applied sciences

Article

A Query Understanding Framework for Earth Data Discovery

Yun Li [1], Yongyao Jiang [1], Justin C. Goldstein [2], Lewis J. Mcgibbney [3] and Chaowei Yang [1,*]

[1] NSF Spatiotemporal Innovation Center, George Mason University, Fairfax, VA 22030, USA; yli38@gmu.edu (Y.L.); yjiang8@gmu.edu (Y.J.)
[2] Riverside Technology, Inc. Supporting the National Oceanic and Atmospheric Administration (NOAA) Technology, Planning, and Integration for Observation Division, Fort Collins, CO 80528, USA; justin.goldstein@noaa.gov
[3] NASA Jet Propulsion Laboratory, Pasadena, CA 91109, USA; Lewis.J.Mcgibbney@jpl.nasa.gov
* Correspondence: cyang3@gmu.edu; Tel.: +1-703-993-4742

Received: 8 January 2020; Accepted: 4 February 2020; Published: 7 February 2020

Abstract: One longstanding complication with Earth data discovery involves understanding a user's search intent from the input query. Most of the geospatial data portals use keyword-based match to search data. Little attention has focused on the spatial and temporal information from a query or understanding the query with ontology. No research in the geospatial domain has investigated user queries in a systematic way. Here, we propose a query understanding framework and apply it to fill the gap by better interpreting a user's search intent for Earth data search engines and adopting knowledge that was mined from metadata and user query logs. The proposed query understanding tool contains four components: spatial and temporal parsing; concept recognition; Named Entity Recognition (NER); and, semantic query expansion. Spatial and temporal parsing detects the spatial bounding box and temporal range from a query. Concept recognition isolates clauses from free text and provides the search engine phrases instead of a list of words. Name entity recognition detects entities from the query, which inform the search engine to query the entities detected. The semantic query expansion module expands the original query by adding synonyms and acronyms to phrases in the query that was discovered from Web usage data and metadata. The four modules interact to parse a user's query from multiple perspectives, with the goal of understanding the consumer's quest intent for data. As a proof-of-concept, the framework is applied to oceanographic data discovery. It is demonstrated that the proposed framework accurately captures a user's intent.

Keywords: geospatial cyberinfrastructure; GeoAI; CyberGIS; semantics; big spatiotemporal data analytics

1. Introduction

Effectively discovering Earth science data is challenging, given the data's increased volume, decreased latency, and heterogeneity across a wide variety of domains [1]. One longstanding problem with Earth data discovery is interpreting a user's search intent from the user's input query. While Google has a "did you mean this . . . " feature and Amazon has the "search instead for . . . " capability, most geospatial search engines do not support such kind of search features. Geospatial search engines that rely on keyword-based match search strategy and lack the capability of phrase query. For example, the query "level 2" is converted to "level" or "2" to retrieve data and dataset containing either "level" or "2" will return to the searcher. However, the real intent of the user is to find the dataset of processing level 2. While humans understand the intent of a query, it is difficult for a machine to capture the desired meaning, because user queries are just a list of words and they commonly have one or several of the following characteristics: incomplete sentences that are hard for machine's syntax analysis; plethora of acronyms in-lieu of full-names; absence of semantic context; and, descriptions of the same object in different forms. Some geospatial data portals have made great efforts to boost their search capabilities by introducing advanced technologies from computer science domain or customized configurations, e.g., PO.DAAC data portal introduced "Google-like" query syntax to support phrase query with input query "level 2" [2]; GeoNetwork [3], an open-source, distributed spatial information management system, indexes geospatial data, and supports data discovery upon Lucene [4]. Relevant keywords will not be smashed apart in the indexing and searching process, with advanced configuration of indexed fields and search fields. Although these features optimize the search performance, it requires users to learn related syntax or understand the indexing workflow. Implicit knowledge hidden in metadata and Web usage data are extracted in our research to improve query understanding to meet this gap and reduce the workload of data consumers/publishers. Domain experts generate the metadata and they contain reliable terminologies in the domain. A query understanding framework is proposed to better interpret users' search intents in Earth data search engines by analyzing metadata and user query logs with advanced Natural language processing algorithms to make good use of valuable query-related knowledge hidden in metadata and Web usage data [5–7]. The query understanding framework consists of four major steps: spatial and temporal parsing; concept recognition; named entity recognition (NER); and, semantic query expansion. With the four components, a user input query is rewritten to a new query, which helps a search engine capture a user's search intents by (1) detecting spatial and temporal range of a query; (2) searching with concepts and their synonyms instead of independent words; and, (3) narrowing search scope with name entity being recognized from the query. An oceanographic data discovery portal is utilized for evaluating the utility of the query understanding framework.

2. Related Research

Earth data discovery has been an active research area in the past few years [8]. Many geospatial information data management portals have been deployed across the world to support interactive data access [9,10], e.g., the European space agency's sentinel online data portal—Copernicus Open Access Hub [11], CIESIN, which serves data for climate, population, soil, etc. [12], the FAO GeoNetwork that contains data, like soil, population, and land use data from global to regional scale [13]. Some of the data portals support data discovery through utilizing distributed search and analytical engine, such as Lucene [4] and Elasticsearch [14], some are built upon comprehensive data portal platforms, such as GeoNetwork [3], GeoNode [15], and CKAN [16]. Various researches have focused on optimizing the search process, such as optimizing the search ranking of retrieved datasets [17], supporting reasoning with semantic ontologies [18,19], and providing query understanding [5], to further improve data search capability. Query understanding serves as a communication channel between users and the search engine before the search engine retrieves and ranks results, and in so doing tries to understand search queries and interpret the intent of a query through multiple methods (e.g., extraction of the semantic meaning of the searcher's keywords [5,20]).

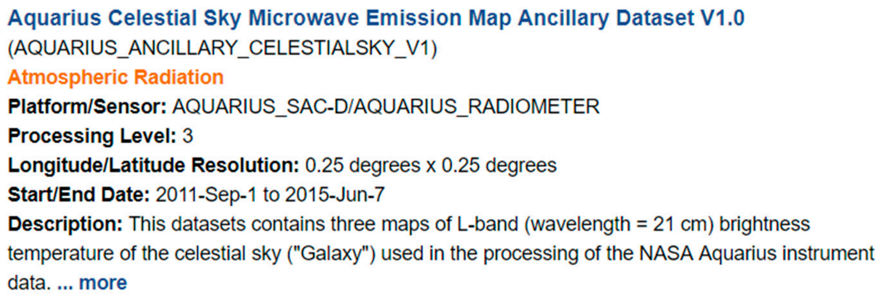
A few existing libraries are available for spatial and temporal parsing. For example, the Google Maps Geocoding API and CLAVIN [1] convert the location names into geographic coordinates. Stanford Temporal Tagger (SUTime) [5] tags the temporal component of a query. Some Earth data search engines have adopted CLAVIN and SUTime to parse the spatial and temporal range from user queries [7]. Wang, e.g., developed a fuzzy grammar and theory-based natural language user interface for spatial queries [21]. However, very few attempted to parse and tag the non-spatial and temporal components of the query syntax, which usually consists of entities, like geophysical variable, instrument name, and processing level. Phrase detection has been applied to large-scale natural language data processing in the search engine and data discovery community [22]. Named entity recognition has been used for learning dictionaries with minimal supervision [23]. Keyword-based geospatial data search engines regard input queries as a bag of words without considering the order, and return results, including at least one word in the query. Advanced geospatial search engines that provide customized configurations or "Google-like" syntax require data consumers/publishers to learn relevant knowledge. We propose a method of phrase detection to address this challenge, and named entity recognition from user query by leveraging metadata and portal usage data.

In addition, a few Earth science researchers have been devoted to building semantic models or ontologies in the Geoscience domain to make domain knowledge computable, organizable, and transparent in standard and formal structure. These semantic models include, but are not limited to, thesauri [24], glossaries [25], and the geospatial ontology Semantic Web for Earth and Environmental Terminology (SWEET) [26]. Taking SWEET, for example, it organizes high-level concepts and relations in the geospatial domain. Concepts and their relations in a knowledge base provide a valuable source for discovering the vocabulary linkages, which can be adopted for reasoning and smart search. However, most existing Earth Science ontologies are not useable in the current research, because the concepts in the ontologies do not cover or are inconsistent with terms in the metadata, since some strong and heavyweight ontologies are very high-level and not detailed enough and others are lightweight or implicit [27]. Developing ontologies from scratch is time-intensive and labor-intensive; an alternative way is to discover latent semantic relationships semi-automatically from user behavior and metadata [19]. When compared to manually built ontologies, the relations and concepts discovered from metadata and logs are easy-to-get, up-to-date, and consistent with metadata, although they maintain a lack of quality control. No workflow has been proposed to translate a raw user query into a set of formatted easy-to-understanding semantic components. To meet this gap, we propose a query understanding framework to better interpret users' search intent by using vocabulary relationships mined from metadata and user query logs for query expansion.

This paper introduces the framework and relevant research in the following fashion: Section 3 introduces the data that were used in the experiment. Section 4 describes the conceptual framework of query understanding and the method of each step. Section 5 presents the experiments, and Section 6 discusses the result and evaluation. The final section offers key findings and discusses future research.

3. Data

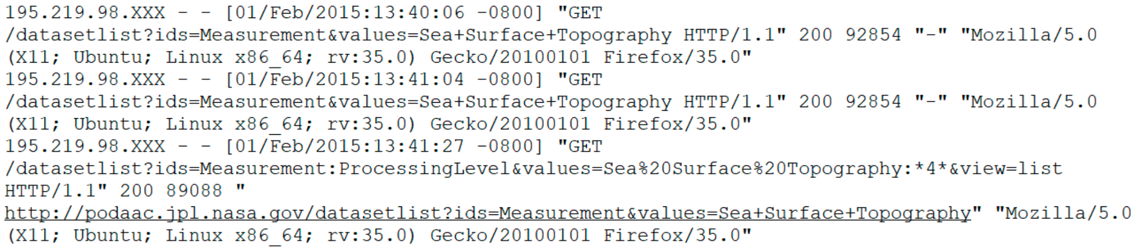
Two types of data are collected to train models in the proposed query understanding framework. The first, geospatial metadata, provides information that describes geospatial data in details (e.g., long name, short name, description, processing level, sensor, platform, category, term, topic, spatial/temporal resolution, start/end date of the dataset records) (Figure 1). Data publishers (e.g., Distributed Active Archive Center, DAAC, data engineers) usually provide metadata to help users find data and applications efficiently and accurately.



Aquarius Celestial Sky Microwave Emission Map Ancillary Dataset V1.0
(AQUARIUS_ANCILLARY_CELESTIALSKY_V1)
Atmospheric Radiation
Platform/Sensor: AQUARIUS_SAC-D/AQUARIUS_RADIOMETER
Processing Level: 3
Longitude/Latitude Resolution: 0.25 degrees x 0.25 degrees
Start/End Date: 2011-Sep-1 to 2015-Jun-7
Description: This datasets contains three maps of L-band (wavelength = 21 cm) brightness temperature of the celestial sky ("Galaxy") used in the processing of the NASA Aquarius instrument data. **... more**

Figure 1. A sample PO.DAAC metadata.

The second, data portal usage logs, records a user's interaction with the Web portal. A Web usage record in combined log format consists of client IP address, request date/time, page requested, HTTP code, bytes, referrer, and user-agent (Figure 2) [28]. Collectively, they identify users, track clicking behavior, and reflect search trends [29]. Logs can also be mined to find the implicit semantic relationship among geospatial vocabularies [30].



```
195.219.98.XXX - - [01/Feb/2015:13:40:06 -0800] "GET
/datasetlist?ids=Measurement&values=Sea+Surface+Topography HTTP/1.1" 200 92854 "-" "Mozilla/5.0
(X11; Ubuntu; Linux x86_64; rv:35.0) Gecko/20100101 Firefox/35.0"
195.219.98.XXX - - [01/Feb/2015:13:41:04 -0800] "GET
/datasetlist?ids=Measurement&values=Sea+Surface+Topography HTTP/1.1" 200 92854 "-" "Mozilla/5.0
(X11; Ubuntu; Linux x86_64; rv:35.0) Gecko/20100101 Firefox/35.0"
195.219.98.XXX - - [01/Feb/2015:13:41:27 -0800] "GET
/datasetlist?ids=Measurement:ProcessingLevel&values=Sea%20Surface%20Topography:*4*&view=list
HTTP/1.1" 200 89088 "
http://podaac.jpl.nasa.gov/datasetlist?ids=Measurement&values=Sea+Surface+Topography" "Mozilla/5.0
(X11; Ubuntu; Linux x86_64; rv:35.0) Gecko/20100101 Firefox/35.0"
```

Figure 2. Sample Physical Oceanography Distributed Active Archive Center (PO.DAAC) HTTP logs in Combined Log Format.

This research uses the Physical Oceanography Distributed Active Archive Center (PO.DAAC) metadata and the Web usage logs. PO.DAAC Metadata are harvested from the PO.DAAC website through its Web service API (https://podaac.jpl.nasa.gov/ws) and the user logs are provided by PO.DAAC. The PO.DAAC serves the Earth science community with NASA's ocean and climate data, including measurements focused on ocean surface topography, sea surface temperature, ocean winds, sea surface salinity, gravity, ocean circulation, and sea ice. All of the publicly available collection-level metadata are harvested from the PO.DAAC Web server. The PO.DAAC metadata are in Directory Interchange Format (DIF), a standard format in NASA Earth Science Data Systems [31]. In addition, the PO.DAAC Web portal usage logs are provided for vocabulary analysis.

4. Methodology

4.1. Query Understanding Framework Architecture

The conceptual query understanding framework has four components: spatial and temporal parsing; concept recognition; name entity recognition; and, semantic query expansion (Figure 3). Spatial and temporal parsing extracts spatial bounding box and temporal range from a query. Concept recognition isolates domain phrases from the query, and name entity recognition focuses on identifying entities, like people, place, and organization within a query. Finally, query expansion adds additional phrases (e.g., synonyms) to expand the original query. For a given query, the spatial bounding box (coordinates of northeast and southwest points of the spatial range) and time range are detected, and a query remainder is sent to the concept recognition module for domain phrase detection. For example, "pacific ocean" and "2003–2010" in the input query "sea surface temperature level 3 pacific ocean 2003–2010" are identified as spatial bounding box and time range, while "sea surface temperature" and "level 3" are domain phrases. The name entity recognition module recognizes "sea surface temperature" and "level 3" as a variable and processing level. The query expansion module augments extracted phrases with their synonyms and acronyms learned from Web usage logs and metadata. For example, "sea surface temperature" in the query is converted to "sea surface temperature OR sst". Once all of the spatial-temporal features, domain phrases, entities, synonyms, and acronyms are detected, the original query is converted to a new one, and the top k results are retrieved by the search engine while using the rewritten query.

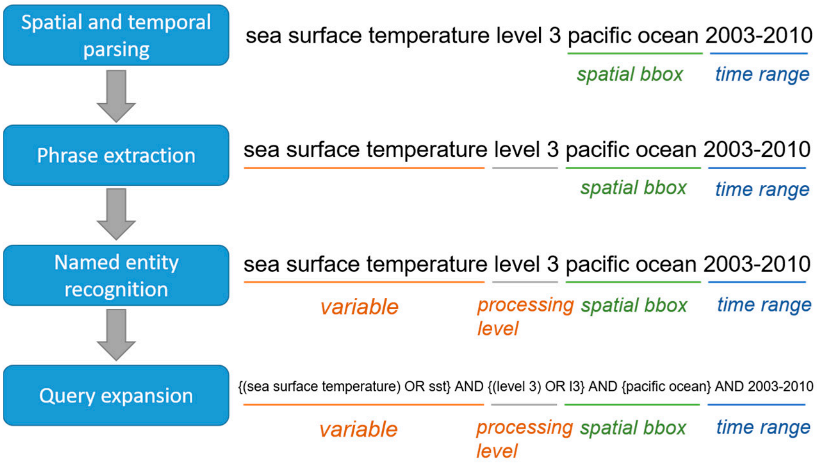

Spatial and temporal parsing
sea surface temperature level 3 pacific ocean 2003-2010
spatial bbox
time range
Phrase extraction
sea surface temperature level 3 pacific ocean 2003-2010
spatial bbox
time range
Named entity recognition
sea surface temperature level 3 pacific ocean 2003-2010
variable
processing level
spatial bbox
time range
Query expansion
{(sea surface temperature) OR sst} AND {(level 3) OR l3} AND {pacific ocean} AND 2003-2010
variable
processing level
spatial bbox
time range

Figure 3. A conceptual framework for query understanding.

4.2. Spatial and Temporal Parsing

Spatial bounding box and temporal range are two important factors in Earth data discovery. Two off-the-shelf tools are used in this framework. For geospatial parsing, an open source geocoding package [32] (Google Geocoding API client) converts the address in the unstructured query to latitudes and longitudes of spatial bounding box of the address. The Google Geocoding API [33] provides geocoding and reverse geocoding of addresses. Geocoding is the process of converting addresses into geographic coordinates to place the markers on a map [33].

The temporal attributes of geographical metadata (e.g., release date, coverage time) are structurally formatted. However, the date in the user query is often written in unstructured natural language, which is unmatched to the format of date in the relevant metadata. The solution is Dateparser [34], an open source python library for temporal parsing. Dateparser translates specific dates (e.g., 20 August 2018) into a DateTime object as the same date format metadata use to retrieve data falling into the time range.

Given a query, Google Geocoding API and Dateparser sequentially detect the spatial and temporal components. Although Google Geocoding API and Dateparser were selected to detect and transform spatial and temporal attributes in the framework, the two software packages are replaceable with similar open-source software, commercial software, or customized spatial and temporal parsers to accurately and efficiently parse spatial and temporal terms from a query.

4.3. Concept Recognition

In general, geospatial data search engines regard input queries as a bag of words without considering the order and returns results, which include at least one word in the query. For example, if a user searches “sea surface temperature”, the engine returns links to pages containing sea, surface, and/or temperature. For an Earth scientist, the real intent is searching the sea surface temperature products. If “sea surface temperate” is detected as a phrase, the search engine retrieves more relevant data to the query. To fill the gap between the query and a user’s real search intent, some geospatial data search engines boost search capability with customized configurations. The PO.DAAC data portal supports “Google-like” query syntax, i.e., quoting a set of words to search for datasets that contain the keyword phrase [2]. GeoNetwork provides portal configuration, in which users can edit search filter parameter with Lucene parser syntax. When combined with the metadata indexing strategy, such a kind of search filters could support phrase query and more complicated searches [35]. Although these functionalities improve search capability of data portals, data consumers need to spend time on learning related knowledge or know existing syntaxes that support composite searches. Two n-gram models trained from metadata in advance for concept recognition are used to extract terms in the query automatically to match the corpus to improve this.

In the field of computational linguistics, an n-gram is a contiguous sequence of n items from a given sample of text or speech. An n-gram model models such kind of contiguous sequence [36] and bigrams and trigrams are two popular n-gram models. Bigrams are possible word pairs in a sentence that formed from neighbouring words, while trigrams are possible combinations of three consecutive words [37]. With an n-gram model, contiguous sequence of n items recorded in it can be detected from a given sample of text. The concept recognition module relies on bigrams and trigrams models to capture terminologies that consist of two or three words from an input query. To prepare the bigrams and trigrams models for the concept recognition implementation, Gensim, [38], which is a free vector space and topic modeling toolkit designed to process the unstructured texts, train a bigram and a trigram model from PO.DAAC metadata with two parameters. The first, "min_count", is for any word or bigram to be ignored if its occurrence is lower than this pre-defined value. The lower the value is, more phrases are kept. The second, "phrase threshold", represents a threshold for forming the phrases. The higher the threshold, fewer phrases are detected. For example, a phrase of words a and b is accepted as a bigram if (count (a, b)—min_count) * N/(count (a) * count (b)) > phrase threshold, where N is the total vocabulary size and count(a) is the number of occurrences of a in the corpus.

The bigram and trigram model trained by Gensim can learn most two or three words phrases in the PO.DAAC metadata. When the search engine receives a query, the two pre-trained n-gram models work sequentially to translate the user input to domain phrases that are captured in the models. The extracted phrases are then passed to the query expansion module (Section 4.5) (Figure 4).

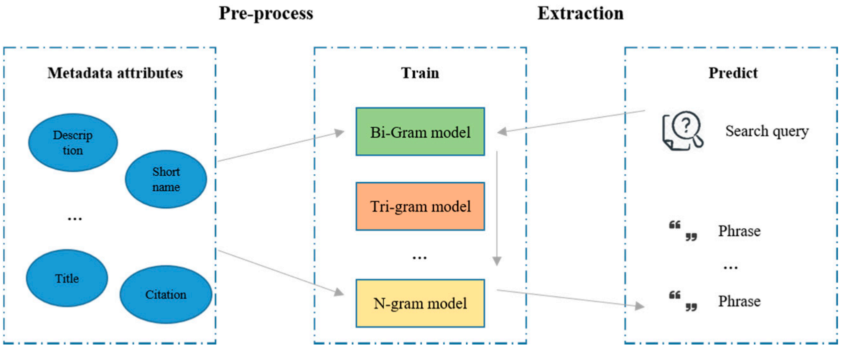


Figure 4. Workflow of -concept recognition.

4.4. Named Entity Recognition

Geospatial search query consists of significant entities, including geophysical variable, version, sensor, and processing level, etc. In addition to the query input search box geospatial data portals usually provide filters (e.g., processing level, sensors) to narrow the search context. With search filters, users can filter and customize search results to certain content types, e.g., to only obtain images as search results on Google, to get data only in NetCDF format from the PO.DAAC data portal. If these entities are automatically detected from user queries, it is unnecessary for a user to set filters to help the search engine efficiently retrieve datasets. When a user sets filters, entities recognized from a query can support filter verification. For example, when a user searches "sea surface temperature level 2", but sets the processing level filter to "level 3", the search engine cannot return datasets of processing level 2. The processing level detected from the input query in this example can help the search engine to figure out the issue by either changing the filter to "level 2" or giving prompts about the mismatch between query and filter settings.

Name entity recognition is a process where an algorithm identifies specific types of nouns from a string of text [39]. While people, places, and organizations are the most common types of entities a NER algorithm detects, a NER algorithm can also recognize user-defined entities with the trained data. In our research, pre-defined entities fall in following types: variable, parameter, processing level, version, collection, and sensor. Queries with entities manually labeled by domain experts are expensive and ineffective to collect. Thus, this research proposes a method that automatically generates train data from metadata attributes to train a customized NER model.

Each metadata attribute belongs to one of the following categories: entity attribute; query attribute; and, irrelevant attribute. Irrelevant attributes are ignored. Entity attributes (e.g., metadata processing level) serves as phrases sources to entities. For example, the processing level entity consists of level 1, level 2, level 2p, level 3, and level 4 in PO.DAAC metadata. The query attributes provide numerous query-like sentences or expressions. Entities that are discovered from entity attributes are labeled in these sentences for training. Many metadata fields are regarded as query attributes (e.g., short name, long name, title of citation). A NER model is trained with the Stanford CoreNLP package [40], an open source software providing various Natural Language Processing tools. This research focuses on incorporating the advanced research in computer science with existing geospatial data to solve issues in Earth science. For an input query, the NER model recognizes user-defined entities, which are leveraged to automatically set or optimize filters (Figure 5).

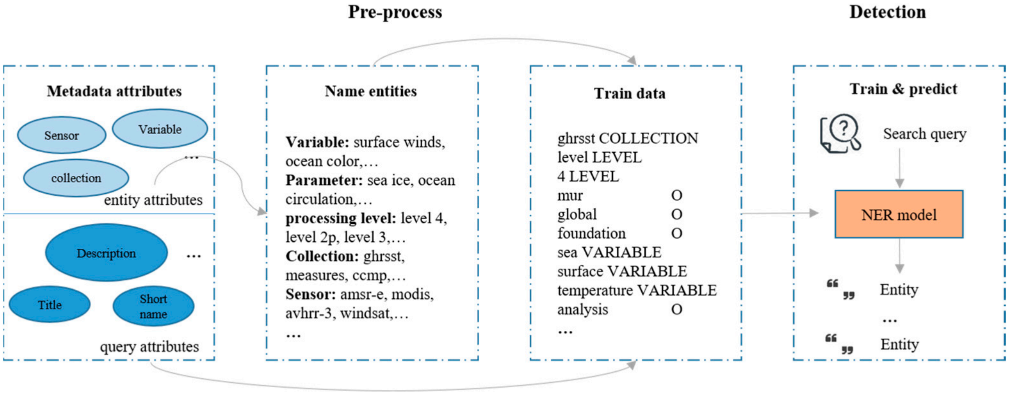


Figure 5. Workflow of named entity recognition (NER) train data preparation, train, and prediction.

4.5. Query Expansion

The query expansion module rewrites the original user query by adding synonyms and acronyms of phrases existing in a knowledge base learned from user query logs and metadata (Figure 6). The knowledge base stores the semantic similarity between domain vocabularies that are computed by a similarity calculator ingesting metadata and Web usage logs. Queries in historical usages logs and terminologies in metadata attributes are the two sources for vocabularies in the knowledge base. For queries, the hypothesis of measuring their similarity is if two queries are similar: (1) they would co-occur in distinct users' search history more frequently and (2) the overlap between datasets clicked after searching the two queries would be larger in the context of large-scale clickstream data. The assumption of measuring terminologies similarity is that two vocabularies have a higher probability of appearing in the same metadata if they are similar to each other. Accordingly, query-user, query-data, and terminology-data co-occurrence matrices are constructed from user logs and metadata for similarity calculation. The Latent Semantic Analysis (LSA) [41] is applied to these co-occurrence matrixes to discover hidden semantic relations among vocabularies that consist of queries and terminologies. The independent vocabulary similarity scores derived from the three matrixes validate each other and become more credible if integrated while using borda voting to a final score ranging from 0 (i.e., no relation) to 1 (i.e., identical). These scores are stored in the knowledge base and periodically updated to learn new knowledge introduced by new metadata or interaction data [30,42]. The user input query is expanded by adding the highly related terms to extracted phrases with similarity scores greater than a threshold (e.g., 0.95) (Figure 6). With the extended query the search engine retrieve datasets relevant to the query, but do not contain phrases in the original query. For example, given a query "sea level pressure", the search engine cannot retrieve documents, including the acronym "SLP". If the relation between the two terms is uncovered from metadata and user, the query is rewritten to "sea level pressure" or "SLP" to improve the recall of the search engine.

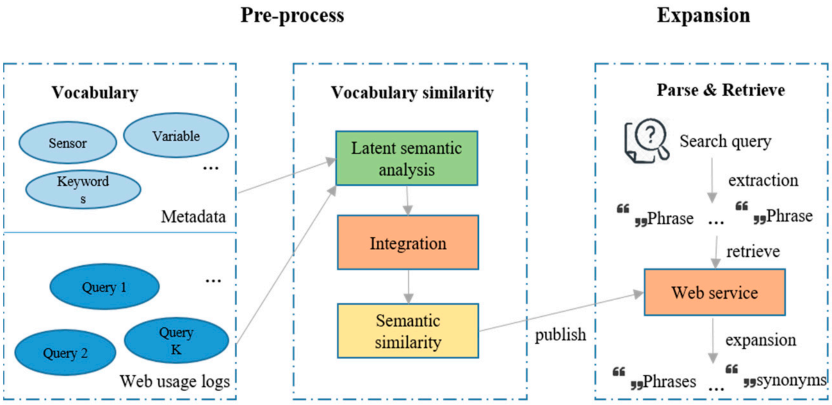


Figure 6. Workflow of query expansion.

5. System Design and Experiment

5.1. System Design

The system works in the following nine steps:

1. In pre-processing, the system collects raw Web usage logs and metadata.
2. Vocabulary similarity database is computed from query logs and metadata [30].
3. Bigram and trigram models are trained from description field in the metadata with Gensim.
4. A NER model is trained with Standford CoreNLP APIs for select entities including variable, term, processing level, version, collection, and sensor.
5. With these pre-trained models, the query understanding framework can parse corresponding information from a query.
6. After receiving a query, the framework extracts the spatial and temporal bounding box from the query calling Google Geocoding API and Dateparser function.
7. The corresponding spatial and temporal phrases are removed from the query and other parts of the query are passed to the bigram model, trigram model, and NER model for phrases and name entities recognition.
8. Detected phrases are expanded with highly related terms by calling the Restful query expansion API in the query expansion module.
9. The query understanding framework converts the original query to a rewritten query, which define the spatial-temporal context and real intent of a query (Figure 7)".

We intentionally made this system process independent from ocean science so the metadata, domain phrases, name entities, and synonyms can be easily changed for other domains for adaptability.

5.2. Experiment Setup

Before training models are called in the query understanding framework, metadata and logs were collected as the train data. Publicly available, collection-level PO.DAAC metadata are harvested from the PO.DAAC Web portal through its Web service API. Private portal user logs were provided by the data center as another source of knowledge. The collected metadata are in Directory Interchange Format (DIF) that was recommended by NASA Earth Science Data systems and contain descriptive information, such as organization, category, topic, term, sensor, platform, and description. User logs record users' behavior in the portal, such as searching a query, downloading a data during a time period.

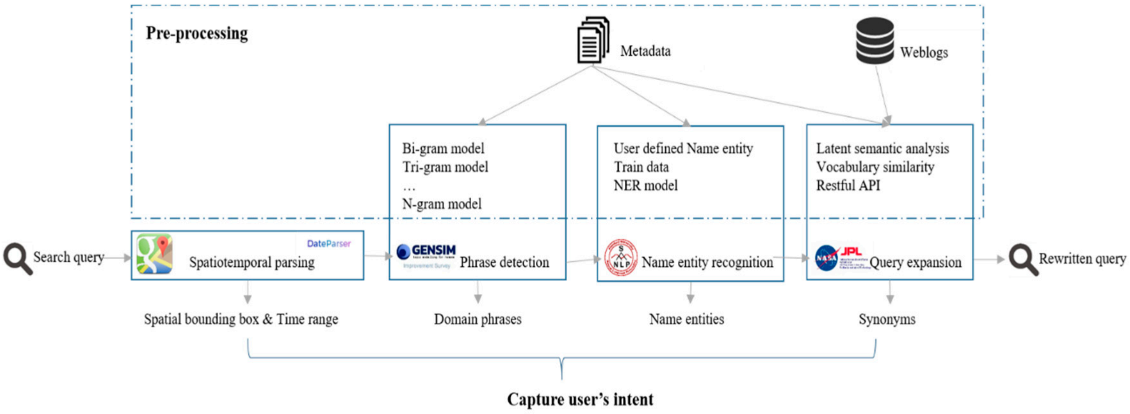


Figure 7. Workflow of query understanding.

In the concept recognition module, the values of "min_count" and "phrase_threshold" in Gensim toolkit are set to 5 and 10, respectively, to build bigram and trigrams models while using the content of metadata description as input. In the name entity recognition step, some metadata attributes were selected as the entity attribute sources and some were regarded as query attributes. Six PO.DAAC metadata attributes (i.e., "DatasetParameter-Variable", "DatasetParameter-Term", "Dataset-ProcessingLevel", "DatasetVersion-Version", "Collection-ShortName", and "DatasetSource-Sensor-ShortName") serve as entity attributes summarizing the unique phrases for the six user-defined name entities. Expressions and sentences in query attributes (i.e., "Dataset-LongName", "DatasetCitation-Title", and "Dataset-Description") are regarded as independent queries. Phrases in these independent queries are labeled with the six types of name entities. Expressions and sentences with labeled information are transformed to a certain format and then passed to the Stanford NLP software for training a NER model. For the query expansion module, a vocabulary similarity database was computed from metadata and usage logs and a RESTful Web service is published to derive highly-related terms for a given phrases. After these models were pre-trained, the query-understanding framework can aid query understanding in the PO.DAAC portal.

A set of synthetic queries are populated from PO.DAAC logs, in which user input query and corresponding filters that narrow the search are combined as a synthetic query, to evaluate the accuracy of the query understanding framework. For these synthetic queries generated from PO.DAAC query logs, NASA and NOAA scientists were invited to evaluate the parsing result of each step in the query understanding tool while using a four-point scale consisting of "Excellent", "Good", "Bad", and "Null":

- Null means the original query does not contain the corresponding fields (e.g., location, time terms).
- Bad indicates the tool fails to detect the corresponding phrase in a step at all (e.g., failing to recognize "sea surface temperature" as a phrase in the concept recognition step).
- Good is used to mark the understanding results that locate somewhere between "Excellent" and "Bad", and it represents the query understanding tool partially captures the search intent in one step (e.g., detecting both "sea surface temperature" and "group high resolution" as phrases or detecting "sea surface temperature" as a phrase, but failing to identify "ocean circulation" as phrases in the concept recognition step).

Excellent demonstrates that the tool successfully achieves its objective in one step (e.g., labeling "sea surface temperature" as variable in the name entity recognition step).

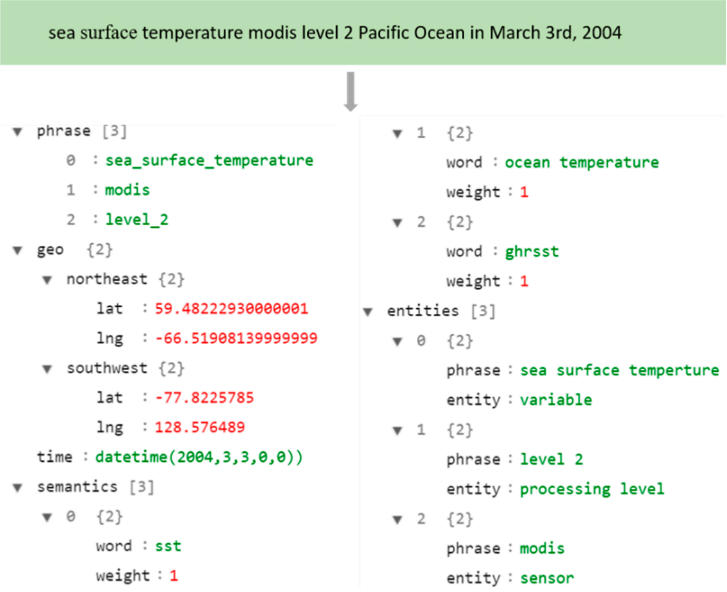
Qualitatively analyzing the results of a sample query and quantitatively calculating the overall accuracy of different modules in the framework for a collection of synthetic queries that were populated from logs are utilized to estimate accuracy evaluation.

6. Results and Discussion

A total of 389 unique user queries are extracted from logs, of which 169 contain either spatial coverage information or temporal search range, these synthetic queries are chosen for quantitative evaluation.

6.1. Qualitative Evaluation on a Sample Query

The parsing results of query "sea surface temperature modis level 2 Pacific Ocean in March 3rd, 2004" is achieved with the geo tab showing the latitude and longitude of the northeast and southwest points of the Pacific Ocean as a spatial bounding box to filter datasets that are located inside the spatial range (Figure 8). "3 March 2004" in the query is converted to "03/03/2004:00:00:00" for time comparison. Three phrases (i.e., "sea surface temperature", "modis", "level 2") are detected from the query, and similar phrases to these phrases are added to the query. Phrases in the original query are rewritten to "sea surface temperature" or "sst" or "ghrsst" or "ocean temperature" and "modis" and "level 2" to improve the search recall and precision. The tool also found that the query consists of three entities, in which "sea surface temperature" is a variable, "level 2" is a processing level, and "modis" is a sensor. The three entities are automatically applied to filters to narrow the search scope.



sea surface temperature modis level 2 Pacific Ocean in March 3rd, 2004

```
▼ phrase [3]
    0 : sea_surface_temperature
    1 : modis
    2 : level_2
▼ geo  {2}
  ▼ northeast {2}
      lat  : 59.48222930000001
      lng  : -66.51908139999999
  ▼ southwest {2}
      lat  : -77.8225785
      lng  : 128.576489
  time : datetime(2004,3,3,0,0))
▼ semantics [3]
  ▼ 0  {2}
      word : sst
      weight : 1
```

```
  ▼ 1  {2}
      word : ocean temperature
      weight : 1
  ▼ 2  {2}
      word : ghrsst
      weight : 1
▼ entities [3]
  ▼ 0  {2}
      phrase : sea surface temperture
      entity : variable
  ▼ 1  {2}
      phrase : level 2
      entity : processing level
  ▼ 2  {2}
      phrase : modis
      entity : sensor
```

Figure 8. Query understanding result of a sample query.

6.2. Quantitative Evaluation on a Set of Synthetic Queries

Table 1 shows part of the synthetic queries and the intermediate results of each step. The first column lists original queries. The next five columns show spatial bounding box, time range, phrase, entity, and expanded phrases in turn. Note that N/A in the table means the original query does not contain the corresponding field, and W/A indicates that the query has words or terms belonging to that field, but the tool does not detect them, which will be evaluated as "bad", since the information has not been successfully discovered. If the spatial bounding box and temporal range are correctly detected, their values will be converted to longitude, latitude, and time object. The concept recognition column shows the phrases that were extracted from the query. In the table, most phrases are accurately detected, such as "ocean waves" and "Pacific Ocean". Name entity recognition column displays all user-defined entities that were recognized from the query. In the name entity recognition column, the pre-trained NER model recognizes entities from most queries successfully. For example, "quikscat" and "ghrsst" were recognized as "collection" entity and "sea surface temperature" is detected as "variable" entity, but, in the last row, the model failed to detect "sea surface topography" as a "parameter" entity.

Query expansion column demonstrates phrases whose similarity to phrases extracted from original query are not smaller than a predefined threshold. The higher the threshold, the less noise and fewer phrases will be added into the expansion list. In our experiment, the threshold is set to 0.9 and these phrases with a similarity larger than 0.9 to extracted phrases in the query were added to the original query for query expansion.

Table 1. Query understanding results of five synthetic queries.

Understanding Result Synthetic Query	Spatial Boundary Box	Temporal Range	Concept Recognition	**Name Entity Recognition	***Query Expansion
quikscat 08/31/2014t19:30:00.000z to 09/20/2014t19:30:00.000z	*N/A	08/31/2014:19:30 to 09/20/2014:19:30	quikscat	Quikscat (collection)	sea wind (0.93)
ghrsst global	*W/A	N/A	ghrsst, global	Ghrsst (collection)	sea surface temperature (1.0), group high resolution sea surface temperature (1.0)
radar Pacific Ocean	59.4822, 66.5190, 77.8225, 128.5764	N/A	radar, Pacific Ocean	N/A	sigma naught (1.0), spectral engineering (0.9)
group high resolution sea surface temperature dataset global	W/A	N/A	group high resolution, sea surface temperature, global	sea surface temperature(variable)	sst (1.0), ocean temperature (1.0), ghrsst (1.0)
sea surface topography significant wave height netcdf ocean waves global	W/A	N/A	sea surface topography, significant wave height, ocean waves, global	W/A	sea surface height (1.0)

* N/A means the corresponding value is not provided in the original query; W/A means the value has not been successfully detected. ** In the name entity recognition column, terms in the parenthesis are the detected entity of corresponding phrases. *** In the last column, the numbers in parenthesis indicates the similar scores between the phrase to a concept in the synthetic query.

A stacked bar graph is used to divide the evaluation results and compare the parts to the whole. Each bar in the chart represents the whole evaluation data, and segments in the bar represent data falling in different categories (Figure 9). Removing the queries that lack the corresponding field for each metric (Figure 10) shows that the percentages of evaluation labeled as "good" and "excellent" are larger than 80%, except for spatial range, which indicates that the proposed query understanding tool performs well. Although more than 50% of spatial parsing result is marked as "bad", all errors are caused by the same issue (i.e., the spatial term "global" in the query cannot be parsed to coordinates by the Google Geocoding API). Similarly, all of the time range terms are converted to time objects, because all of them have the same formats as "2014-08-31t19:30:00.000z to 2014-09-20t19:30:00.000z". The phrases are sucessfully recognized in about 90% queries, since the bigram and trigram models detect most phrases in the corpus. Customized entities are accurately recognized from around 80% queries. More than 70% query expansion results are regarded as "good" or "excellent", and the quality of query expansion depends on the frequency a query occurs in the metadata and Web usage logs. In summary, the overall percentage of "excellent" and "good" for each metric indicates that the proposed query understanding framework captures the intent of a query.

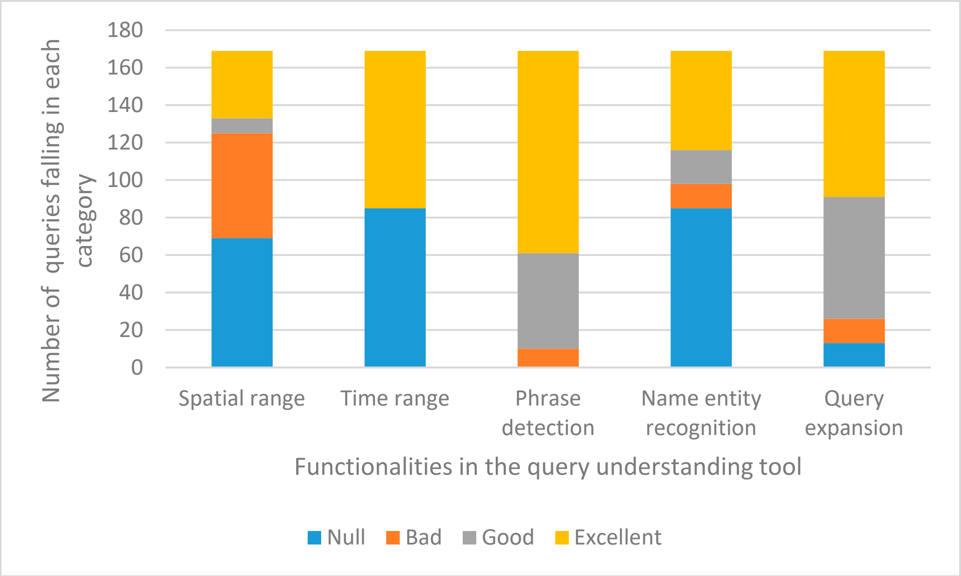


Figure 9. Evaluation of query understanding results of 169 synthetic queries.

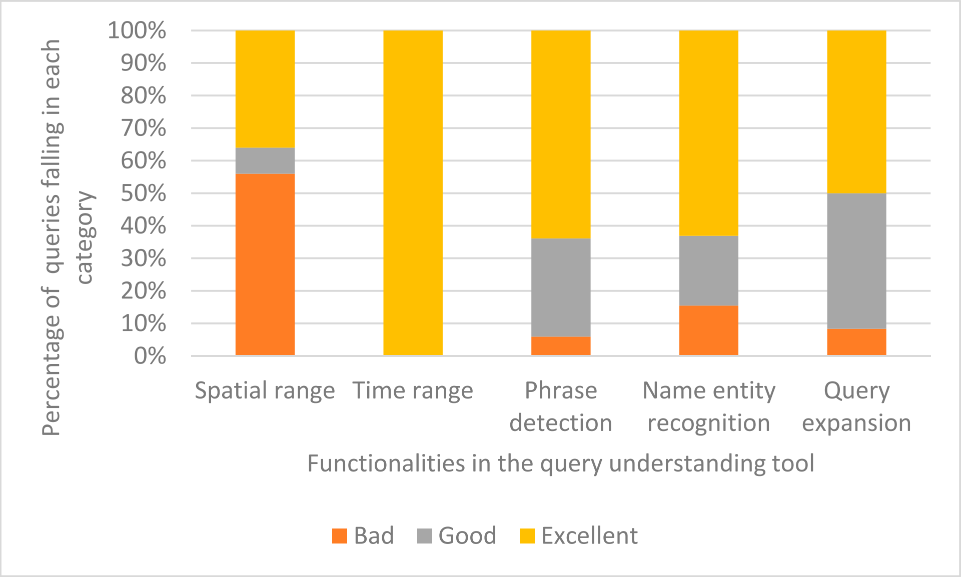


Figure 10. Evaluation of valid query understanding results of 169 synthetic queries.

7. Conclusion and Future Work

Query understanding focuses on the beginning of the search process and it is significant for capturing a user's search intent. This research reports our findings on a comprehensive query understanding solution to fill the intent gap in Earth data discovery and using oceanographic data query as an example.

- Qualitative and quantitative evaluation indicates a query understanding framework consisting of spatial and temporal parsing, concept recognition, name entity recognition, and query expansion can capture a user's intent.
- N-gram models trained from metadata can detect reasonable phrases from a query.
- Metadata attributes and descriptions serve as valuable sources to train a customized NER model for user-defined entities. The NER model can recognize the name entities from a query for filters correction or supplement.

- A vocabulary similarity knowledge base built from metadata and Web usage logs provide acronyms and synonyms to phrases detected from the original query.

Although we used ocean science data as example here, other domain data can be handled similarly with easily replacing of metadata, synonyms, and domain phrases, name entities in a flexible fashion, as noted in the architecture (Figure 6) section. Each component in the framework can be improved or replaced with better solutions, respectively, in the future research. One limitation of the proposed framework is that phrases containing more than three words are undetected. The state-of-the-art methods (e.g., CNN or convolutional neural network) model could solve this problem when enough train data are available, instead of simply increasing the number of n in the n-gram model [43]. Another limitation concerns the complexity of the user input query. The proposed tool concentrates on the phrases and entities in the query and it is unable to understand more complicated queries (e.g., "what is the temperature data at 700 m higher than the sea surface?"). In the current research, the spatial and temporal parsing module simply relies on open source APIs. In future research, more spatial and temporal parsing tools can be integrated to provide comprehensive search capability with customized configuration, e.g., converting ocean names to polygons for spatial matching. In addition, customized spatial and temporal parsing tools can be developed to improve the accuracy of the spatiotemporal parsing results. Future research will also integrate the query understanding tool into Big Earth data analytics frameworks [44,45] and Earth science data centers to make the search function of the existing systems operational.

Author Contributions: The contributions of the respective authors are as follows: Y.L. and Y.J. came up with the original idea; Y.L. and Y.J. designed and developed the study; Y.L., Y.J., J.C.G. and L.J.M. conducted the experiments; J.C.G., L.J.M., and C.Y performed the analysis; C.Y. supervised the research, revised the paper and obtained research funding; and Y.L. wrote a draft of the manuscript, which was iteratively improved by all other authors of the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by Earth Science Information Partners (ESIP) Lab as an incubator project and based on the NASA AIST MUDROD project.

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

References

1. Vatsavai, R.R. Spatiotemporal data mining in the era of big spatial data: Algorithms and applications. In Proceedings of the 1st ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data, Redondo Beach, CA, USA, 6 Nov 2012; ACM: Redondo Beach, CA, USA.
2. PO.DAAC. 2019 PO.DAAC Web Portal Search Help Page. Available online: https://podaac.jpl.nasa.gov/DatasetSearchHelp (accessed on 30 December 2019).
3. Ticheler, J.; Hielkema, J.U. Geonetwork opensource internationally standardized distributed spatial information management. *OSGeo J.* **2007**, *2*, 1–5.
4. McCandless, M.; Hatcher, E.; Gospodnetic, O. *Lucene in Action: Covers Apache Lucene 3.0*; Manning Publications Co: Shelter Island, NY, USA, 2010.
5. Liu, J. Query understanding enhanced by hierarchical parsing structures. In Proceedings of the 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, Czech Republic, 8–13 December 2013.
6. Demartini, G. CrowdQ: Crowdsourced Query Understanding. In Proceedings of the CIDR, Asilomar, CA, USA, 6–9 January 2013.
7. AlJadda, K. Crowdsourced query augmentation through semantic discovery of domain-specific jargon. In Proceedings of the 2014 IEEE International Conference on Big Data, Washington, DC, USA, 27–30 October 2014; IEEE: Piscataway, NJ, USA, 2014.
8. Yang, C.; Yu, M.; Hu, F.; Jiang, Y.; Li, Y. Utilizing Cloud Computing to Address Big Geospatial Data Challenges. *Comp. Environ. Urban Syst.* **2017**, *61*, 120–128. [CrossRef]
9. Yang, C.P.; Yu, M.; Xu, M.; Jiang, Y.; Qin, H.; Li, Y.; Bambacus, M.; Leung, R.Y.; Barbee, B.W.; Nuth, J.A. An architecture for mitigating near Earth object's impact to the earth. In Proceedings of the 2017 IEEE Aerospace Conference, Big Sky, MT, USA, 4–11 March 2017.

10. Jiang, Y.; Yang, C.; Xia, J.; Liu, K. Polar CI Portal: A Cloud-based Polar Resource Discovery Engine. In *Cloud Computing in Ocean and Atmospheric Sciences*; Academic Press: Cambridge, MA, USA, 2016; pp. 163–185.
11. Copernicus, E. Copernicus Open Access Hub. Available online: https://scihub.copernicus.eu/ (accessed on 6 February 2020).
12. Balk, D.; Yetman, G. *The Global Distribution of Population: Evaluating the Gains in Resolution Refinement*; Center for International Earth Science Information Network (CIESIN), Columbia University: New York, NY, USA, 2004.
13. GeoNetwork, F. Sub-National Administrative and Political Boundaries of Africa (2000). Available online: http://www.fao.org/geonetwork (accessed on 6 February 2020).
14. Gormley, C.; Tong, Z. *Elasticsearch: The Definitive Guide: A Distributed Real-Time Search and Analytics Engine*; O'Reilly Media, Inc.: Sevastopol, CA, USA, 2015.
15. Clifton, C.; Griffith, J.; Holland, R. GeoNode: An End-to-End System from Research Components. In Proceedings of the 17th International Conference on Data Engineering, Heidelberg, Germany, 2–6 April 2001.
16. Winn, J. Open Data and the Academy: An Evaluation of CKAN for Research Data Management. In Proceedings of the IASSIST, Cologne, Germany, 28–31 May 2013.
17. Jiang, Y.; Li, Y.; Yang, C.; Hu, F.; Armstrong, E.M.; Huang, T.; Moroni, D.; McGibbney, L.J.; Finch, C.J. Towards intelligent geospatial data discovery: A machine learning framework for search ranking. *Int. J. Dig. Earth* **2017**, *11*, 956–971. [CrossRef]
18. Hu, Y. A linked-data-driven and semantically-enabled journal portal for scientometrics. In Proceedings of the International Semantic Web Conference 2013, Sydney, Australia, 21–25 October 2013; Springer: Berlin/Heidelberg, Germany, 2013.
19. Jiang, Y.; Li, Y.; Yang, C.; Liu, K.; Armstrong, E.M.; Huang, T.; Moroni, D.F.; Finch, C.J. A comprehensive methodology for discovering semantic relationships among geospatial vocabularies using oceanographic data discovery as an example. *Int. J. Geogr. Inf. Sci.* **2017**, *31*, 2310–2328. [CrossRef]
20. Rose, D.E.; Levinson, D. Understanding user goals in web search. In Proceedings of the 13th International Conference on World Wide Web, New York, NY, USA, 17–20 May 2004; ACM: New York, NY, USA, 2004.
21. Wang, F. A fuzzy grammar and possibility theory-based natural language user interface for spatial queries. *Fuzzy Sets Syst.* **2000**, *113*, 147–159. [CrossRef]
22. Kadel, L.B.; Soni, D.K.; Yadav, R. Noun phrase detection and its challenges in large-scale natural language data processing. *Artif. Intell. Syst. Mach. Learn.* **2015**, *7*, 139–143.
23. Neelakantan, A.; Collins, M. Learning dictionaries for named entity recognition using minimal supervision. *arXiv* **2015**, arXiv:1504.06650.
24. Deliiska, B. Thesaurus and domain ontology of geoinformatics. *Trans. GIS* **2007**, *11*, 637–651. [CrossRef]
25. Neuendorf, K.; Mehl, J., Jr.; Jackson, J. *Glossary of Geology, (Revised)*; American Geosciences Institute: Alexandria, VA, USA, 2011.
26. Raskin, R.G.; Pan, M.J. Knowledge representation in the semantic web for Earth and environmental terminology (SWEET). *Comput. Geosci.* **2005**, *31*, 1119–1125. [CrossRef]
27. Whitehead, B.; Gahegan, M. Deep Semantics in the Geosciences: Semantic building blocks for a complete geoscience infrastructure. In Proceedings of the Eighth Australasian Ontology Workshop, Sydney, Australia, 4 December 2012.
28. Apache, Apache HTTP Server Version 2.4. Available online: http://httpd.apache.org/docs/current/logs.html (accessed on 1 January 2016).
29. Jiang, Y.; Li, Y.; Yang, C.; Liu, K.; Armstrong, E.M.; Huang, T.; Moroni, D. Reconstructing sessions from data discovery and access logs to build a semantic knowledge base for improving data discovery. *ISPRS Int. J. Geo Inf.* **2016**, *5*, 54. [CrossRef]
30. Li, Y.; Jiang, Y.; Gu, J.; Lu, M.; Yu, M.; Armstrong, E.M.; Huang, T.; Moroni, D.; McGibbney, L.J.; Frank, G.; et al. A Cloud-Based Framework for Large-Scale Log Mining through Apache Spark and Elasticsearch. *Applied Sciences 9.6 (2019): 1114.Devarakonda, R. Data sharing and retrieval using OAI-PMH. Earth Sci. Inf.* **2011**, *4*, 1–5.
31. Hu, Y. Metadata topic harmonization and semantic search for linked-data driven geoportals: A case study using ArcGIS Online. *Trans. GIS* **2015**, *19*, 398–416. [CrossRef]
32. Bernhard, S. *GEOCODE3: Stata Module to Retrieve Coordinates or Addresses from Google Geocoding API Version 3*, 2013.

33. DateParser. 2019 Dateparser—Python Parser for Human Readable Dates. Available online: https://dateparser.readthedocs.io/en/latest/ (accessed on 30 December 2019).
34. GeoNetwork. 2019 Portal Configuration. Available online: https://geonetwork-opensource.org/manuals/trunk/eng/users/administrator-guide/configuring-the-catalog/portal-configuration.html?highlight=search%20syntax (accessed on 30 December 2019).
35. Brown, P.F. Class-based n-gram models of natural language. *Comput. Linguist.* **1992**, *18*, 467–479.
36. Clarkson, P.; Rosenfeld, R. Statistical language modeling using the CMU-Cambridge toolkit. In Proceedings of the Fifth European Conference on Speech Communication and Technology, Rhodes, Greece, 22–25 September 1997.
37. Rehurek, R.; Sojka, P. *Gensim–Python Framework for Vector Space Modelling*; NLP Centre, Faculty of Informatics, Masaryk University: Brno, Czech Republic, 2011; Volume 3.
38. Guo, J. Named entity recognition in query. In Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Boston, MA, USA, 19–23 July 2009; ACM: New York, NY, USA, 2009.
39. Manning, C. The Stanford CoreNLP natural language processing toolkit. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Baltimore, MD, USA, 23–24 June 2014.
40. Dumais, S.T. Latent semantic analysis. *Annu. Rev. Inf. Sci. Technol.* **2004**, *38*, 188–230. [CrossRef]
41. Jiang, Y. A Smart Web-Based Geospatial Data Discovery System with Oceanographic Data as an Example. *ISPRS Int. J. Geo Inf.* **2018**, *7*, 62. [CrossRef]
42. Hashemi, H.B.; Asiaee, A.; Kraft, R. Query intent detection using convolutional neural networks. In Proceedings of the International Conference on Web Search and Data Mining, Workshop on Query Understanding, San Francisco, CA, USA, 22–25 February 2016.
43. Yang, C.; Yu, M.; Li, Y.; Hu, F.; Jiang, Y.; Liu, Q.; Sha, D.; Xu, M.; Gu, J. Big Earth data analytics: A survey. *Big Earth Data* **2019**, *3*, 83–107. [CrossRef]
44. Huang, T. An Integrated Data Analytics Platform. *Front. Mar. Sci.* **2019**, *6*, 354.
45. Yang, C.; Clarke, K.; Shekhar, S.; Tao, C.V. Big spatiotemporal data analytics: A research and innovation frontier. *Int. J. Geogr. Inf. Sci.* **2019**. [CrossRef]