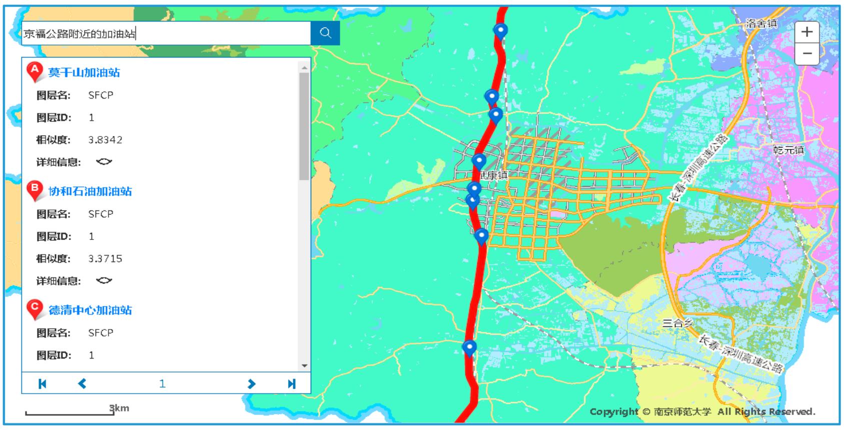


(**a**) (**b**)

Figure 13. (**a**) Time comparison of attributes retrieval; (**b**) time comparison of spatial retrieval.

5.6. Retrieval Running Examples

In this paper, several typical retrieval examples are selected to analyze the rich forms of geographic information retrieval methods for GML data resources. As shown in Figure 14, the red line is the Jing-Fu highway and is taken as the reference object. We entered "gas stations near the Jing-Fu highway" in the retrieval box. The default retrieval is gas stations within 1 km around the Jing-Fu highway. The retrieval results are basically distributed along the Jing-Fu route. With the retrieval of spatial relations, search results can be effectively filtered, and a large number of results which are not satisfied with spatial relations are excluded. Compared with a traditional search, the search results are more targeted, which provides a richer form of retrieval than simple, keyword-based full-text retrieval.


京福公路附近的加油站

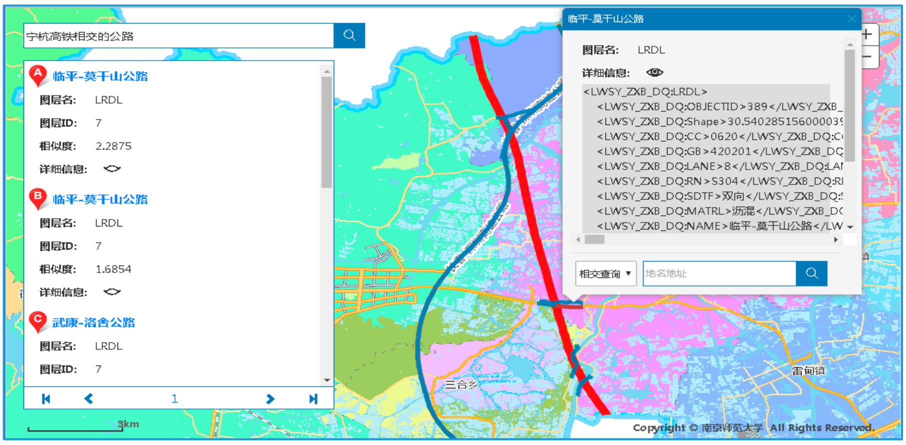
A 莫干山加油站
图层名: SFCP
图层ID: 1
相似度: 3.8342
详细信息:

B 协和石油加油站
图层名: SFCP
图层ID: 1
相似度: 3.3715
详细信息:

C 德清中心加油站
图层名: SFCP
图层ID: 1

Figure 14. "Gas stations near the Jing-Fu highway".

As shown in Figure 15, the red line is the Ning-Hang highway. Taking the Ning-Hang highway as a reference object, we retrieved "highways intersecting with the Ning-Hang highway". The blue lines are highways, which are retrieval results. For example, we select one retrieved highway, which is returned in the form of GML elements. This clearly expresses the constraints of geospatial relations between the retrieved object and the reference object, and reflects the intersection characteristics between the line elements. It extends the retrieval mode and application scope of the GML data.


宁杭高铁相交的公路

A 临平-莫干山公路
图层名: LRDL
图层ID: 7
相似度: 2.2875
详细信息:

B 临平-莫干山公路
图层名: LRDL
图层ID: 7
相似度: 1.6854
详细信息:

C 武康-洛舍公路
图层名: LRDL
图层ID: 7

临平-莫干山公路
图层名: LRDL
详细信息:

```
<LWSY_ZXB_DQ:LRDL>
   <LWSY_ZXB_DQ:OBJECTID>389</LWSY_ZXB_
   <LWSY_ZXB_DQ:Shape>30.540285156000039
   <LWSY_ZXB_DQ:CC>0620</LWSY_ZXB_DQ:C
   <LWSY_ZXB_DQ:GB>420201</LWSY_ZXB_DQ
   <LWSY_ZXB_DQ:LANE>8</LWSY_ZXB_DQ:LAI
   <LWSY_ZXB_DQ:RN>S304</LWSY_ZXB_DQ:RI
   <LWSY_ZXB_DQ:SDTF>双向</LWSY_ZXB_DQ:S
   <LWSY_ZXB_DQ:MATRL>沥混</LWSY_ZXB_DC
   <LWSY_ZXB_DQ:NAME>临平-莫干山公路</LW
```

相交查询 ▾
地名地址

Figure 15. "Highways intersecting with the Ning-Hang highway".

6. Conclusions

In this study, by analyzing and extracting the attribute features, spatial features, and structure features of GML data, we took GML elements as retrieval units and designed retrieval modes to normalize the expression of the retrieval conditions. Then, in order to take into account all the features of GML data, we put forward a GML_GIR model for the first time. On the basis of the open-source, full-text retrieval framework Lucene, we constructed compressive indexes, designed relevance calculations for different features, modified related computing modules of Lucene retrieval results, and realized retrievals of GML information. An experimental solution is proposed with 1st GCS data. The experiment also proves its efficiency and effectiveness in different retrieval conditions. The recall and precision rate of searching results was more than 90%, which meets users' requirements. Compared with traditional GML query methods, the experiment could provide a friendly interface. The feasibility of the design is proved by the experimental data.

In summary, the proposed method can solve most GML data retrieval problems, but does not cover all the GML data. It should be noted that this study has examined the data about WFS data service, but needs a more relevant GML data format for further modifications. In addition, this paper is limited by the amount of experimental data; the results do not reflect the efficiency and quality issues in mass data, so how to create efficient indexes and retrieval methods is a key issue to address in our future research.

Acknowledgments: This study was supported by grants from the National Natural Science Foundation of China (Grant Nos. 41171301 and 41771424).

Author Contributions: Shuliang Zhang conceived the idea for the study and provided financial support. Caili Fang was responsible for the design of the study, setting up experiments, and completing most of the experiments, and wrote the initial draft of the manuscript.

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

References

1. Zhu, Q.; Li, Y.; Xiong, Q.; Zlatanova, S.; Ding, Y.; Zhang, Y.; Zhou, Y. Indoor multi-dimensional location gml and its application for ubiquitous indoor location services. *ISPRS Int. J. Geo-Inf.* **2016**, *5*, 220. [CrossRef]
2. Brunk, B.K.; Porosnicu, E. Aeronautical Information Exchange Model (AIXM) GIS interoperability through GML. In Proceedings of the Twenty-Fifth Annual ESRI, International User Conference, San Diego, CA, USA, 25–29 July 2008.
3. Gröger, G.; Plümer, L. Citygml-interoperable semantic 3d city models. *ISPRS J. Photogramm. Remote Sens.* **2012**, *71*, 12–33. [CrossRef]
4. Papadimitriou, F. The Algorithmic Complexity of Landscapes. *Landsc. Res.* **2012**, *37*, 591–611. [CrossRef]
5. Selvaganesan, S.; Haw, S.C.; Soon, L.K. Effective XML keyword search using dual indexing technique. *Inf. Technol. J.* **2014**, *13*, 643–651. [CrossRef]
6. Ren, J.H.; Zhou, J.; Meng, X.F.; Wei, K. Results ranking approach of XML keyword search based on keyword's structural relationships. *Comput. Sci.* **2013**, *6*, 178–182.
7. Li, J.; Xiong, H.L. XML keywords retrieval by integrating semantics of document and user inquiries. *J. Comput. Appl.* **2010**, *11*, 2945–2948. [CrossRef]
8. Li, X.; Li, Z.H.; Zhang, L.J.; Chen, Q.; Li, N. MXDR: Distributed information retrieval for multi-XML document based on keywords. *Comput. Sci.* **2011**, *10*, 152–156.
9. Zhou, X.P.; Shi, Y.M.; Zhang, J. Parallel top-k keyword search algorithm in probabilistic XML documents. *Comput. Sci.* **2013**, *3*, 232–237.
10. Chan, C.Y.; Felber, P.; Garofalakis, M. Efficient filtering of XML documents with XPath expressions. *VLDB J. Int. J. Very Large Data Bases* **2002**, *11*, 354–379. [CrossRef]
11. Lian, X.; Lin, W.J.; Zhang, F.W. Similarity evaluation between XML documents based on bidirectional path constraint model. *J. Comput. Res. Dev.* **2010**, *47*, 60–65.
12. Wang, L.; Cheung, D.W.; Mamoulis, N. An efficient and scalable algorithm for clustering XML documents by structure. *IEEE Trans. Knowl. Data Eng.* **2004**, *16*, 82–96. [CrossRef]
13. Liu, X.P.; Wan, C.X.; Liu, D.X. Effective XML vague content and structure retrieval and scoring. *J. Comput. Res. Dev.* **2010**, *6*, 1070–1078.
14. Corcoles, J.E.; Gonzalez, P. A specification of a spatial query language over GML. In Proceedings of the 9th ACM International Symposium on Advances in Geographic Information Systems, Allanta, GA, USA, 9–10 November 2001; pp. 112–117.
15. Boucelma, O.; Colonna, FM. Gquery: A query language for GML. In Proceedings of the 24th Urban Data Management Symposium, Chioggia Venice, Italy, 27–29 October 2004; pp. 27–39.

16. Almendros-Jiménez, J.M.; Becerra-Terón, A.; García-García, F. Xpath for querying GML-based representation of urban maps. In *Computational Science and Its Applications—ICCSA 2011, Proceedings of the 2011 International Conference on Computational Science and Its Applications, Santander, Spain, 20–23 June 2011*; Lecture Notes in Computers Science; Springer: Berlin, Germany, 2011; Volume 6782, pp. 177–191.
17. Savary, L.; Gardarin, G.; Zeitouni, K. GeoCache: A cache for GML geographical data. *Int. J. Data Warehous. Min.* **2007**, *3*, 67–88. [CrossRef]
18. Lan, X.J.; Lv, G.N.; Liu, D. Xquery based GML query language. *Sci. Surv. Mapp.* **2005**, *6*, 100–103.
19. Guan, J.H.; Zhu, F.B.; Zhou, J.G.; Niu, L.P. GQL: Extending XQuery to query GML documents. *Geo-Spat. Inf. Sci.* **2006**, *2*, 118–126.
20. Tong, X.H.; Xu, G.S.; Gong, J.Y. Modeling spatial features based on geography markup language in GIS. *Geomat. Inf. Sci. Wuhan Univ.* **2005**, *30*, 209–213.
21. Jones, C.B.; Purves, R.S. Geographic Information Retrieval. *Int. J. Geogr. Inf. Sci.* **2008**, *22*, 219–228. [CrossRef]
22. Cai, G. GeoVSM: An integrated retrieval model for geographic information. In Proceedings of the International Conference on Geographic Information Science, Boulder, CO, USA, 25–28 September 2002; Springer: London, UK, 2002; pp. 65–79.
23. Buscaldi, D.; Rosso, P. Explicit query diversification for geographical information retrieval. In Proceedings of the 33rd European Conference on Information Retrieval, ECIR 2011, Dublin, Ireland, 18–21 April 2011; pp. 73–80.
24. Papadimitriou, F. Artificial intelligence in modelling the complexity of mediterranean landscape transformations. *Comput. Electron. Agric.* **2012**, *81*, 87–96. [CrossRef]
25. Ren, K.J.; Zhang, S.W.; Lin, H.F. A document's placenames-aware document ranking for GIR. *Acta Sci. Nat. Univ. Pekin.* **2013**, *49*, 219–226.
26. Salton, G.; Wong, A.; Yang, C.S. A vector space model for automatic indexing. *Commun. ACM* **1974**, *18*, 613–620. [CrossRef]
27. Cardoso, N.; Cruz, D.; Chaves, M.; Silva, M.J. Using geographic signatures as query and document scopes in geographic IR. In *Advances in Multilingual and Multimodal Information Retrieval*; Springer: Berlin, Germany, 2008; Volume 5152, pp. 802–810.
28. Bruns, H.T.; Egenhofer, M.J. Similarity of spatial scenes. In Proceedings of the Seventh International Symposium on Spatial Data Handling, Delft, The Netherlands, 12–16 August 1996; pp. 173–184.
29. Goyal, R.; Egenhofer, M.J. Consistent queries over cardinal directions across different levels of detail. In Proceeding of the 11th International Workshop on Database and Expert Systems Applications, London, UK, 4–8 September 2000; pp. 876–880.
30. Liao, H.W.; Yang, Y.; Jia, Z. An improved web structure similarity based on matching algorithm of tree paths. *J. Jilin Univ.* **2012**, *50*, 1199–1203.