Geographic Metadata Searching with Semantic and Spatial Filtering Methods

Tristan W. Reed*, Elizabeth-Kate Gulland*, Geoff West*, David A. McMeekin* and Simon Moncrieff*
* Cooperative Research Centre for Spatial Information
Department of Spatial Sciences, Curtin University, Bentley, Western Australia
Email: {tristan.reed, e.gulland, g.west, d.mcmeekin, s.moncrieff}@curtin.edu.au

Abstract—**Web search engines, such as from Google, are very good at finding relevant information in documents and web pages. However, when such tools are used to find spatial web services, the user has to be very specific in describing what they are looking for to find relevant results in high-ranked positions. To locate an Open Geospatial Consortium-compatible web service relating to soil in Australia, a query such as "getcapabilities australia soil" is required to find relevant results, as there are no spatial constraints available. Current spatial data discovery systems, such as spatial catalogue systems, generally keyword match user queries to the content of metadata catalogues. Such systems also provide basic spatial constraints, which limit the user's ability to find results. A combination of semantic and spatial search techniques are required to effectively search geospatial data, as existing systems are primarily designed to search human-readable documents. A search algorithm is presented which uses such techniques to expand text queries to find more relevant spatial datasets through spatial filtering, natural language query decomposition and the use of thesaurus graphs to expand queries. A Resource Description Framework (RDF) schema that extends the ISO 19115 specification is explored as part of the query expansion technique, including the evaluation of tools to generate these graphs from unstructured documents which allows the overcoming of restrictions to access data behind an organisation's firewall. A prototype has been written as a web application, using the Django framework and the Python programming language and the Natural Language Toolkit (NLTK) interface to WordNet. Initial tests on seperate components of the system as well as the above system has shown the feasibility of the search system as a whole.**

***Keywords*–Semantic search; Resource Description Framework; Spatial search; Metadata; Thesaurus; Graph; Ontology.**

I. INTRODUCTION

Index-based web search engines, such as Google [1], are successful at generating automated methods to build indexes of information available on publicly accessible documents and HTML pages on the Web [2]. Such search engines cannot index information that is hidden from the Web, such as that held in file systems and databases behind firewalls.

Using search tools to find spatial data that is publicly available on the Web from an Open Geospatial Consortium (OGC)-compatible web service is only possible if the user specifies their query in a very precise manner. Such an example would be "getcapabilities australia soil". This query returns links to relevant OGC-compatible web services about soil in Australia, but the user has to know that "getcapabilities" is a word used in the schema of said web services. Without the use of "getcapabilities" in the query, relevant results are ranked lowly as the content of the machine-to-machine XML-based structure is different to the content of the human-readable HTML structure the search engine is looking for.

Specific spatial information tools exist for searching catalogues of metadata, such as GeoNetwork. The commercial Google Map Engine (GME) also searches metadata catalogues, but does not take advantage of technology used in Google's own web search engine. The CKAN cataloguing system can be used for spatial data, also searching a metadata catalogue. All of these search tools are restricted by search methods that keyword-match the user's query with the content of metadata records [3].

Limitations of keyword-matching approaches include incomplete source data, such as the content of metadata records, as well as syntactic differences between user specified queries and metadata record content with similar meanings.

Typically, metadata generation is a manual process and leads to minimal metadata being supplied by spatial data custodians for many data sets. Manually generated metadata is rarely a complete description complying with the ISO 19115 metadata standard. Due to the difficulties in automatically generating metadata to fill records, such as determining the context and relevance of data [4], it is proposed to approach improvements from the other side, automatically expanding user queries instead. It is easier to find relevant metadata records by creating contextually relevant queries than attempting to create contextually-relevant metadata, due to the size of the query compared to the size of the metadata record.

To achieve this, natural language processing is applied to queries to separate the spatial and non-spatial components of a query, allowing the application of spatial operations on data sets. Graph-based query expansion is used to parse the non-spatial part of the query, which allows the discovery of more data sets that have metadata syntactically different to the user's query, but similar in meaning. The expanded queries are run over traditional metadata records, while integrating into the expansion a graph-based domain thesaurus extracted from non-structured resources, such as reports found behind firewalls within internal repositories. Queries such as "Parks in Perth" identify an object ('Park'), a spatial operator ('in') and a location ('Perth') and can look for data sets of interest inside bounding boxes or polygons describing Perth, depending on the services used.

Much development has gone into the standardisation of metadata, including ISO 19115, used by many spatial data providers. The ISO 19115 specification is explored to help determine the best methodology to automatically generate metadata discovered through searching file systems and databases, possibly behind a firewall. This allows metadata to be acquired from sources such as PDF reports hidden in a data provider's repository. This technique is used to populate a thesaurus of similar domain-specific terms also acquired from the repository. The source of the data is noted and related to other records found in the same document, as well as in other documents containing the same terms. This information is then used to expand the location and object part of the query respectively.

Such metadata must be generated by each data provider to overcome restrictions on access behind firewalls. To this end, a number of commercially available software tools have been explored to determine their capabilities including how they can generate publishable metadata for consumption by the system. On the web, RDF models can be used to store metadata. An RDF schema of ISO 19115 [5] has been explored for its suitability, as well as research being conducted for a 'domain thesaurus'.

The paper is organised as follows: Section II explores current systems used to search and manage geospatial metadata; Section III discusses the use of semantic and spatial filtering techniques to improve search results and Section IV presents the results of a prototype system implementing some of the proposed techniques.

II. Current Approaches and Systems

Three systems currently used to search and manage geospatial metadata are GME [6], CKAN [7] and GeoNetwork [8]. GME is a commercial product from Google, Inc. which extends the abilities of Google Maps to allow more complex spatial data to be overlaid upon Google's base maps. Data custodians upload data files alongside their associated metadata to Google's cloud. Each layer, or other asset, has associated metadata which can be searched through the Maps Engine API or the Google Maps Interface itself. The search is based on keyword matching, looking for occurrences of the user's exact phrase within the metadata. CKAN is an open-source cataloguing system that, whilst not designed solely for spatial data, is commonly used to catalogue and search spatial data by various jurisdictions, including many Australian government departments.

The open source GeoNetwork is a similar system, except that data is not stored within the system itself but rather is accessed through OGC-compatible web services such as the Web Feature Service (WFS), Web Map Service (WMS) and Catalogue Service for the Web (CSW) [9]. These services expose relevant metadata about geographic data sets, which GeoNetwork keyword-matches with the user's query. The function 'GetCapabilities' exposes much of the metadata accessed by GeoNetwork, which complies with some of ISO 19115 [10]. CKAN functions in the same manner as GeoNetwork for OGC-compatible services, but also allows spatial data to be uploaded in file-based formats as well. In that case, the metadata must be manually generated rather than harvested. These services also expose the data sets themselves for use in other systems.

The OGC WFS standard defines a number of possible spatial operators including 'Contains', 'Intersect' and 'Equals' which can be applied to any known spatial feature type such as polygons or points. However, as there is no requirement for a WFS dataset to implement all of these operators, the availability of these operations cannot be assumed in all cases [11]. Another complication is that the syntax used to describe these operations varies depending upon the version of WFS specified in the data request.

All three of these systems rely upon keyword matching of the user's query; if the user misspells a word or uses a synonym of a keyword within the metadata, valid results will not be included in the result set. Much like traditional web search, keyword indexing is the primary way this is achieved. Optimisation of queries and a lack of support for alternatives means that important spatial operators such as 'in', 'within' and 'near' are ignored.

These systems do not have free access behind a firewall; GeoNetwork allows only basic authentication rather than more sophisticated methods which would use permissions to expose extra data to certain groups of machines or people. Without access behind a firewall, it is possible that many data sets and their metadata cannot be interrogated, despite the fact the user may have access to the data set.

All three systems provide the ability to restrict the search set spatially with a bounding box. However, there is no ability to restrict the search based on a polygon or text term. Being able to restrict a search by a polygon is important as polygons allow the user to use a complex many-edged shape that is more representative of real-world spatial boundaries than a bounding box.

The use of a visual bounding box drawn on screen by the user in CKAN and GeoNetwork's case is difficult and time consuming. This is particularly so when the map is small. A rectangular box is not always representative of an area of interest; consider a collection of irregular islands such as Hawaii. The bounding box as implemented in these tools also only allows the use of an implicit 'within' spatial operator; others such as 'next to' are unable to be used.

Another common theme with all of these tools is the manual generation of metadata - even in the case of automatically harvesting metadata from a 'GetCapabilities' call, the data must originally be manually generated by the data custodian. This leads to issues of quality and completeness, as metadata is typically a low priority for data custodians. Such metadata includes a title and description of the data set alongside metadata tags which briefly describe the dataset. Often these fields are subject to standardisation by data providers, leading to metadata which is sufficient for some groups but less useful for other users. It is rare that more complete ISO 19115 descriptions are provided, either formally or as part of a description field.

The features of contemporary web search tools exceed those of geospatial web search tools by allowing more complex queries from users. Many of these capabilities are examples of semantic search capabilities used in a general sense. Through using linked data, web search engines are able to deliver results that are similar to but not exactly the same as the user's query. By matching components of the user's query with synonyms and correctly spelt words (in the case of misspellings), more relevant results can be returned.

In Google's web search, this can be seen through the use of their proprietary Knowledge Engine graph, which injects some semantic capabilities into Google's web search, which expands the user's query to find data that is not expressed in the same way [12]. This can be achieved through the use of a thesaurus graph, which details relationships between words and even in some cases misspelled words. The Knowledge Graph is Google's proprietary version of this. However, it is to be noted that Google's search algorithm is designed to search human-readable HTML pages, rather than machine-to-machine XML documents such as OGC-compatible web services. The methods can however be applied to a system designed to read this format of data.

Systems such as that from [13] improve upon geospatial search to provide more modern semantic features, however they are manual in nature and designed specifically for the geospatial context in which they are used. Rather, they are good at generating a corpus of known OGC-compatible web services and how they are related, but are not specialised in providing an improved user-facing query method.

III. Approach

A. Proposed New Semantic and Spatial Filtering Techniques

These issues highlighted above are addressed through the use of semantic technologies as seen in contemporary web search tools, including ontology-based, graph-powered natural language processing and extending the use of ontologies into handling geographic phenomena to find semantic matches to a text query containing location information. Spatial filtering techniques are also employed to further filter results of user queries to return more relevant results.

The use of these technologies enable more targeted and relevant data sets to be returned to the user for a given search query, by finding metadata that is phrased differently to the user's query, alongside being able to use spatial filters to remove geographically irrelevant data sets.

To effect these techniques, the user enters a query in one of a number of forms. Currently these are restricted, but more complex formats will be considered as needed. Natural language processing is then used to classify the query as one of the following four formats understood by the algorithm, in precedence order as seen in Figure 1.

I { Object } { Operation } { Location }
(e.g. Parks in Perth)
II { Location } { Object }
(e.g. Melbourne forests)
III { Object } { Location }
(e.g. Boundaries Sydney)
IV { Object }
(e.g. Admin boundaries)

Figure 1. Natural language precedence algorithm

In these rules, a 'Location' is a geographic area in which to restrict the search, an 'Operation' is a spatial operation upon the said 'Location' and the 'Object' is the data being inquired for, related to the 'Location'. It is assumed that, for this system, the operation will be 'within' for rules III and IV.

Rule IV is a fallback for when a location cannot be determined using rules I to III - in essence, the query is treated as a standard text query that will not take advantage of the improved ability of spatial operations and geographic restriction. Logically, it is meaningless to express this in terms of an operation and location. An overall representation of the design of the system can be seen in Figure 2.

If the query is of type I, II or III, the system will attempt to find the most relevant area using a WFS call to a service providing boundary information. The design of this part of the system allows administrators to specify the WFS, layer and field types required.

The queries are decomposed using a simple 'split' method on a list of spatial operations for rule I - these operations are 'in', 'near', 'next to' and 'intersects', however WFS and the GeoDjango system used to complete the filtering can use other operations [14]. The location will be found to the right of the split, and the object to the left.

If the algorithm cannot find an operation in the query, it attempts to find the location and object based on rules II and III. For rules II and III, the system makes calls to the WFS for the beginning and end components of the query - attempting up to three words on each side. For example, a query such as ”West Perth bus stops” would try the following in precedence order as locations and the balance as objects:

- West *(rule II)*
- Stops *(rule II)*
- West Perth *(rule II, polygon found)*
- Bus Stops *(rule III)*
- West Perth Bus *(rule III)*
- Perth Bus Stops *(rule III)*

The search stops as soon as at least one result is returned by the WFS. Otherwise, the query is assumed to be of type IV.

As the user's query is not subject to any restrictions on spatial operations, it is then attempted to match the geographic location (where possible) in the text query to a polygon region (step 2a in Figure 2). Depending on the data source used, this polygon may be located in a database or accessed via a web service. It is this polygon that is then used, in conjunction with the available spatial operations, to restrict the search to more relevant results.

In some cases, it may be preferable to use a point rather than a polygon, for example spatial queries using the 'near' operator. This requires a second service that returns points rather than polygons. This can be complex, as the centroid is not always an accurate indicator of a point representing regions, as such a number of results may be needed.

The Spatial Identifier Reference Framework (SIRF) [15] was investigated for determining location information, however a method was chosen independent of SIRF that allows further extensibility and modularity. SIRF is a developing repository of location information about features within Australia, linking records where they appear in more than one dataset. SIRF primarily uses the same source data for Australia, with the linked data aspect of the system not required for the purposes of this system - only a single 'ground truth' is needed, rather than cross-referencing (assuming the supplied WFS is authoritative). There are disadvantages to not using SIRF; namely that alternative boundaries can be chosen using SIRF; ideally this should not be an issue, but calls to the SIRF API did not return the alternatives during exploration.

To determine the matching boundaries, a call is made to the chosen WFS used for the boundaries. A list is returned based on features matching the below criteria specified in the GET query string (Figure 3), where `LAYER` is the layer in which the boundaries are stored, `NAMESPACE` is the namespace used by the WFS, `PROP_NAME` is the name of the property storing the boundaries and `LOCATION_TERM` is the 'Location' term from the user's query. The approach allows the WFS to search for any properties in the boundary layer matching and containing the location. This is effected through a WFS filter on the boundary data set.

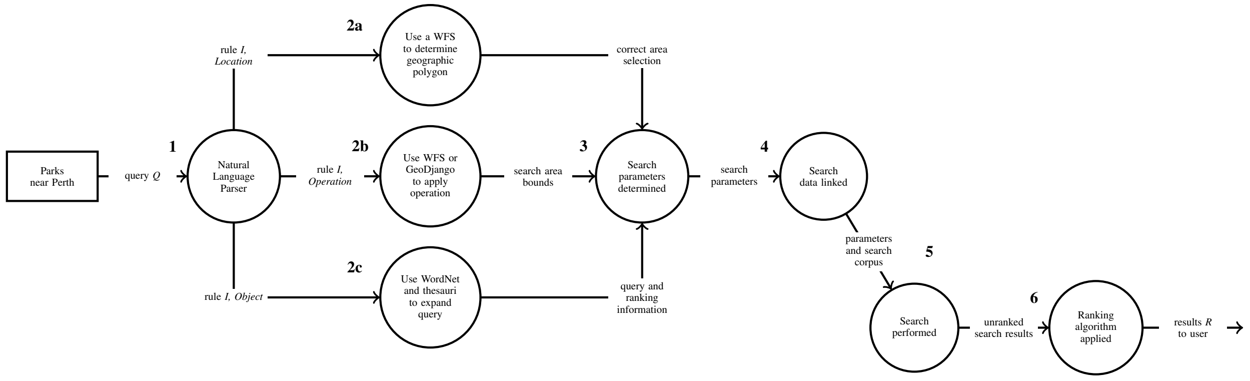

Figure 2. Diagram of the query process

```
?service=WFS&version=1.1.0&request=
GetFeature&outputFormat=json&typeName=
NAMESPACE:LAYER&propertyName=NAMESPACE:
PROP_NAME,NAMESPACE:the_geom\&filter=
<Query><Filter><PropertyIsLike wildCard=
"*" "singleChar="." matchCase="false"
escape="!"><PropertyName>NAMESPACE:
PROP_NAME</PropertyName><Literal>
*LOCATION_TERM*</Literal>
</PropertyIsLike></Filter></Query>
```

Figure 3. WFS Call for administrative boundaries

In cases where there are multiple alternatives to the boundaries, the user is able to choose from a list the boundary they intended, with the least specific area (i.e. largest sized) initially assumed. However, the query results will take into account all alternatives for the boundary, with the ranking positively influenced by the increased size of the area.

Semantic resources allow for this approach to be extended further. Using data stored in unstructured reports, including those behind firewalls, thesaurus graphs of related words can be generated alongside including information about related geographic areas. Using this information, similar areas are also searched but ranked lower based on a combination of the 'distance' of each word from the original, due to being less relevant. The information is stored as an RDF graph, which is explained in more detail in Section III-B.

If the query is of type I, a WFS call will be made to check whether the particular source supports spatial operations. If it does not, the spatial operation will be performed manually using the GeoDjango extension to Django, which allows spatial operations to be performed on data sets. In query types II or III, where this is only a geographic area specified, queries will be reduced to data falling within the boundaries of the polygon – in effect, an "in" spatial operator will be assumed. For query type IV, no processing of this type is completed.

Most current tools can only indirectly restrict results to spatial criteria via text searches - that is, to results that contain the geographic terms within the metadata records, or allowing the user to restrict their query to data sets that contain features within a bounding box. However, they cannot take advantage of spatial data sources with the ability to apply more specific spatial operations such as 'in', 'near' or 'intersects'.

Two methods are used to achieve this: either the built-in function as part of an OGC-compatible web service data set (where available), or as a manual spatial operation through GeoDjango. As this data is extracted from the user's text query, the method is hidden from the user, improving usability of the interface. These syntax and implementation specifics should not be required in a clean interface focusing on natural language queries, as they complicate the interface and are unlikely to be known by users seeking data.

As shown in Figure 4, after parsing the query to determine the spatial operation required in step 1 (of Figure 4), the system adjusts to the web services available operations by then requesting and searching its capabilities, as seen in step 2a. It can then select the most specific operation available to it, following a sequence of possible operations such as:

1) Operations include "Or", "Intersects", and "Within".
    - Return all records that are within or intersect with the boundaries of the comparison polygon(s).
2) Operations include "Within".
    - Return all records within the comparison polygon(s).
3) Operations include "BBOX".
    - Convert the comparison polygon(s) into a bounding box.
    - Return all records within the bounding box.
4) No relevant operations found.
    - Return all records.

The boundary polygon on which to apply the operation is then retrieved in step 2b (of Figure 4). Retrieving boundary polygons can go beyond simple geocoding of region names from WFS's. For example, a query for "bus stops *near* me" can make use of a user's location, obtained from their web browser, and create a buffer around that point location.

At processing step 3, the capabilities of the data source can be combined with the search areas from step 2b, depending on the system's capabilities. If the contents of a WFS contaning bus stop data were searched, for instance, the point features could be filtered by multiple polygons or, if the WFS does not have this capability, a single bounding box is created from the polygons and used as an alternative filter.

Once the geographic location and spatial operation has been determined, the rest of the query is interpreted similar

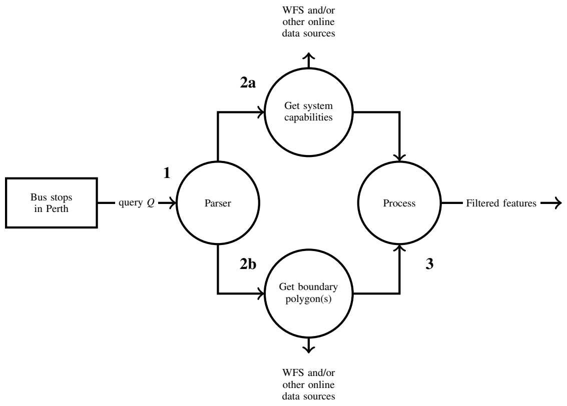

Figure 4. WFS spatial operations

to a traditional 'keyword' search, but with the query expanded using the semantic graphs. Query Expansion is performed to ensure that a broader base is used to find relevant metadata to the user's query, effectively creating and using multiple queries which are ranked depending on relevance.

The NLTK [16] toolkit is used to determine similar words and phrases to that supplied by the user. Using one of the built-in similarity ranking algorithms within NLTK, a ranking is applied to how similar an expanded query is to the original. A keyword-match of each of these queries is undertaken against the metadata records of the OGC-compatible web service data sets supplied to the system. The results are then ranked by a combination (where applicable) of the similarity of the query, the level of matching of the keywords and the geographic proximity of the results (in the case of a spatial operation such as 'near').

Through leveraging WordNet, GeoMeta is able to achieve a similar ability to Google's web search to match indirect queries, through query expansion of metadata corresponding to geospatial data sets. This consists of comparing queries using a ranking algorithm against the content of metadata records (as used in the existing approach by GeoNetwork et al.), but also through other sources of metadata. Metadata records are sourced from data sets they are attached to, located using OGC-compatible web services. Other sources of metadata are generated from unstructured reports and similar documents.

The NLTK is able to provide an interface to WordNet that links related words such as synonyms. For example, if the user entered the word 'park' as part of their query, the system would also look for 'reserve' as well. It has been previously shown [17] that it is feasible to use WordNet for query expansion. As such, investigation is ongoing as to the best way to use WordNet for this purpose. As the interface allows the user to determine and supply also the particular type of the word (noun, verb), relevant expansions can be determined. In cases where it is likely that the query has two meanings and the NLTK is unable to determine which, the user is asked to choose their intended meaning from a choice of possibilities, or the user can choose the option to rank both equally. A dictionary of domain-specific terms is also used in parallel and in the same manner, explained in more detail in Section III-B.

Investigations are ongoing to determine if any of the many similarity algorithms within WordNet are suitable. As each algorithm weights differently the relationship between two words, testing is being undertaken to determine which is most accurate in the context of GeoMeta. The similarity algorithm will then be used as part of the ranking algorithm. The system can also easily by extended by the user's own controlled vocabulary ontologies either automatically generated by an extension to the software or manually by the user.

Finally, the result set is returned to the user interface in JSON format (a lightweight data container used to store complex data), through the callback of the original AJAX request. The Google Maps API is used to visualise the data set by displaying a bounding box of the data on a map, alongside some traditional text-based metadata (such as a description) and a link to the data set being returned.

B. Use of Semantics in Geospatial Search

The use of semantically linked data greatly increases the relevance of returned results. Semantic graphs allow data to be linked together by meaning, and as such can be used to extend the context of the user's search query. In the context of this search system, similar locations are linked together alongside similar domain terminology in a 'thesaurus'-type format. An RDF schema is proposed for each of these, with the schema for the locations being influenced by the ISO 19115 RDF schema. This allows further expansion of queries to be matched by the algorithm.

ISO 19115 is the de facto standard for defining relevant metadata for geospatial data sets, providing a set of mandatory and optional metadata [10]. It is advantageous for use within a geospatial search system, as the standard allows for searching over the description and classification of a wide range and type of geographic metadata. Examples of this metadata includes traditional text-based descriptions alongside more detailed information about the physical, spatial and lineage aspects of the corresponding data set [18].

In practice, ISO 19115 is rarely used to its full potential, due to the burden of manually generating the required metadata and large quantity of optional fields [19]. OGC-compatible web services only require the bare minimum of the standard to be complied with, as that data is used as part of the 'GetCapabilities' function. For these reasons, often very little additional use of ISO 19115 is described in said web services.

As part of this search system, the extension of metadata available in ISO 19115 to a more flexible representation is proposed, which would allow more comprehensive coverage of data sets and hence more likely to satisfy a variety of user needs.

This would make use of the automatic generation of linked data from non-structured sources. Each of these allows the extension of queries into logically similar but syntactically different forms, therefore catching more metadata record resources for each query than keyword-matching alone. This therefore returns more complete search results. Such an approach therefore reduces the burden for both the user and the data custodian, as this information will be generated automatically.

The RDF metadata graphs are expressed through an RDF schema, which detail the kind of metadata terms of interest to be extracted from unstructured documents (such as reports or PDF files) and linked together based on lexical distance within

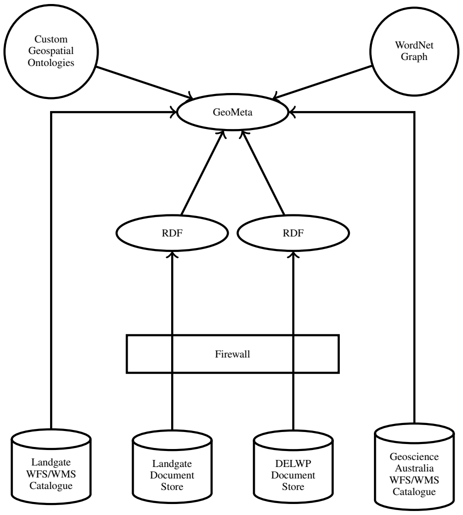
Custom Geospatial Ontologies
WordNet Graph
GeoMeta
RDF
RDF
Firewall
Landgate WFS/WMS Catalogue
Landgate Document Store
DELWP Document Store
Geoscience Australia WFS/WMS Catalogue

Figure 5. Data sources for GeoMeta

the text. In this way, it is similar to the WordNet graph used for query expansion, but rather than the links being determined by the similarity of the word, it is determined by both the similarity and distance within the text. This technique is used directly for the purposes of the domain terminology schema, whereas it is applied only for locations with respect to that schema.

The generation of schema-complying RDF descriptions is achieved through the use of existing tools which already search unstructured data. Voyager [20], Sintelix [21] and Omniscient [22] are being explored as part of the project. Plugins are being written to convert the underlying databases of these products to RDF descriptions compliant with the schemas. These tools allow the automated creation and updating of the RDF descriptions with minimal user input.

The design of the interoperable schema allows any tool to be used for this purpose in a plug-in modular format, as long as the tool can produce RDF that complies with the schema. The design of this architecture allows custodians to run the tool behind a firewall, only exposing the minimum data required for the schema, allowing the use of a much greater corpus of metadata that would often be left behind the firewall. This is a benefit over existing tools, which are generally restricted to searching the public Web, or private networks through basic authentication.

The sources of data for the case study of the search system can be seen in Figure 5, with the RDF metadata being the central two sources. Each data source can be used to generate data complying to either schema, depending on content is within the file. For the location schema, such data as described in the ISO 19115 standard is generated from unstructured documents.

Alongside geographic information, descriptions of more ‘general’ terms of relevance related to physical ‘things’ will be generated, which may aid finding results for both the spatial and non-spatial component of the user’s query. Examples of such extra fields are ‘Person’, ‘Organisation’ and ‘Currency’. These are determined by the above software tools, as they are designed to look for specific types of information. ISO 19115 has already been generated into RDF [5], and is explored, modified and expanded for this purpose.

IV. Results

A system named GeoMeta has been produced, consisting of a user facing front-end (where the user enters a text-based query) and a server side back-end. The front end, written using HTML5 and jQuery, provides a native application-like experience giving more feedback to the user. For example, as the text queries are edited, the results are updated on-the-fly, and loading screens are provided to show that the system is processing a query. This is achieved through Asynchronous JavaScript and XML (AJAX) queries from the front-end to the back-end.

The back-end system is written using the Django [23] framework in the Python programming language. Modular components were designed to allow for new features to be dropped in. This enables, for example, the WordNet interface (used in part to expand queries) to be exchanged with a comparable system. Currently, this interface is provided through the NLTK [16] library. It is entirely possible to substitute other thesauri graphs through the NLTK library (or other libraries) to use them instead. The NLTK connects locally to downloaded versions of the WordNet corpora. WordNet is able to be used as a service from a remote server [24]; however as the corpora is static this method is not being used.

The first version of the prototype is a demonstration of using a text-based query that can be split into both spatial and non-spatial components. The system used the Google Geolocation API to determine a bounding box of the location, and was fixed into only being able to search using the ‘in’ spatial operator. As a data source, the Shared Land Information Platform (SLIP) [25] of Landgate, the Western Australian Land Information Authority, was used. This was accessed through the Google Maps Engine API, however as the API will soon be discontinued, future versions will support different ingest mechanisms for data sources (namely OGC-compatible web services).

The GME API allowed the dataset title, description, metadata tags and bounding boxes to be extracted to create an effective metadata record for each data set. Although there are other metadata fields available for use within GME, these were rarely used. Hence, these three fields were the only ones used through the API. This data was generated manually by Landgate, and contains some gaps and repetitive template text with limited relevance.

Many of the descriptions followed a standard format consisting of generic information about the agency and contact information, which negatively influenced the search results. For example, a search for ‘imagery’ had many matches because the agency is described as supplying imagery, even when the dataset itself did not contain any. This data is also not fully viewable without authentication, hence a system such as proposed in this paper would solve both of the above issues of inadequate metadata and the metadata being behind a firewall.

This demonstrator proved that even the basic additions of spatial filtering proved useful. The ability to filter spatially allowed results that were not fit for purpose, by being located in other areas of Western Australia, to be excluded from the search results, leaving only more relevant candidate data sets for consideration.

A second demonstrator, currently under development, uses a more advanced natural language processing classifier as described in Section III-A. This enables the program to sort queries into four types, allowing more advanced spatial operations to be performed. The second version allows these to be performed but only through the GeoDjango method. A rudimentary version of the WordNet graph used for query expansion has been implemented, alongside the use of OGC-compatible web services as data sources due to the retirement of the Google Maps Engine API. This also allows the system to sit on top of existing systems such as FIND [26] (a GeoNetwork instance publishing many datasets with the custodian being the national Australia geographic agency) and other GeoNetwork installations (which primarily catalogue OGC-compatible data sets). This architecture reduces the work required by custodians to exploit the GeoMeta system.

Small-scale tests on parts of the second system have been undertaken on each component to determine their suitability. A ‘region finder’ has been implemented based on the Administrative Boundaries data set from the Australian Bureau of Statistics, available through the Australian Government’s ‘National Map’ [27]. This component of the system works successfully, with polygons being returned for various types of boundaries that match the user’s ‘Location’ part of the query. The polygons returned are are as expected. The ‘rule classifier’ has also been built, and has been successful in categorising a wide range of queries into the relevant type. Tests continue to fine-tune and improve the system for more advanced uses, such as long ‘Location’ terms for rules II and III that exceed three words in length.

The method to apply spatial operations built in to some OGC-compatible web services has been explored for suitability. The method has the benefit of being able to offload some of the heavy processing to remote servers, reducing response time and the need for caching (as the queries will need to be processed live on the remote server). It is possible that remote server load issues could be encountered, and as such a method will need to be determined as to when the local processing should be used instead. Indeed, this has already happened in informal tests. Formal tests are being conducted to determine the best way to work out if a OGC-compatible web service will be ‘too slow’, as the apparent processing speed is a function of latency, data set complexity, the processing power of the remote machine and other factors, not all of which are able to be determined ahead of time.

Examination of the three tools used to automatically generate metadata from unstructured documents show that they are all able to pick out relevant information that can then be used within an RDF schema. Comparisons of these results are being undertaken to measure the quality of results from each system. Initial results show that all three tools are suitable to generate useful metadata.

The system can be investigated at http://research.haxx.net.au/geometa where a prototype of the system resides.

V. Conclusion and Future Plans

This paper has presented a search algorithm to overcome many of the limitations of using contemporary geospatial search engines to find spatial data relevant to a user’s query. The algorithm presented extends upon the traditional search technique of keyword-matching the user’s query with the content of metadata records, by using natural language processing to split a query into spatial and non-spatial components.

The splitting of the spatial component of the query allows sophisticated spatial operations to be undertaken by the user to find relevant data sets that satisfy the operation, defined by bounding boxes and polygons. The use of a natural language text-based interface enables a user-friendly experience that more easily articulates the users’ intentions, compared to a map image-based input system for spatial queries. This reduces effort and uncertainty, as well as providing better results than a pure keyword-matching approach.

The design of a prototype, GeoMeta, is presented as well as some results from an initial proof of concept based on a case study with Landgate, the Western Australian Land Information Authority, that has been using the Google cloud to store spatial data which is accessed using GME. This experience indicated that there are advantages in not only opening up data sets to be indexed by search engines, but also having data custodians run automatic metadata-generating software over their own internal document repositories to generate improved metadata.

Future plans include the integration of automatically generated metadata. As this metadata will be generated by third-party tools in a format complying with the proposed RDF schema, future research will need to be able to interpret these and link them in with existing data. This will be achieved by using the data within the RDF files to further facilitate query expansion, ranking results by distance in the same manner as the existing system.

The WordNet algorithm currently implemented will be fine-tuned to better articulate the users’ intention to ensure that the query expansion is relevant, and will remove automatically generated ‘nonsensical’ queries from those being searched. This will ensure that only relevant expanded queries are used, improving response time to the end user. Future research will also allow the user to augment this with their own ontologies, allowing more domain specific terms to be used in queries.

Finally, polygon comparison will be implemented on the data sets where available. This will further enhance the accuracy of results when compared to a bounding box comparison. Further investigation is needed to determine whether it is feasible to process data files when the polygon is not exposed via a Web Feature Service. Polygons will also be displayed in the preview map image in later versions.

VI. Acknowledgement

The work has been supported by the Cooperative Research Centre for Spatial Information, whose activities are funded by the Business Cooperative Research Centres Program.

References

[1] Alphabet, Inc., “Google,” https://google.com/, 2016, [Online; accessed 2016-02-10].

[2] S. Brin and L. Page, “The anatomy of a large-scale hypertextual web search engine,” Computer Networks and ISDN Systems, vol. 30, no. 1/7, 1998, pp. 107–117. [Online]. Available: http://dx.doi.org/10.1016/S0169-7552(98)00110-X

[3] S. Grill and M. Schneider, “Geonetwork opensource as an application for SDI and education,” in GIS Ostrava, 2009, pp. 25–28. [Online]. Available: http://gis.vsb.cz/GIS_Ostrava/GIS_Ova_2009/sbornik/Lists/Papers/039.pdf

[4] C. Jenkins, M. Jackson, P. Burden, and J. Wallis, "Automatic RDF metadata generation for resource discovery," Computer Networks, vol. 31, no. 11, 1999, pp. 1305–1320.

[5] A. Saiful, "Development of a web-based modeling system using metadata concepts and databases," Doctoral Dissertation, Drexel University, 2004.

[6] Alphabet, Inc., "Google Maps Engine," https://developers.google.com/maps-engine, 2015, [Online; accessed 2015-10-12].

[7] Open Knowledge Foundation, "ckan - The open source data portal," http://ckan.org/, 2016, [Online; accessed 2016-02-10].

[8] Open Source Geospatial Foundation, "GeoNetwork opensource," http://geonetwork-opensource.org/, 2016, [Online; accessed 2016-02-10].

[9] N. Chen, X. Wang, and X. Yang, "A direct registry service method for sensors and algorithms based on the process model," Computers Geosciences, vol. 56, jul 2013, pp. 45–55. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0098300413000691

[10] K. Senkler, U. Voges, and A. Remke, "An ISO 19115/19119 Profile for OGC Catalogue Services CSW 2.0," in 10th EC GI GIS Workshop, Warsaw, Poland, 2004.

[11] A. Friis-Christensen, M. Lutz, and N. Ostländer, "Designing Service Architectures for Distributed Geoprocessing: Challenges and Future Directions," Transactions in GIS, vol. 11, no. 6, 2007, pp. 799–818.

[12] M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich, "A Review of Relational Machine Learning for Knowledge Graphs From Multi-Relational Link Prediction to Automated Knowledge Graph Construction," 2015.

[13] W. Li, "Automated data discovery, reasoning and ranking in support of building an intelligent geospatial search engine," Doctoral Dissertation, George Mason University, 2010. [Online]. Available: http://ebot.gmu.edu/handle/1920/6013

[14] Django Software Foundation, Django Documentation (Release 1.8.6), 2015. [Online]. Available: http://media.readthedocs.org/pdf/django/1.8.x/django.pdf

[15] Commonwealth Scientific and Industrial Research Organisation, "SIRF," http://portal.sirf.net/, 2015, [Online; accessed 2015-12-16].

[16] NLTK Project, "Natural Language Toolkit - NLTK 3.0 documentation," http://nltk.org/, 2016, [Online; accessed 2016-01-15].

[17] D. Buscaldi, P. Rosso, and E. Arnal, "A WordNet-based Query Expansion method for Geographical Information Retrieval," in Working notes for the CLEF workshop, 2005. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.99.8031&rep=rep1&type=pdf

[18] Standards Australia, "AS/NZS ISO 19115.1 Geographic information Metadata Part 1: Fundamentals," Tech. Rep., 2015.

[19] O. Karschnick, F. Kruse, S. Töpker, T. Riegel, M. Eichler, and S. Behrens, "The UDK and ISO 19115 Standard," in Proceedings of the 17th International Conference Informatics for Environmental Protection EnviroInfo, 2003.

[20] AAM, "AAM Group — Geospatial Excellence," http://aamgroup.com/, 2016, [Online; accessed 2016-01-21].

[21] Semantic Sciences, "Sintelix - gaining value from your corporate documents," http://sintelix.com/, 2016, [Online; accessed 2016-02-09].

[22] Omnilink, "Omniscient Spatial Metadata — OMNILINK Property and Location Data Management," http://omnilink.com.au/products/omniscient-spatial-metadata/, 2016, [Online; accessed 2016-02-09].

[23] Django Software Foundation, "The web framework for perfectionists with deadlines — Django," https://djangoproject.com/, 2016, [Online; accessed 2016-01-15].

[24] A. Fred, J. L. G. Dietz, K. Liu, and J. Filipe, Knowledge Discovery, Knowlege Engineering and Knowledge Management. Springer, 2011. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-19032-2

[25] Landgate, "SLIP Home," http://slip.landgate.wa.gov.au/, 2016, [Online; accessed 2016-02-09].

[26] Department of Communications, "Find - Office of Spatial Policy," http://find.ga.gov.au/, 2016, [Online; accessed 2016-02-09].

[27] Australian Government, "NationalMap," http://nationalmap.gov.au/, 2016, [Online; accessed 2016-02-09].