Template-Based Question Answering over Linked Geospatial Data

Dharmen Punjani[a], Markos Iliaki[a], Theodoros Stefou[a], Kuldeep Singh[b], Andreas Both[c,d], Manolis Koubarakis[a], Iosif Angelidis[a], Konstantina Bereta[a], Themis Beris[a], Dimitris Bilidas[a], Theofilos Ioannidis[a], Nikolaos Karalis[a], Christoph Lange[e], Despina-Athanasia Pantazi[a], Christos Papaloukas[a], Georgios Stamoulis[a]

[a] *Department of Informatics and Telecommunications, National and Kapodistrian University of Athens {dpunjani, koubarak, iosang, konstantina.bereta, tberis, d.bilidas, tioannid, nkaralis, langec, dpantazi, christospap, gstam}@di.uoa.gr*
[b] *Cerence GmbH, Aachen, Germany, kuldeep.singh1@cerence.com*
[c] *DATEV eG, Germany, contact@andreasboth.de*
[d] *Anhalt University of Applied Science, Köthen, Germany, andreas.both@hs-anhalt.de*
[e] *Fraunhofer FIT, Sankt Augustin, Germany, lange@informatik.rwth-aachen.de*

Abstract

Large amounts of geospatial data have been made available recently on the linked open data cloud and the portals of many national cartographic agencies (e.g., OpenStreetMap data, administrative geographies of various countries, or land cover/land use data sets). These datasets use various geospatial vocabularies and can be queried using SPARQL or its OGC-standardized extension GeoSPARQL. In this paper, we go beyond these approaches to offer a question-answering engine for natural language questions on top of linked geospatial data sources. Our system has been implemented as re-usable components of the Frankenstein question answering architecture. We give a detailed description of the system's architecture, its underlying algorithms, and its evaluation using a set of 201 natural language questions. The set of questions is offered to the research community as a gold standard dataset for the comparative evaluation of future geospatial question answering engines.

Keywords: Linked Geospatial Data, General Administrative Divisions dataset (GADM), OpenStreetMap, Question Answering, GeoSPARQL, Information Retrieval, Semantic web

1. Introduction

The number of data sources in private environments, enterprises, and the Web is increasing continuously, increasing the effort of making data accessible. One important means of making data accessible is *question answering (QA)*, which provides a natural language interface for common users to express their information needs [1]. Users commonly pose questions or information requests with a *geospatial dimension* to search engines, e.g., "Christmas market in Germany", "Schools in London", "Is there a Macy's near Ohio?", "Which countries border Greece?". Answering such questions or information requests requires data that has a geospatial dimension as well.

Geospatial or *geographic knowledge* has been studied for many years by researchers in Geography, Geographic Information Systems (GIS), Geographic Information Retrieval (GIR), Databases, Artificial Intelligence and the Semantic Web, and there is a wealth of research results concerning representation, querying and inference for geographic knowledge. In GIS terminology which we use in this paper, a *geographic feature* (or simply *feature*) is an abstraction of a real world phenomenon and can have various attributes that describe its *thematic* and *spatial* characteristics. For example, the country Greece is a feature, its name and population are thematic attributes, while its location on Earth in terms of polar co-ordinates is a spatial attribute. Knowledge about the spatial attributes of a feature can be *quantitative* or *qualitative*. For example, the fact that the distance between Athens and Salonika is 502 km is quantitative knowledge, while the fact that river Evros crosses Bulgaria and Turkey and is at the border of Greece with Turkey is qualitative knowledge. Quantitative geographic knowledge is usually represented using *geometries* (e.g., points, lines and polygons on the Cartesian plane) while qualitative geographic knowledge is captured by *qualitative binary relations* between the geometries of features.

A significant fraction of the available data on the Web is geospatial, and this fraction is growing by 20 percent or more per year [2]. Qualitative Geospatial data can be expressed as a property of an entity or an explicit assertion. For example, in the RDF dataset DBpedia[1] extracted from Wikipedia, the resource `dbr:Berlin` has a data property `dbo:Country` with value `dbr:Germany` enabling the answering of questions such as "Cities in Germany" using DBpedia. Or a dataset like DBpedia can contain the fact `dbr:Berlin ogc:sfWithin dbr:Germany` and the question "Which cities are in Germany?" can again be answered using this dataset. Quantitative Geospatial data can be expressed by a property of an entity which has value a geometry (latitude/longitude pair or polygon). Then, the question "Which cities are within 100 km of Berlin?" can be answered by retrieving the geometry of the resource `dbr:Berlin` from an appropriate geospatial dataset, and then computing the distance of this geometry to the geometries of cities outside Berlin. In this paper, we focus on question answering from qualitative and quantitative geospatial knowledge made available on the Web as *linked open data.*

Examples of geospatial data published on the Web as linked open data include geospatial data from various countries (e.g., the United Kingdom[2] or The Netherlands[3]), OpenStreetMap data (published in RDF by project LinkedGeoData[4] [3] but also by our AI group at the National and Kapodistrian University of Athens[5]), and land cover/land use data sets (e.g., the European CORINE land cover dataset published in RDF by our group in the context of various European projects). Queries over such data can be asked using the linked data query language SPARQL and its geospatial extensions GeoSPARQL[6] and stSPARQL [4]. However, to better serve the needs of non-technical end users, it would be worthwhile to offer a natural language interface to linked geospatial data based on QA techniques. To the best of our knowledge, none of the QA systems utilizing linked data developed in recent years [5] deals with geospatial data. The work presented in this paper makes the first steps towards this direction.

In a similar spirit, the GIR community has been emphasizing the need to develop techniques for answering geographic questions expressed in natural language over text data since 2004.[7] The importance of the research issues studied in GIR can also be seen by the fact that interaction with geospatial search engines on mobile devices today is often done using spoken natural language

[1] `http://wiki.dbpedia.org/`
[2] `http://data.ordnancesurvey.co.uk/`
[3] `https://www.kadaster.nl/-/bag-linked-data`
[4] `http://linkedgeodata.org/About`
[5] `http://ai.di.uoa.gr/#datasets`
[6] `http://www.opengeospatial.org/standards/geosparql`
[7] `http://www.geo.uzh.ch/~rsp/gir18/`

(e.g., in Google Maps you can ask for directions to a place and these directions are then spoken to you). This is in agreement with the vision of multimodal spatial querying presented in [6]. Geographical knowledge is also very important in the new generation of personal assistants such as Amazon Alexa or Google Home.

Important assets of the GIR and QA research communities are the *gold standards* i.e., datasets and sets of questions that can be used to test the effectiveness of developed systems and perform detailed comparisons of them. In the area of QA over linked data, such gold standards have recently been provided by the QALD challenge[8]. To the best of our knowledge, no gold standard for geospatial question answering over linked data has been proposed so far by QALD or any other relevant research activity.

The contributions of this paper are the following. We have designed and implemented GeoQA, the *first* question answering system for linked geospatial data. GeoQA is implemented using reusable components as part of the component-oriented Qanary question answering methodology [7, 8] and its most recent implementation Frankestein [9].

We have also developed a gold standard for question answering over linked geospatial data which consists of two parts. The first part is a linked geospatial dataset built from DBpedia, the GADM database of global administrative areas[9] and OpenStreetMap (OSM)[10]. For the purposes of the gold standard, GADM and OSM have been restricted to the United Kingdom and Ireland. The second part of the gold standard consists of 201 geospatial questions that have been collected by student volunteers at the National and Kapodistrian University of Athens. The gold standard is used in a evaluation of the effectiveness of GeoQA and it is also made freely available to the research community for evaluating other future proposals.[11] In this way, we contribute to a long-term research agenda towards question answering systems with geospatial features.

A previous version of this paper has been presented in the 12th Workshop on Geographic Information Retrieval (GIR'18) [10]. The current version of the paper contains the following additional contributions:

- We added support to the question patterns of Categories 1 and 6, and count questions of Category 7 of the gold standard question set.
- We have performed a detailed study of existing named entity recognizers and disambiguators to select the one we have used in the GeoQA engine (see Section 5.2).
- We performed various optimizations and code improvements in the instance identifier, property identifier and query generator of our engine (see Section 5).
- As a result of the above, we have now achieved much higher precision, recall and f-measures compared to the previous version of the GeoQA engine presented in [10].

The rest of the paper is organized as follows. The next section presents related work. Section 3 presents the three datasets of the gold standard and the interlinking of GADM and OSM with DBpedia. Section 4 presents the gold standard questions. Section 5 presents our approach to building the query answering pipeline used by GeoQA. Section 6 presents an evaluation of GeoQA using the gold standard. Section 7 concludes the paper and discusses future work.

[8] `https://qald.sebastianwalter.org/`
[9] `http://www.gadm.org/`
[10] `https://www.openstreetmap.org`
[11] `http://geoqa.di.uoa.gr/`

2. Related Work

Since the first workshop in this field in 2004, question answering over textual data with geospatial information has been studied by Geographic Information Retrieval researchers. Relevant problems in this area include detecting place names (a special case of named entity recognition) and associated spatial natural language qualifiers in text and user queries, and disambiguating place names (a special case of named entity disambiguation). Two representative examples of systems where some of these issues have been studied are SPIRIT [11] and STEWARD [12].

An important evaluation initiative for geographic information retrieval from multilingual text has been GeoCLEF.[12] From the 2008 version of GeoCLEF, the GiKiP pilot is complementary to our paper since it concentrated on answering geospatial questions in three languages (Portuguese, English and German) from Wikipedia [13].

Query processing for linked geospatial data has been an active field of research recently culminating in the definition of the OGC standard GeoSPARQL, an extension of SPARQL with a vocabulary, datatypes and functions for expressing geospatial queries over linked data. There has also been substantial activity in the implementation of query processing systems such as Strabon [4] and Ontop-spatial [14], which both support GeoSPARQL. The work of the present paper goes beyond these query processors to offering question answering services over linked geospatial data, i.e. supporting queries expressed in natural language.

The work by Younis et al. [15] is most closely related to our work since it presents a system for answering geospatial questions over DBpedia. The system is based on a PostGIS[13] database containing precise geospatial information of features in the United Kingdom provided by Ordnance Survey, a spatial index of DBpedia resources built using their point coordinates, and a SPARQL endpoint storing the DBpedia dataset. The three classes of questions considered are proximity "Find churches within 1 km of the River Thames"), crossing (e.g., "Find the mouths of the rivers that cross Oxford") and containment (e.g., "Find churches in Manchester"). As we will see in Section 5, these kinds of questions are a subset of the ones that can be handled by GeoQA. Younis et al.[15] informally discusses the techniques that can be used to answer such questions, however, the system and dataset are not provided. Finally, the discussion in the paper pays some attention to the quality of returned answers.

Grutter et al. [16] explore the use of DBpedia and Geonames for answering topological queries involving administrative divisions of Switzerland and Scotland (since the authors are very familiar with the administrative geographies of these two countries). The paper contains a detailed discussion of quality issues in linked geospatial data and especially the two linked data sources used by the authors (e.g., incompleteness, inconsistency of data etc.). Finally, the paper considers queries for neighbouring and containing/contained administrative divisions, and measures precision and recall when only one of datasets or both linked datasets are used.

Hamzaie et al. [17] analyzes the natural language question-answer dataset MS MARCOV2.1 [18] which contains questions posed to the Bing search engine and human-generated answers. They concentrate on place-related questions of this dataset and define a set of patterns that can be used to characterize semantically questions and their answers. They also present a deeper understanding of the dataset using techniques based on word embeddings and clustering.

Tang and Mooney in [19] presents an inductive logic programming approach for learning a semantic parser and applies its techniques to two areas, one of which is querying geospatial

[12] `http://www.clef-initiative.eu/track/GeoCLEF`

[13] `http://postgis.net/`

databases. They have experimented with a dataset consisting of 1000 Prolog facts from the U.S. Geography domain, and have also developed a corpus of 880 natural language questions and their corresponding logical queries in Prolog.[14] A part of this corpus is used to train the semantic parser developed by the authors.

As it is already mentioned, in the area of QA there is currently no engine that deals with geospatial questions like GeoQA. From the existing systems, PowerAqua needs to be mentioned in our context since it also assumes that questions will be answered from many datasets or ontologies [20].

3. Constructing a Gold Standard Geospatial Data Set

In this section we discuss how to construct a gold standard geospatial dataset by interlinking DBpedia, OpenStreetMap and the GADM dataset of global administrative areas. Since DBpedia contains very limited geospatial information (e.g., latitude/longitude pairs, qualitative information via predicates such as `dbo:Country`), we enrich DBpedia with quantitative geospatial information (i.e., geometries) by interlinking it with OSM and GADM.

GADM is a dataset containing information about administrative divisions of various countries and their boundaries. GADM 3.4 (released on May 2018) contains information about 386,735 administrative areas. Particularly, the multi-polygon for each administrative area is provided along with a set of qualitative data, including its name and variant names. As the already existing linked data form of GADM [21] was based on very old version of GADM (and its representation was not based on the GeoSPARQL standard) we created a new linked data form of GADM from the available shapefiles using the tool GeoTriples[15]. We have the data from release version 2.8 (released on November 2015). For the purposes of this paper, we have only used GADM data from the United Kingdom (England, Scotland, Wales and Northern Ireland) and Ireland. The graphical representation[16] and the RDF/XML form[17] of the GADM ontology used are publicly available. Throughout the paper we use the prefix `gadmr:` instead of `http://www.app-lab.eu/gadm` for resources in the GADM data, and `gadmo:` for `http://www.app-lab.eu/gadm/ontology` for resources in the GADM ontology.

OSM is a collaborative project to create a free editable map of the world. It contains information about various features like rivers, lakes, cities, roads, points of interest (e.g., museums, restaurants and schools) etc. The geometries of these features can be points, lines or polygons. In addition to the geometry of a feature, OSM contains useful information such as name, feature class and layer. OSM data can be obtained in various formats. The first project to transform OSM data into RDF was LinkedGeoData [3]. Currently, this project does not provide an up-to-date version of OSM data that we could use for our study. For this reason, we had to repeat some of the work presented in [22] and, by doing this, go beyond [22] in the way that we will explain below. In the rest of the paper we use the prefix `osmr:` instead of `http://www.app-lab.eu/osm` for resources in the OSM data, and `osmo:` instead of `http://www.app-lab.eu/osm/ontology` for resources in the OSM ontology.

We obtained the OSM dataset in shapefile format from the company GEOFABRIK [18] and converted it into RDF using the tool GeoTriples. These shapefiles contain data available on

[14] http://www.cs.utexas.edu/users/ml/nldata/geoquery.html
[15] http://geotriples.di.uoa.gr
[16] http://geoqa.di.uoa.gr/images/gadm_ontology.png
[17] http://geoqa.di.uoa.gr/assets/GADM.owl
[18] http://download.geofabrik.de/europe.html

Table 1: Interlinking GADM with DBpedia

Country	Total entities	Linked automatically	Linked manually
UK	197	164	33
Ireland	27	17	10

date 30th August 2017. Like GADM, we have restricted our attention to the United Kingdom and Ireland. We designed a new ontology for OSM data which closely models the data in the shapefiles and made it publicly available in graphical format[19] and in RDF/XML format[20]. The ontology uses the GeoSPARQL vocabulary to model the geometries of various OSM features. Note that OSM does *not* have detailed administrative boundaries of various countries, hence we retrieve this information from GADM.

DBpedia is one of the most popular knowledge graphs derived from Wikipedia and its ontology which we use in the paper is publicly available[21]. Throughout the paper we use the prefix dbo: instead of `http:dbpedia.org/ontology` for resources in the DBpedia ontology, and dbr: instead of `http://dbpedia.org/resource` for resources in the DBpedia knowledge graph. Interlinking of GADM and OSM with DBpedia allows us to answer geospatial questions that cannot be answered by any of the datasets in isolation. For example, the question"Which of the English counties that border Greater Manchester has the highest percentage of ethnic Asians?" can only be answered by consulting GADM to find the counties that border Greater Manchester, and then DBpedia to find the percentage of various ethnic groups in these counties. Also, the question "Which Greek politicians are graduates of a university located in a Greek island belonging to the region of Western Greece?" can be answered only by consulting all three datasets.

3.1. Interlinking GADM with DBpedia

The interlinking of GADM with DBpedia was done as follows. For each administrative area mentioned in GADM we obtain the DBpedia resource which has the same label, by using DBpedia SPARQL endpoint. The few remaining GADM resources were mapped manually. This procedure resulted in most of the GADM resources being linked. The remaining ones were linked manually. Table 1 shows the relevant numbers.

3.2. Interlinking of OSM with DBpedia

The task of interlinking OSM with DBpedia had some interesting challenges. First of all, we manually identified classes that have the same or very similar label in DBpedia and OSM. These classes are: `Airport, Bank, Beach, Building, Canal, Castle, Cemetery, Church, City, College, Dam, Forest, Fort, Glacier, Golfcourse, Hospital, Hotel, Island, Library, Lighthouse, Locality, Memorial, Mine, Monument, Mosque, Museum, Park, Place, Prison, RailwayStation, Region, Restaurant, River, Road, School, Stadium, Stream, Temple, Theatre, Tower, Town, Tram, University, Village, Volcano, Watertower, Watermill, Windmill,` and `Zoo`. Then, interlinking was performed on a class-by class basis using the tool Silk [23]. The OSM

[19] `http://sites.pyravlos.di.uoa.gr/dragonOSM.svg`
[20] `http://pyravlos-vm5.di.uoa.gr/osm.owl`
[21] `http://mappings.dbpedia.org/server/ontology/classes/`

data is stored in a Strabon endpoint and the online DBpedia endpoint is used for the DBpedia data. The labels of the entities and the spatial distance of their geometries were considered equally for matching. In other words, we use the formula $(S(x, y) + MinDist(x, y))/2 = 1.0$ where

- x and y are the instances considered for matching in OSM and DBpedia respectively.
- $S(x, y)$ is the Levenshtein string similarity measure between the labels of x and y. The threshold considered for string similarity is 85%.
- $MinDist(x, y)$ is the minimum Euclidean distance between the geometries of x and y. After experimenting with different number of threshold values for Euclidean distance, we finalized the threshold to 1 kilometer.

Table 2 gives the number of instances of the various classes in both datasets, as well as the number of instances that were interlinked. The DBpedia instances have been selected by retrieving only the resources that have coordinates falling inside the minimum bounding rectangles of the geometries of the United Kingdom and Ireland. As it is expected, some classes have many more instances in one of the datasets. For example, the class `Restaurant` has 24055 instances in the subset of OSM that we consider and only 152 instances in DBpedia. Also, some of the classes having the same label are at different places in the class hierarchies of the two datasets. For example, the class `Building` is the parent class of `Restaurant, Hotel, Hospital, Museum` etc. in the DBpedia ontology, while it does not have any subclasses in the hierarchy of the OSM ontology, so we interlink instances of the subclasses of `Building` with the instances of corresponding classes of OSM. Similarly, `Road` has subclasses that we consider in OSM ontology, while it does not have any subclasses in DBpedia. Naturally, when a class had zero instances in one or both datasets (e.g., `Beach` in the DBpedia subset we consider and `Glacier` in both datasets) then the class does not participate in the interlinking and does not appear in Table 2. Finally, we would like to mention that we found many misclassified instances in DBpedia in contrast to the other two datasets; this has also been pointed out in [22].

Let us now comment on some rows of Table 2 where there is an unexpectedly big difference in the number of instances in OSM and DBpedia for the same class. Let us take for example the class `Airport`. Unfortunately, the freely available OSM shapefiles for the United Kingdom and Ireland, provided by GEOFABRIK [22], contain only 7 airports (not even Heathrow airport of London is included!). On the contrary, DBpedia has a rather large number resources classified as airports. In some cases, these are wrongly classified e.g., `dbr:Brahan_Castle`, a castle, is wrongly classified as `dbo:Airport`. It is also interesting to consider the row for class `River`. There are many more instances of `River` in OSM than in DBpedia because OSM has a different entry for each of the segments/polygons making up a river in its full length. The same issue exists for the classes `Canal` and `Stream`. This is the reason that the number of total interlinked instances is bigger than the cardinality of the intersection of the two datasets for classes like `River`. Finally, another reason for the difference in instances between the same classes in DBpedia and OSM is the nature of the domain of interest in the datasets. For example, DBpedia has information about only 1339 hotels in its full dataset of which 212 hotels are in the United Kingdom and Ireland. The corresponding number of hotels in OSM is 9819 hotels as we see from

[22] `http://download.geofabrik.de/europe.html`

the Table 2. In a similar way, the class `Restaurant` in DBpedia has few instances compared to OSM.

After completing the interlinking with Silk, there were some entities that were not linked. These were checked and linked, if appropriate, manually. For matches below 100 all matching pairs were checked manually for correctness. For larger numbers of matching pairs, we checked manually 100 random pairs and found them all to be correct. So, we conclude that our matching process is very accurate.

Comparing the interlinking of OSM and DBpedia that we have done with the interlinking done in LinkedGeoData [22], we can see that we have interlinked instances belonging to many more classes. The OSM dataset in the case of LinkedGeoData is stored using Virtuoso which has support only for point geometries. Therefore, no queries involving complex geometries can be done, and the interlinked resources in the case of LinkedGeoData are limited to OSM nodes.

The GADM and OSM datasets as well as the interlinking dataset is publicly available on the Web site of the gold standard.[23] We will call this data part of the gold standard *GeoData201*.

4. Creating a gold standard set of geospatial questions

To be able to evaluate the effectiveness of our query engine and compare it with other QA engines available, we have created a new benchmark set of 201 questions which we have collectively called *GeoQuestions201*. The questions have been written by third-year students of the 2017-2018 Artificial Intelligence course in our department. The students were asked to target the above three data sources by imagining scenarios where geospatial information will be needed and could be provided by an intelligent assistant, and to propose questions with a geospatial dimension that they considered "simple" (a few examples of such questions were provided). The authors of the paper have then "cleaned" the given set of questions and produced the SPARQL or GeoSPARQL queries that correspond to them assuming ontologies that describe the three data sources using the GeoSPARQL vocabulary. The complete set of resources (data sources, ontologies, natural language questions and SPARQL/GeoSPARQL queries) is available on the Web at `http://geoqa.di.uoa.gr`.

The questions in the benchmark GeoQuestions201 fall under the following categories. For each one of the categories, we also comment on whether two major search engines (Google and Bing) can answer questions in this category.

1. *Asking for the attribute of a feature*. For example, "Where is Loch Goil located?" or "What is length of River Thames?". In GeoQA, these questions can be answered by posing a SPARQL query to DBpedia. Google and Bing both can also answer such questions precisely.
2. *Asking whether a feature is in a geospatial relation with another feature*. For example, "Is Liverpool east of Ireland?". The geospatial relation in this example question is a cardinal direction one (east of). Other geospatial relations in the set of questions include topological (borders) or distance (near or "at most 2km from"). In GeoQA, these questions are answered most of the time by using GADM and OpenStreetMap because the relevant qualitative geospatial knowledge is not present in DBpedia and/or the detailed geometries of features are needed for evaluating the geospatial relation of the question. Google and

[23] `http://geoqa.di.uoa.gr`

Table 2: Interlinking OSM with DBpedia

Class	**No. of Instances in OSM**	**No. of Instances in DBpedia**	**Interlinked Instances**	**Interlinked Instances (semi automatically)**	**Total Interlinked Instances**
Airport	7	815	1	5	6
Bank	7621	29	1	2	3
Canal	7902	167	2171	920	3091
Castle	1357	486	161	36	197
City	86	101	45	18	63
College	1529	38	0	2	2
Dam	330	26	1	3	4
Hospital	2352	537	244	149	393
Hotel	9819	212	73	81	154
Island	2477	750	219	138	357
Library	3635	119	47	25	72
Lighthouse	423	39	9	14	23
Monument	2108	38	5	3	8
Museum	2313	933	327	219	546
Park	54830	382	252	103	355
Prison	207	199	28	119	137
Railway Station	3932	45	0	0	0
Region	13	151	0	0	0
Restaurant	24058	152	31	30	61
River	52897	785	4342	237	4579
School	33217	5556	2683	691	3374
Stadium	799	687	120	78	198
Stream	240293	470	885	265	1150
Theatre	1224	86	19	33	52
Tower	2373	35	0	0	0
Town	1960	1066	132	18	150
University	2466	1099	169	41	210
Village	15743	15346	4308	4087	8395

Bing both cannot answer such factoid questions; both can only return a list of relevant Web pages.

3. *Asking for features of a given class that are in a geospatial relation with another feature*. For example, "Which counties border county Lincolnshire?" or "Which hotels in Belfast are at most 2km from George Best Belfast City Airport?". The geospatial relation in the first example question is a topological one (border). As in the previous category, other geospatial relations in the set of questions include cardinal (e.g., southeast of) or distance (near or "at most 2km from" as in the second example question). In GeoQA, these questions can be answered by using not just DBpedia but also GADM and OpenStreetMap when the detailed geometries of features are needed for evaluating the geospatial relations. Google can also answer such questions precisely in many but not all cases (e.g., it can answer precisely the first and third questions but not the second). Bing cannot answer such questions precisely but gives list of relevant web pages.
   Questions in this category might also have a second geospatial relation and a third feature which are used to further constrain the second feature. For example, "Which restaurants are near Big Ben in London?" or "Which rivers cross London in Ontario?". In the first question, we have also provided some more information about Big Ben although this might not have been necessary.[24] In the second question, "in Ontario" is used to make clear that we are referring to the city London in Ontario, Canada not the more well-known city of London in England.[25]
4. *Asking for features of a given class that are in a geospatial relation with any features of another class.* For example, "Which churches are near castles?". Arguably, this category of questions might not be useful unless one specifies a geographical area of interest; this is done by the next category of questions.
5. *Asking for features of a given class that are in a geospatial relation with an unspecified feature of another class which, in turn, is in another geospatial relation with a feature specified explicitly.* An example of such a question is "Which churches are near a castle in Scotland?". Google and Bing both cannot answer such questions precisely.
6. *The questions in this category are like the ones in Categories 3 to 5 above, but in addition, the thematic and/or geospatial characteristics of the features that are expected as answers (i.e., the features of the first class mentioned in the question) satisfy some further condition (e.g., numeric).* For example, "Which mountains in Scotland have height more than 1000 meters?" or "Which villages in Scotland have a population of less than 500 people?" or "Is there a church in the county of Greater Manchester dedicated to St. Patrick?" or "Which Greek restaurants in London are near Wembley stadium?". In these examples, the extra attribute conditions may require GeoQA to consult all three data sources to find the answer to a question. Google can answer precisely the first, third and fourth example question, but not the second, since its knowledge graph does not contain population information for villages in Scotland. Bing cannot answer any questions precisely but returns relevant links to points of interest.
7. *Questions with quantities and aggregates.* For example, "Which is the highest mountain in Ireland?" or "Which hotel is the nearest to Old Trafford Stadium in Manchester?" or

[24]The authors of this paper are not aware of another Big Ben.

[25]Boringly enough, London, Ontario is also crossed by a Thames river. We bet this is not how this river was called by native Indians in 1534 when Canada was discovered.

"Which is the largest lake by area in Great Britain?" Questions with quantities but without aggregates have recently been studied by [24] in a non-geospatial setting. Interestingly, Google and Bing both can answer all three example questions precisely. Note that questions in this class might also exhibit features of the previous two classes e.g., when a topological relation is involved or when the condition on an attribute refers to a quantity (e.g., height of a mountain). Such questions cannot be handled by QA engines as well as Google and Bing at the moment. For example, the question "Which is the largest county of England by population which borders Lincolnshire?" is answered incorrectly by Google (county Bristol is given as the answer) as well as by Bing.

The list of benchmark questions is available publicly[26].

5. Creating a Geospatial Question Answering Pipeline

We now present our approach to translating a natural language question into a GeoSPARQL query that can be executed on the union of the datasets presented in the previous section. For this, we build a geospatial question answering system using Qanary [25] and Frankenstein [9].

5.1. The Frankenstein Framework for Building QA Systems

Qanary is a lightweight component-based QA methodology for the rapid engineering of QA pipelines [8, 26]. Frankenstein [9] is the most recent implementation of the ideas of Qanary; this makes it an excellent framework for developing reusable QA components and integrating them in QA pipelines. Frankenstein is built using the Qanary methodology developed by Both et al. [8] and uses standard RDF technology to wrap and integrate existing standalone implementations of state-of-the-art components that can be useful in a QA system. The Qanary methodology is driven by the knowledge available for describing the input question and related concepts during the QA process. Frankenstein uses an extensible and flexible vocabulary [7] for data exchange between the different QA components. This vocabulary establishes an abstraction layer for the communication of QA components. While integrating components using Frankenstein, all the knowledge associated with a question and the QA process is stored in a process-independent knowledge base using the vocabulary. Each component is implemented as an independent microservice implementing the same RESTful interface. During the start-up phase of a QA pipeline, a service registry is automatically called by all components. As all components are following the same service interface and are registered to a central mediator, they can be easily activated and combined by developers to create different QA systems.

5.2. GeoQA: A Geospatial QA System

In our work, we leverage the power of the Frankenstein framework to create QA components which collectively implement the geospatial QA pipeline of GeoQA. The QA process of GeoQA uses the following modules implemented as components in the Frankenstein framework: dependency parse tree generator, concept identifier, instance identifier, geospatial relation identifier, property identifier, SPARQL/GeoSPARQL query generator and SPARQL/GeoSPARQL query executor. Our components are fully integrated in the Frankenstein ecosystem and can be reused to implement geospatial features in other QA systems, as our implementation is not monolithic

[26] `http://geoqa.di.uoa.gr/benchmarkquestions.html`

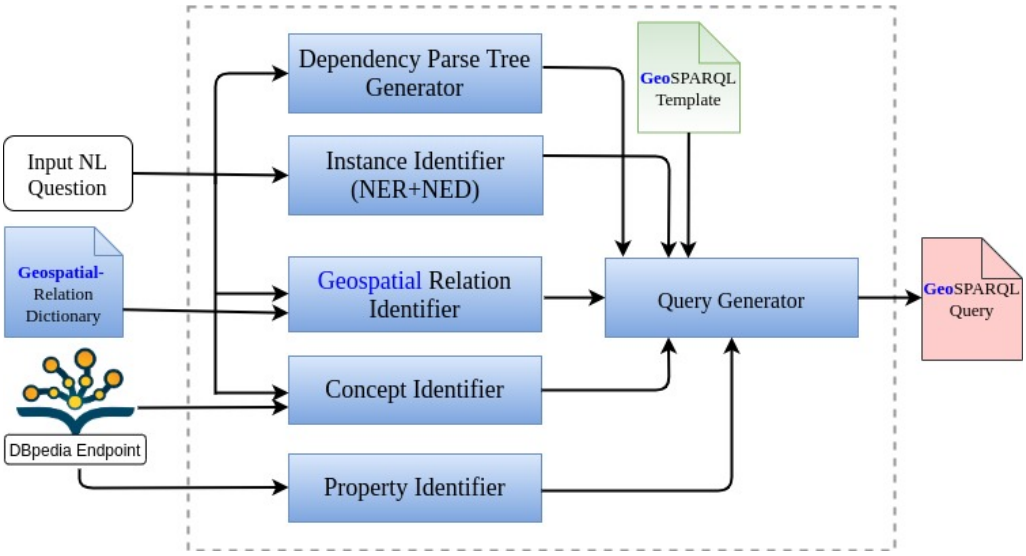

Figure 1: The conceptual architecture of the GeoQA system

like the implementation of many other QA systems [27, 28, 29]. GeoQA takes as input a question in natural language (currently only English is supported) and the three linked geospatial datasets presented in Section 3, and produces one or more answers that are resources of the given datasets. Question answering is performed by translating the input question to a set of SPARQL or GeoSPARQL queries, ranking these queries, and executing the top ranked query over two endpoints using the SPARQL `SERVICE` keyword. For DBpedia, we use its public Virtuoso endpoint[27] while for GADM, OSM and their interlinking dataset we use a Strabon endpoint. In Figure 1, we present the conceptual view of the implemented GeoQA system and Figure 2 presents how our system is actually implemented. The various components of GeoQA are discussed below.

Dependency parse tree generator. This component carries out part-of-speech tagging and generates a dependency parse tree for the input question using the Stanford CoreNLP software. The dependency parse tree is produced in CoNLL-U format [30].

Concept identifier. The concept identifier module identifies the *types of features* specified by the user in the input question and maps them to the corresponding classes in the DBpedia, GADM and OSM ontologies. We use the equivalent ontology-oriented term *concept* for a feature type in this paper. For example, if the input question is “Which restaurants are near Big Ben in London?”, then the term “restaurants” is identified as a feature type and mapped to the class `osmo:Restaurant` in the OSM ontology and `dbo:Restaurant` in the DBpedia ontology. The matching classes are found using string matching on the labels of the classes (the Java library function `java.util.regex.Pattern.matcher()`is used)together with lemmatization from Stanford CoreNLP and synonyms from Wordnet. In its final stage, the concept identifier annotates the appropriate node of the dependency parse tree with its results.

[27] `http://dbpedia.org/sparql`

Table 3: Accuracy of various named entity recognizer and disambiguator over Geoquestion201

NER+NED tool	**Accuracy over GeoQuestions201(%)**	**Disambiguate to**
StanfordNER [32] + AIDA [33]	80.0	Wikipedia
DBpedia Spotlight [34]	79.22	DBpedia
StanfordNER [32] + AGDISTIS [35]	81.50	DBpedia
TagMe [31]	**89.5**	Wikipedia
MeaningCloud[29]	67.0	Wikipedia
TextRazor[30]	76.5	Wikipedia
Babelfly [36]	40.5	DBpedia
Entity-fishing[31]	84.5	Wikipedia

Instance identifier. The next useful information to be identified in an input question is the *features* mentioned. These can be e.g., the country Ireland or the city Dublin or the river Shannon etc. We use the equivalent ontology-oriented term *instance(s)* for features in this paper. Once instances are identified, they are mapped to DBpedia, OSM or GADM resources using the entity recognition and disambiguation tool TagMeDisambiguate [31][28].

Let us explain why we selected TagMeDisambiguate. We have considered the tools that have been used in [9] for entity recognition and disambiguation. We have tested these tools over *GeoQuestions201*. It is to keep in mind that these tools may or may not be targeted for short text and text documents. We have used web services of these tools to annotate the *GeoQuestions201*. Table 3 shows information on these tools with how they perform over *GeoQuestion201*. Column accuracy in Table 3 is percentage of questions from the 201 we have that are correctly annotated with entities and their instances from DBpedia, Wikipedia or Wikidata. The annotations from each tool were verified manually. As can be seen in Table 3, TagMeDisambiguate annotates most of the questions correctly compared to other tools.

We also search for resources in the OSM and GADM dataset that have the same label as the entity identified by the TagMeDisambiguate component, and add them to the list of identified instances. For illustration, consider the input question “Which airports are in London?”. The term “London” is the identified instance (feature) and it is disambiguated to the wikipedia link and we get DBpedia resource `dbr:London` by `owl:sameAs` link from DBpedia Virtuoso endpoint[32], and to `osmr:england/places/id/107775` and `gadmr:administrativeUnit_GBR_adm2_56` by our code. In its final stage, the instance identifier annotates the appropriate node of the dependency parse tree with its results.

Geospatial relation identifier. Geospatial questions such as the ones targeted by GeoQA almost always include a qualitative geospatial relation such as “borders” or a quantitative one such as “at most 2km from”. The current implementation supports the 14 geospatial relations shown on Table 4. These include some topological, some distance and some cardinal direction relations [37, 38, 39]. Table 5 gives a dictionary of the various synonyms for these relations that can appear instead of them in a question. The semantics of topological relations are as in the dimensionally extended 9-intersection model [40]. Qualitative spatial relations of proximity like “close

[28] `https://tagme.d4science.org/tagme/`
[32] `http://dbpedia.org/sparql`

Table 4: Geospatial relation categories and relations

Category	**Geospatial relation**
Topological relations	“within”, “crosses”, “borders”
Distance relations	“near”, “at most *x* units”, “at least *x* units”
Cardinal direction relations	“north of”, “south of”, “east of”, “west of”, “northwest of”, “northeast of”, “southwest of”, and “southeast of”

to", “near" etc. are translated into (rather arbitrary) quantitative distance relations based on the concept identified earlier by the concept identifier (e.g., when asking for “hotels near a place", “near” is taken to mean at most 1 kilometer). The semantics of cardinal direction relations are the usual ones i.e., a relation *A* north of *B* is given meaning by considering the bounding box of the reference region *B* and the partition of the plane in nine areas that is induced by it [39]. The same semantics are implemented by the Strabon system and its query language stSPARQL which is used as our back end geospatial RDF store [4]. GeoSPARQL does not support any cardinal direction functions or relations. Finally, Kreveld and Reinbacher [41] provide a more intuitive semantics of cardinal directions for polygons, but an implementation of this semantics is more expensive computationally than the semantics used in Strabon [4].

Like the previous modules, this module first identifies geospatial relations in the input question, and then maps them to a spatial function of the GeoSPARQL or stSPARQL vocabulary, or a data property with a spatial semantics in the DBpedia ontology. As we have already discussed in the introduction, DBpedia contains limited explicit or implicit geospatial knowledge using latitude/longitude pairs, and properties such as `dbp:northeast` for cardinal direction relations or class-specific properties such as `dbo:city` (e.g., for class `dbr:River`). GeoQA does not make use of quantitative geospatial information (i.e., latitude/longitude pairs) from DBpedia since we have more detailed geospatial knowledge in the form of polygons in the datasets GADM and OSM. However, it does makes use of qualitative geospatial knowledge from DBpedia expressed using the data properties just mentioned (although this knowledge is rather scarce as discussed in [42]). As an example, for the question “Which counties border Lincolnshire?”, the geospatial relation “borders” is identified from the verbs in the dependency tree, and it is mapped to the spatial function `geof:sfTouches` of the GeoSPARQL vocabulary.

In its final stage, the geospatial relation identifier annotates the appropriate node of the dependency parse tree with its results. In the near future, GeoQA will cover all the prototypical spatial relations shown experimentally to correspond to natural language utterances by Egenhofer, Mark and their colleagues in [43, 44, 45].

Property Identifier. The property identifier module identifies *attributes of types of features* and *attributes of features* specified by the user in input questions and maps them to corresponding properties in DBpedia. To answer questions like “Which rivers in Scotland have more than 100 km length?” or “Which mountains in Scotland have height more than 1000 meters?”, we need information about length of rivers and height of the mountains in addition to their geometry from OSM. This information is not present in OSM but we can retrieve this information from DBpedia. We use Table 6 and 7 for this task. The identified concept from the concept identifier module is used to search Table 6 to get `dbp:height` and `dbp:length` in the case of example questions mentioned before. In case of question “Which Greek restaurants in London are near Wembley stadium?”, it is to be inferred that Greek should be a cuisine in context of restaurants

Table 5: Geospatial relations and their synonyms

Geospatial relation	Synonyms in dictionary
within	in, inside, is located in, is included in
crosses	cross, intersect
near	nearby, close to, around
borders	is/are at the border of, is/are at the outskirts of, at the boundary of
north of	above of
south of	below
east of	to the right
west of	to the left

Table 6: DBpedia Properties

DBpedia Class	DBpedia Property	Label of property
Mountain	http://dbpedia.org/property/height	Height
Mountain	http://dbpedia.org/property/elevation	elevation
Mountain	http://dbpedia.org/property/parentPeak	Parent peak
River	http://dbpedia.org/property/length	length
River	http://dbpedia.org/property/name	name
River	http://dbpedia.org/property/dischargeLocation	discharge location
River	http://dbpedia.org/property/mouth	Mouth

and we need to check all the possible values of properties for the identified concept. We achieve this with the use of Table 7. We stress that Tables 6 and 7 contain only examples of classes, properties and values that are of interest to the example questions. In reality the tables contain 11,392 and 2,61,455 entries respectively and cover all the DBpedia classes of Table 2. These tables have been generated querying DBpedia and stored in different files with their class names. In similar manner for question like "What is the total area of Northern Ireland?" we query DBpedia endpoint to retrieve property `dbp:areaKm` that is present in DBpedia for identified instance `dbr:Northern_Ireland`. We use string similarity measures while searching Table 6 and pattern matching while searching Table 7. In its final stage, the property identifier annotates the appropriate node of the dependency parse tree with its results.

Table 7: DBpedia Properties and Values

DBpedia Class	DBpedia Property	Value of property
Restaurant	http://dbpedia.org/ontology/cuisine	Asian Cuisine
Restaurant	http://dbpedia.org/ontology/cuisine	Italian,pizzeria
Restaurant	http://dbpedia.org/ontology/cuisine	Italian, Greek, French, Spanish, and Creole table delicacies

Listing 1: SPARQL/GeoSPARQL Query for Motivating Example 1

```
Question: Which rivers cross Limerick?
SPARQL:
select ?x
where {
       ?x rdf:type dbo:River.
       ?x dbo:city dbr:Limerick.
}

GeoSPARQL:
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX geof: <http://www.opengis.net/def/function/geosparql/>
PREFIX osmo: <http://www.app-lab.eu/osm/ontology#>

select ?x
where {
       ?x rdf:type osmo:River;
          geo:hasGeometry ?xGeom.
       ?xGeom geo:asWKT ?xWKT.

            gadmr:Limerick geo:hasGeometry ?iGeom.
       ?iGeom geo:asWKT ?iWKT.

       FILTER(geof:sfCrosses(?xWKT, ?iWKT))
}
```

Query generator. This module creates a SPARQL or a GeoSPARQL query using handcrafted query templates. From gathering questions from Google Trends and also studying the questions in our gold standard, we have identified the question patterns shown in Tables 8 and 9. In these tables C stands for "concept", I for "instance" , R for "geospatial relation" , P for "property" and N for "Count of" following the terminology we have introduced above. For each pattern, the tables give an example question and the corresponding GeoSPARQL and/or SPARQL query template. The query templates contain slots (strings starting with an underscore) that can only be identified when an example question is encountered and will be completed by the query generator (see below).

For each input question, the slots in the template are replaced by the query generator with the output of the previous modules, to generate a SPARQL or a GeoSPARQL query. For example, for the question "What is length of the river that crosses Limerick?", the identified pattern is PCRI. The question pattern is identified by searching the dependency parse tree in which the nodes have been annotated with the results of the concept, instance, property and geospatial relation identifier modules presented above. We walk through the parse tree with inorder traversal and identify the question pattern. If the question does not follow any of the patterns, a message is passed to the next component that no query has been generated. The appropriate templates are selected from Table 8 and 9, their slots are filled with the resources identified earlier and the corresponding GeoSPARQL or SPARQL queries are generated. Here the concepts are `dbr:River` from DBpedia and `osmo:River` from OSM, the property is `dbp:length` from DBpedia, the instances are `gadmr:Limerick` from GADM and `osmo:irelandandnorthernireland/places/id/2518952` from OSM, and the geospatial relations are `dbo:city` from the DBpedia ontology and the GeoSPARQL function `geof:sfCrosses`.

Table 8: Supported question patterns with examples and corresponding SPARQL/GeoSPARQL query templates for categories 1-5

Pattern	Example natural language question	Templates

IP	Where is Emirates Stadium located?	**SPARQL**:
		select ?x where {
		_Instance _Property ?x.
		}

CRI	Which rivers cross Limerick?	**SPARQL**:
		select ?x where {
		?x rdf:type _Concept.
		?x _Relation _Instance.
		}
		GeoSPARQL v1:
		select ?x where {
		?x rdf:type _Concept; geo:hasGeometry ?xGeom.
		?xGeom geo:asWKT ?xWKT.
		_Instance geo:hasGeometry ?iGeom. ?iGeom geo:asWKT ?iWKT.
		FILTER(_Relation(?xWKT, ?iWKT))
		}
		GeoSPARQL v2:
		select ?x where {
		?x rdf:type _Concept;
		geo:hasGeometry ?xGeom.
		?xGeom geo:asWKT ?xWKT.
		?instance owl:sameAs _Instance; geo:hasGeometry ?iGeom.
		?iGeom geo:asWKT ?iWKT.
		FILTER(_Relation(?xWKT, ?iWKT))
		}

CRIRI	Which churches are close to the Shannon in Limerick?	select ?x where {
		?x rdf:type _Concept; geo:hasGeometry ?xGeom.
		?xGeom geo:asWKT ?xWKT.
		_Instance1 geo:hasGeometry ?i1Geom.
		?i1Geom geo:asWKT ?i1WKT.
		_Instance2 geo:hasGeometry ?i2Geom.
		?i2Geom geo:asWKT ?i2WKT.
		FILTER(_Relation1(?xWKT, ?i1WKT) && _Relation2(?i1WKT, ?i2WKT))
		}

CRC	Which restaurants are near hotels?	select ?x where {
		?x rdf:type _Concept1; geo:hasGeometry ?xGeom.
		?xGeom geo:asWKT ?xWKT.
		?y rdf:type _Concept2; geo:hasGeometry ?yGeom.
		?yGeom geo:asWKT ?yWKT.
		FILTER(_Relation(?xWKT, ?yWKT))
		}

CRCRI	Which restaurants are near hotels in Limerick?	select ?x where {
		?x rdf:type _Concept1; geo:hasGeometry ?xGeom.
		?xGeom geo:asWKT ?xWKT.
		?y rdf:type _Concept2; geo:hasGeometry ?yGeom.
		?yGeom geo:asWKT ?yWKT.
		_Instance geo:hasGeometry ?zGeom.
		?zGeom geo:asWKT ?zWKT.
		FILTER(_Relation1(?xWKT, ?yWKT) && _Relation2(?xWKT, ?zWKT) &&
		_Relation2(?yWKT, ?zWKT)
		}

IRI	Is Hampshire north of Berkshire?	ASK where {
		_Instance1 geo:hasGeometry ?iGeom1.
		?iGeom1 geo:asWKT ?iWKT1.
		_Instance2 geo:hasGeometry ?iGeom2.
		?iGeom2 geo:asWKT ?iWKT2.
		FILTER(_Relation(?iWKT1, ?iWKT2))
		}

Table 9: Supported question patterns with examples and corresponding SPARQL/GeoSPARQL query templates for categories 6,7

Pattern	Example natural language question	Templates

NCRI	How many hospitals are there in Oxford?	**SPARQL**:
		`select (count(distinct ?x) as ?total) where {`
		`?x rdf:type _Concept.`
		`?x _Relation _Instance.`
		`}`
		GeoSPARQL v1:
		`select (count(distinct ?x) as ?total) where {`
		`?x rdf:type _Concept; geo:hasGeometry ?xGeom.`
		`?xGeom geo:asWKT ?xWKT.`
		`_Instance geo:hasGeometry ?iGeom. ?iGeom geo:asWKT ?iWKT.`
		`FILTER(_Relation(?xWKT, ?iWKT))`
		`}`
		GeoSPARQL v2:
		`select (count(distinct ?x) as ?total) where {`
		`?x rdf:type _Concept;`
		`geo:hasGeometry ?xGeom.`
		`?xGeom geo:asWKT ?xWKT.`
		`?instance owl:sameAs _Instance; geo:hasGeometry ?iGeom.`
		`?iGeom geo:asWKT ?iWKT.`
		`FILTER(_Relation(?xWKT, ?iWKT))`
		`}`

PCRI	What is the length of the river that crosses Limerick?	**SPARQL**:
		`select ?property where {`
		`?x rdf:type _Concept.`
		`?x _Relation _Instance.`
		`?x _Property ?property.`
		`}`
		GeoSPARQL :
		`select ?property where {`
		`SERVICE <http://pyravlos1.di.uoa.gr:8080/geoqa/> {`
		`?x rdf:type _Concept; geo:hasGeometry ?xGeom.`
		`?xGeom geo:asWKT ?xWKT.`
		`_Instance geo:hasGeometry ?iGeom.`
		`?iGeom geo:asWKT ?iWKT.`
		`?x owl:sameAs ?dbpediaLink. FILTER(_Relation(?xWKT, ?iWKT)) }`
		`SERVICE <http://dbpedia.org/sparql> {`
		`?dbpediaLink _Property ?property }`
		`}`

PCRIRI	What is the name of the river that flows under the Queensway Bridge in Liverpool?	**GeoSPARQL** :
		`select ?property where {`
		`SERVICE <http://pyravlos1.di.uoa.gr:8080/geoqa/> {`
		`?x rdf:type _Concept; geo:hasGeometry ?xGeom.`
		`?xGeom geo:asWKT ?xWKT.`
		`_Instance1 geo:hasGeometry ?i1Geom.`
		`?i1Geom geo:asWKT ?i1WKT.`
		`_Instance2 geo:hasGeometry ?i2Geom.`
		`?i2Geom geo:asWKT ?i2WKT.`
		`?x owl:sameAs ?dbpediaLink.`
		`FILTER(_Relation1(?xWKT, ?i1WKT)) && _Relation2(?i1WKT, ?i2WKT) }`
		`SERVICE <http://dbpedia.org/sparql> {`
		`?dbpediaLink _Property ?property }`
		`}`

Listing 2: SPARQL/GeoSPARQL Query for Motivating Example 2

```
Question: How many hospitals are there in Oxford?
SPARQL:
select (count(distinct ?x) as ?total)
where {
       ?x rdf:type dbo:Hospital.
       ?x dbp:locatedIn dbr:Oxford.
}

GeoSPARQL:
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX geof: <http://www.opengis.net/def/function/geosparql/>
PREFIX osmo: <http://www.app-lab.eu/osm/ontology#>

select (count(distinct ?x)as ?total)
where {
           ?x rdf:type osmo:Hospital;
               geo:hasGeometry ?xGeom.
           ?xGeom geo:asWKT ?xWKT.

          gadmr:Oxford geo:hasGeometry ?iGeom.
           ?iGeom geo:asWKT ?iWKT.

           FILTER(geof:sfWithin(?xWKT, ?iWKT))
}
```

The row of Table 8 for pattern CRI contains two GeoSPARQL queries (v1 and v2). The second query is for the case when the identified instance is a DBpedia resource for which geometry information is available in GADM or OSM. In addition to that, the rows for the patterns PCRI and PCRIRI in Table 9 contains a service tag for a GeoSPARQL query in order to fetch information from two different endpoints to execute the query. This query is for the case when the identified instance is a DBpedia resource for which geometry information is available in OSM while attributes like `dbp:length` or `dbp:height` are in DBpedia. This is where the `owl:sameAs` sentences produced by our interlinking process discussed in Section 3 are used. Listing 3 and Listing 4 show examples of these cases. Similar templates exist for all the other patterns.

Because we want to increase recall, our strategy is to use more than one component of GeoData201 for answering a question. For example, for the question "Which towns in England are east of Manchester?" DBpedia gives us 3 answers (Glossop, Stallybridge and Hyde) while OSM gives us 1626 towns. Our strategy for increasing precision is to have the query generator take into account class and property information from the ontologies of the three datasets. This is illustrated by the SPARQL query in Listing 1 where we make use of the fact that the property `dbo:city` is used in DBpedia to refer to the cities crossed by a river. To implement this strategy we keep a table with three columns which contains triples of the form domain-property-range for each property in the dataset GeoData201. Some example rows can be seen in Table 10. This approach has also been taken in [15].

The last job of the query generator is to rank the generated queries. Query ranking is a crucial component of a question answering system. In the current version of GeoQA, we use a very simple heuristic for the ranking of generated queries based on the estimated selectivity of the generated queries. We compute the selectivity of a SPARQL or GeoSPARQL query taking into account only the triple patterns present in the query and using the formulas of [46]. The

Listing 3: GeoSPARQL Query for Motivating Example 3

```
Question:Which forest is near Manchester?
GeoSPARQL:
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX geof: <http://www.opengis.net/def/function/geosparql/>
PREFIX osmo: <http://www.app-lab.eu/osm/ontology#>
select ?x
where {
          ?x rdf:type osmo:Forest;
             geo:hasGeometry ?xGeom.
          ?xGeom geo:asWKT ?xWKT.
          ?instance owl:sameAs dbr:Manchester;
             geo:hasGeometry ?iGeom.
          ?iGeom geo:asWKT ?iWKT.
          FILTER(geof:distance(?xWKT,?iWKT,uom:metre) <= 5000)
}
```

Table 10: domain-property-range Table

Domain	**Property**	**Range**
Airport	Within	http://dbpedia.org/ontology/city
Airport	Within	http://dbpedia.org/ontology/country
Airport	Within	http://dbpedia.org/ontology/county
River	Within	http://dbpedia.org/ontology/country
River	Within	http://dbpedia.org/ontology/county
River	Crosses	http://dbpedia.org/ontology/Bridge
River	Crosses	http://dbpedia.org/ontology/city

Listing 4: SPARQL/GeoSPARQL Query for Motivating Example 4

```
Question:What is the length of the river that crosses Limerick?
SPARQL:
select ?length
where {
          ?x rdf:type   dbo:River.
          ?x dbo:city   dbr:Limeric.
          ?X dbp:length ?length.
}
GeoSPARQL:
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX geof: <http://www.opengis.net/def/function/geosparql/>
PREFIX osmo: <http://www.app-lab.eu/osm/ontology#>

select ?length
where {
      SERVICE <http://pyravlos1.di.uoa.gr:8080/geoqa/Query>
      {
        ?x rdf:type osmo:River;
        geo:hasGeometry ?xGeom;
        owl:sameAs ?dbpediaLink.
        ?xGeom geo:asWKT ?xWKT.

        gadmr:Limerick geo:hasGeometry ?iGeom.
        ?iGeom geo:asWKT ?iWKT.

        FILTER(geof:sfCrosses(?xWKT, ?iWKT))
     }
    SERVICE <http://dbpedia.org/sparql>
    {
       ?dbpediaLink dbp:length ?length
    }
}
```

generated query with the lowest selectivity is selected to be executed; in this way, we expect to generate more results to the user question.

Expressive Power of Patterns.. It is interesting to consider the expressive power of patterns in Tables 8 and 9 by giving a corresponding binary first-order logic formula.[33] Questions following pattern IP can be written as $x\ :\ P(I,x)$. Questions following the CRI pattern can be written formally as $x\ :\ C(x) \wedge (\exists i)R(x,i)$. Questions following the pattern CRIRI can be written as $x\ :\ C(x) \wedge (\exists i_1)(\exists i_2)(R_1(x,i_1) \wedge R_2(i_1,i_2))$. Questions following the pattern CRC can be written as $x\ :\ C_1(x) \wedge (\exists i)(C_2(i) \wedge R(x,i))$. Questions written as CRCRI can be written as $x\ :\ C_1(x) \wedge (\exists i_1)(\exists i_2)(R(x,i_1) \wedge C_2(i_2) \wedge R_2(i_1,i_2))$. Questions following pattern IRI can be written as : $R(I_1,I_2)$. Questions following the PCRI pattern can be written formally as $v : (\exists x)(C(x) \wedge R(x,I) \wedge P(I,v))$.

[33] In the following formulas, we assume that identifiers (i.e., geographic features) are denoted by constants, concepts (i.e., classes of features) by unary predicates and geospatial relations by binary predicates. Constants and predicates are denoted by capital letters while variables are denoted by lowercase letters. Variables are assumed to range over identifiers with the exception of variable v in the case pf PCRI which ranges over values. The ":" symbol should be read as "such that".

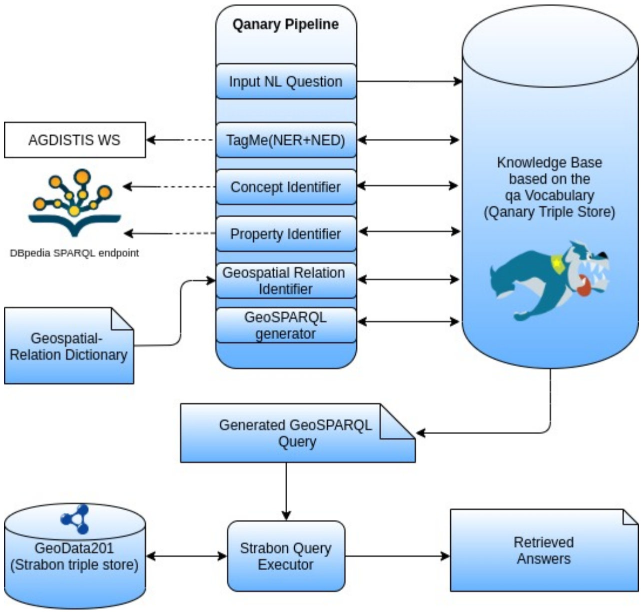

Figure 2: Architecture of Implementation of GeoQA

Query executor.. The last module executes the top-ranked SPARQL or GeoSPARQL query against a Strabon [4] endpoint which also communicates with a DBpedia endpoint through the use of the `SERVICE` keyword in queries. If no query has been generated, the user is notified that the question could not be answered.

6. Evaluation

The current version of the GeoQA engine presented above has been evaluated using the gold standard dataset GeoData201 and questions GeoQuestions201 presented in Sections 3 and 4. GeoQA was run using the 153 questions that fall under categories 1 to 6 and only the count questions from category 7 i.e. "How many hospitals are there in Oxford?". The questions like "What is the longest bridge in Scotland?" fall under category 7 but are not targeted at the moment.

Table 11 summarizes the effectiveness of GeoQA on the gold standard using the well-known metrics of precision, recall and F1. We calculate precision, recall and F1 using the formulas below for the 153 questions.

$Precision = (|CorrectAnswers \cap RetrievedAnswers|)/|RetrivedAnswers|$

$Recall = (|CorrectAnswers \cap RetrievedAnswers|)/|CorrectAnswers|$

$F1 = 2 * ((Precision * Recall)/(Precision + Recall))$

We calculated these f-measures for each questions individually and considered average of them as final f-measure for our system. Table 11 presents the average of precision,recall and F1 on all 153 questions. For the 22 questions (out of 153), GeoQA does not identify the correct question pattern and as a result no query is generated. From 131 generated queries, only 115 queries return results. The remaining 16 queries do not generate any answers, although such

Table 11: Evaluation of GeoQA

Gold Questions	Answered Questions	Precision	Recall	F1
153	131	64.63%	65.10%	62.21%

answers exist in the GeoData201 dataset, because of errors in different component that we will discuss further and summerise in Table 12. There are some questions for which the instance identifier component fails to annotate the correct entity or does not annotate any entity in the question. E.g., in Q37 “Which counties of Scotland border England?”, TagMe disambiguate to `dbr:Scottish_Borders` that is wrong instance. For some of the questions the concept identifier fails to identify the appropriate class. E.g., in Q146 “Which city council includes Dublin?”, the concept identifier identifies class *city* instead of *city council*. Sometimes the property identifier component fails to map or to infer the correct property from question text. E.g., in Q86 “Which villages in Scotland have a population of less than 500 people?”, Property identifier maps population to `dbp:population` while the most common property for villages in Scotland inside DBpedia is `dbp:populationTotal`. In Q80 “Is there a mountain in the county of Greater Manchester taller than 1300 meters above sea level?”, the question is asking about elevation of mountain and property identifier fails to find the right property. Some other times, the query generator module fails to identify the appropriate pattern for some of the questions. E.g., in Q29 “Which airports are in the city of Salford?”, the identified pattern must be CRI but instead, the query generator identifies CRIRI. Also sometimes the selection of wrong queries would result in no answers. E.g., in Q151 “Which pubs are near Mercure Hotel in Glasgow, Scotland?” the query generator selects a query containing `owl:sameAs dbr:Mercure_Hotels` which is not linked in our dataset resulting in 0 answers.

7. Conclusion

In this paper we have addressed the challenges of providing access to linked geospatial data for non-expert users using natural language QA interfaces. Given the use of geospatial contexts in many practical situations this challenge is of major importance while adopting QA for wide use. Our main contribution was the implementation of GeoQA which is, to the best of our knowledge, the first QA engine which is able to answer questions with a geospatial dimension. We have also evaluated GeoQA using a gold standard dataset and set of questions which we make publicly available so it can also be used by other researchers.

In future work we plan to improve all the components of GeoQA so we can increase precision, recall and F1 measure further. In addition, we plan to deal with more complicated questions going beyond the question patterns we discussed in this paper. Finally, we plan to consider temporal questions [47] since temporal and spatial questions arise together naturally in many application contexts.

Funding: This work has been funded by the Horizon 2020 Marie Sklodowska-Curie Innovative Training Network WDAqua (Answering Questions using Web Data) with grant agreement No. 642795 and by Hellenic Foundation for Research and Innovation (HFRI) and the General Secretariat for Research and Technology (GSRT), under the grant GeoQa (GA. no HFRI-FM17-2351).

Table 12: Breakdown of problematic questions according to pattern they follow

	Question Number						
Module Responsible	**CRI**	**CRIRI**	**CRCRI**	**IRI**	**IP**	**PCRI**	**PCRIRI**
Instance Identifier	Q159, Q172	Q37, Q38, Q63, Q64, Q67, Q182, Q194		Q180	Q40, Q158		Q100
Concept Identifier	Q4, Q118, Q146, Q149					Q177	
Property Identifier						Q80, Q82, Q86	
Query Generator	Q23, Q29, Q68, Q92, Q127	Q13, Q38, Q48, Q77, Q126, Q129, Q151, Q182, Q196	Q164, Q170	Q127, Q168, Q180		Q187	Q120

References

[1] L. Hirschman, R. J. Gaizauskas, Natural language question answering: the view from here, Nat. Lang. Eng. 7 (2001) 275–300. URL: https://doi.org/10.1017/S1351324901002807. doi:10.1017/S1351324901002807.

[2] J. Lee, M. Kang, Geospatial big data: Challenges and opportunities, Big Data Res. 2 (2015) 74–81. URL: https://doi.org/10.1016/j.bdr.2015.01.003. doi:10.1016/j.bdr.2015.01.003.

[3] S. Auer, J. Lehmann, S. Hellmann, Linkedgeodata: Adding a spatial dimension to the web of data, in: A. Bernstein, D. R. Karger, T. Heath, L. Feigenbaum, D. Maynard, E. Motta, K. Thirunarayan (Eds.), The Semantic Web - ISWC 2009, 8th International Semantic Web Conference, ISWC 2009, Chantilly, VA, USA, October 25-29, 2009. Proceedings, volume 5823 of *Lecture Notes in Computer Science*, Springer, 2009, pp. 731–746. URL: https://doi.org/10.1007/978-3-642-04930-9_46. doi:10.1007/978-3-642-04930-9_46.

[4] K. Kyzirakos, M. Karpathiotakis, M. Koubarakis, Strabon: A semantic geospatial DBMS, in: P. Cudré-Mauroux, J. Heflin, E. Sirin, T. Tudorache, J. Euzenat, M. Hauswirth, J. X. Parreira, J. Hendler, G. Schreiber, A. Bernstein, E. Blomqvist (Eds.), The Semantic Web - ISWC 2012 - 11th International Semantic Web Conference, Boston, MA, USA, November 11-15, 2012, Proceedings, Part I, volume 7649 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 295–311. URL: https://doi.org/10.1007/978-3-642-35176-1_19. doi:10.1007/978-3-642-35176-1_19.

[5] D. Diefenbach, V. López, K. D. Singh, P. Maret, Core techniques of question answering systems over knowledge bases: a survey, Knowl. Inf. Syst. 55 (2018) 529–569. URL: https://doi.org/10.1007/s10115-017-1100-y. doi:10.1007/s10115-017-1100-y.

[6] I. Schlaisich, M. J. Egenhofer, Multimodal spatial querying: what people sketch and talk about, in: C. Stephanidis (Ed.), Universal Access In HCI: Towards an Information Society for All, Proceedings of HCI International '2001 (the 9th International Conference on Human-Computer Interaction), New Orleans, USA, August 5-10, 2001, Volume 3, Lawrence Erlbaum, 2001, pp. 732–736.

[7] K. Singh, A. Both, D. Diefenbach, S. Shekarpour, D. Cherix, C. Lange, Qanary - the fast track to creating a question answering system with linked data technology, in: H. Sack, G. Rizzo, N. Steinmetz, D. Mladenic, S. Auer, C. Lange (Eds.), The Semantic Web - ESWC 2016 Satellite Events, Heraklion, Crete, Greece, May 29 - June 2, 2016, Revised Selected Papers, volume 9989 of *Lecture Notes in Computer Science*, 2016, pp. 183–188. URL: https://doi.org/10.1007/978-3-319-47602-5_36. doi:10.1007/978-3-319-47602-5_36.

[8] A. Both, D. Diefenbach, K. Singh, S. Shekarpour, D. Cherix, C. Lange, Qanary - A methodology for vocabulary-driven open question answering systems, in: H. Sack, E. Blomqvist, M. d'Aquin, C. Ghidini, S. P. Ponzetto, C. Lange (Eds.), The Semantic Web. Latest Advances and New Domains - 13th International Conference, ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Proceedings, volume 9678 of *Lecture Notes in Computer Science*, Springer, 2016, pp. 625–641. URL: `https://doi.org/10.1007/978-3-319-34129-3_38`. doi:`10.1007/978-3-319-34129-3\_38`.

[9] K. Singh, A. S. Radhakrishna, A. Both, S. Shekarpour, I. Lytra, R. Usbeck, A. Vyas, A. Khikmatullaev, D. Punjani, C. Lange, M. Vidal, J. Lehmann, S. Auer, Why reinvent the wheel: Let's build question answering systems together, in: P. Champin, F. L. Gandon, M. Lalmas, P. G. Ipeirotis (Eds.), Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018, ACM, 2018, pp. 1247–1256. URL: `https://doi.org/10.1145/3178876.3186023`. doi:`10.1145/3178876.3186023`.

[10] D. Punjani, K. Singh, A. Both, M. Koubarakis, I. Angelidis, K. Bereta, T. Beris, D. Bilidas, T. Ioannidis, N. Karalis, et al., Template-based question answering over linked geospatial data, in: Proceedings of the 12th Workshop on Geographic Information Retrieval, 2018, pp. 1–10.

[11] C. B. Jones, R. Purves, A. Ruas, M. Sanderson, M. Sester, M. J. van Kreveld, R. Weibel, Spatial information retrieval and geographical ontologies an overview of the SPIRIT project, in: K. Järvelin, M. Beaulieu, R. A. Baeza-Yates, S. Myaeng (Eds.), SIGIR 2002: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 11-15, 2002, Tampere, Finland, ACM, 2002, pp. 387–388. URL: `https://doi.org/10.1145/564376.564457`. doi:`10.1145/564376.564457`.

[12] M. D. Lieberman, H. Samet, J. Sankaranarayanan, J. Sperling, STEWARD: architecture of a spatio-textual search engine, in: H. Samet, C. Shahabi, M. Schneider (Eds.), 15th ACM International Symposium on Geographic Information Systems, ACM-GIS 2007, November 7-9, 2007, Seattle, Washington, USA, Proceedings, ACM, 2007, p. 25. URL: `https://doi.org/10.1145/1341012.1341045`. doi:`10.1145/1341012.1341045`.

[13] D. Santos, N. Cardoso, P. Carvalho, I. Dornescu, S. Hartrumpf, J. Leveling, Y. Skalban, Getting geographical answers from wikipedia: the gikip pilot at clef, in: CLEF, 2008.

[14] K. Bereta, M. Koubarakis, Ontop of geospatial databases, in: P. T. Groth, E. Simperl, A. J. G. Gray, M. Sabou, M. Krötzsch, F. Lécué, F. Flöck, Y. Gil (Eds.), The Semantic Web - ISWC 2016 - 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016, Proceedings, Part I, volume 9981 of *Lecture Notes in Computer Science*, 2016, pp. 37–52. URL: `https://doi.org/10.1007/978-3-319-46523-4_3`. doi:`10.1007/978-3-319-46523-4\_3`.

[15] E. M. G. Younis, C. B. Jones, V. Tanasescu, A. I. Abdelmoty, Hybrid geo-spatial query methods on the semantic web with a spatially-enhanced index of dbpedia, in: N. Xiao, M. Kwan, M. F. Goodchild, S. Shekhar (Eds.), Geographic Information Science - 7th International Conference, GIScience 2012, Columbus, OH, USA, September 18-21, 2012. Proceedings, volume 7478 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 340–353. URL: `https://doi.org/10.1007/978-3-642-33024-7_25`. doi:`10.1007/978-3-642-33024-7\_25`.

[16] R. Grütter, R. S. Purves, L. Wotruba, Evaluating topological queries in linked data using dbpedia and geonames in switzerland and scotland, Trans. GIS 21 (2017) 114–133. URL: `https://doi.org/10.1111/tgis.12196`. doi:`10.1111/tgis.12196`.

[17] E. Hamzei, H. Li, M. Vasardani, T. Baldwin, S. Winter, M. Tomko, Place questions and human-generated answers: A data analysis approach, in: P. C. Kyriakidis, D. G. Hadjimitsis, D. Skarlatos, A. Mansourian (Eds.), Geospatial Technologies for Local and Regional Development - Proceedings of the 22nd AGILE Conference on Geographic Information Science, Limassol, Cyprus, June 17-20, 2019, Lecture Notes in Geoinformation and Cartography, Springer, 2019, pp. 3–19. URL: `https://doi.org/10.1007/978-3-030-14745-7_1`. doi:`10.1007/978-3-030-14745-7\_1`.

[18] T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, L. Deng, MS MARCO: A human generated machine reading comprehension dataset, in: T. R. Besold, A. Bordes, A. S. d'Avila Garcez, G. Wayne (Eds.), Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016, volume 1773 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2016.

[19] L. R. Tang, R. J. Mooney, Using multiple clause constructors in inductive logic programming for semantic parsing, in: L. D. Raedt, P. A. Flach (Eds.), Machine Learning: EMCL 2001, 12th European Conference on Machine Learning, Freiburg, Germany, September 5-7, 2001, Proceedings, volume 2167 of *Lecture Notes in Computer Science*, Springer, 2001, pp. 466–477. URL: `https://doi.org/10.1007/3-540-44795-4_40`. doi:`10.1007/3-540-44795-4\_40`.

[20] V. López, M. Fernández, E. Motta, N. Stieler, Poweraqua: Supporting users in querying and exploring the semantic web, Semantic Web 3 (2012) 249–265. URL: `https://doi.org/10.3233/SW-2011-0030`. doi:`10.3233/SW-2011-0030`.

[21] J. Salas, A. Harth, Finding spatial equivalences accross multiple rdf datasets, in: Proceedings of the Terra Cognita Workshop on Foundations, Technologies and Applications of the Geospatial Web, Citeseer, 2011, pp. 114–126.

[22] C. Stadler, J. Lehmann, K. Höffner, S. Auer, Linkedgeodata: A core for a web of spatial open data, Semantic Web 3 (2012) 333–354. URL: https://doi.org/10.3233/SW-2011-0052. doi:10.3233/SW-2011-0052.

[23] J. Volz, C. Bizer, M. Gaedke, G. Kobilarov, Silk - A link discovery framework for the web of data, in: C. Bizer, T. Heath, T. Berners-Lee, K. Idehen (Eds.), Proceedings of the WWW2009 Workshop on Linked Data on the Web, LDOW 2009, Madrid, Spain, April 20, 2009, volume 538 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2009.

[24] V. T. Ho, Y. Ibrahim, K. Pal, K. Berberich, G. Weikum, Qsearch: Answering quantity queries from text, in: The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part I, 2019, pp. 237–257.

[25] D. Diefenbach, K. Singh, A. Both, D. Cherix, C. Lange, S. Auer, The qanary ecosystem: Getting new insights by composing question answering pipelines, in: J. Cabot, R. D. Virgilio, R. Torlone (Eds.), Web Engineering - 17th International Conference, ICWE 2017, Rome, Italy, June 5-8, 2017, Proceedings, volume 10360 of *Lecture Notes in Computer Science*, Springer, 2017, pp. 171–189. URL: https://doi.org/10.1007/978-3-319-60131-1_10. doi:10.1007/978-3-319-60131-1_10.

[26] A. Both, K. Singh, D. Diefenbach, I. Lytra, Rapid engineering of QA systems using the light-weight qanary architecture, in: J. Cabot, R. D. Virgilio, R. Torlone (Eds.), Web Engineering - 17th International Conference, ICWE 2017, Rome, Italy, June 5-8, 2017, Proceedings, volume 10360 of *Lecture Notes in Computer Science*, Springer, 2017, pp. 544–548. URL: https://doi.org/10.1007/978-3-319-60131-1_40. doi:10.1007/978-3-319-60131-1_40.

[27] M. Dubey, S. Dasgupta, A. Sharma, K. Höffner, J. Lehmann, Asknow: A framework for natural language query formalization in SPARQL, in: H. Sack, E. Blomqvist, M. d'Aquin, C. Ghidini, S. P. Ponzetto, C. Lange (Eds.), The Semantic Web. Latest Advances and New Domains - 13th International Conference, ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Proceedings, volume 9678 of *Lecture Notes in Computer Science*, Springer, 2016, pp. 300–316. URL: https://doi.org/10.1007/978-3-319-34129-3_19. doi:10.1007/978-3-319-34129-3_19.

[28] V. López, E. Motta, V. S. Uren, Poweraqua: Fishing the semantic web, in: Y. Sure, J. Domingue (Eds.), The Semantic Web: Research and Applications, 3rd European Semantic Web Conference, ESWC 2006, Budva, Montenegro, June 11-14, 2006, Proceedings, volume 4011 of *Lecture Notes in Computer Science*, Springer, 2006, pp. 393–410. URL: https://doi.org/10.1007/11762256_30. doi:10.1007/11762256_30.

[29] C. Unger, L. Bühmann, J. Lehmann, A. N. Ngomo, D. Gerber, P. Cimiano, Template-based question answering over RDF data, in: A. Mille, F. L. Gandon, J. Misselis, M. Rabinovich, S. Staab (Eds.), Proceedings of the 21st World Wide Web Conference 2012, WWW 2012, Lyon, France, April 16-20, 2012, ACM, 2012, pp. 639–648. URL: https://doi.org/10.1145/2187836.2187923. doi:10.1145/2187836.2187923.

[30] J. Nivre, M. de Marneffe, F. Ginter, Y. Goldberg, J. Hajic, C. D. Manning, R. T. McDonald, S. Petrov, S. Pyysalo, N. Silveira, R. Tsarfaty, D. Zeman, Universal dependencies v1: A multilingual treebank collection, in: N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis (Eds.), Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016, European Language Resources Association (ELRA), 2016.

[31] P. Ferragina, U. Scaiella, TAGME: on-the-fly annotation of short text fragments (by wikipedia entities), in: J. Huang, N. Koudas, G. J. F. Jones, X. Wu, K. Collins-Thompson, A. An (Eds.), Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, Ontario, Canada, October 26-30, 2010, ACM, 2010, pp. 1625–1628. URL: https://doi.org/10.1145/1871437.1871689. doi:10.1145/1871437.1871689.

[32] J. R. Finkel, T. Grenager, C. D. Manning, Incorporating non-local information into information extraction systems by gibbs sampling, in: K. Knight, H. T. Ng, K. Oflazer (Eds.), ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA, The Association for Computer Linguistics, 2005, pp. 363–370.

[33] M. A. Yosef, J. Hoffart, I. Bordino, M. Spaniol, G. Weikum, AIDA: an online tool for accurate disambiguation of named entities in text and tables, PVLDB 4 (2011) 1450–1453.

[34] P. N. Mendes, M. Jakob, A. García-Silva, C. Bizer, Dbpedia spotlight: shedding light on the web of documents, in: C. Ghidini, A. N. Ngomo, S. N. Lindstaedt, T. Pellegrini (Eds.), Proceedings the 7th International Conference on Semantic Systems, I-SEMANTICS 2011, Graz, Austria, September 7-9, 2011, ACM International Conference Proceeding Series, ACM, 2011, pp. 1–8. URL: https://doi.org/10.1145/2063518.2063519. doi:10.1145/2063518.2063519.

[35] R. Usbeck, A. N. Ngomo, M. Röder, D. Gerber, S. A. Coelho, S. Auer, A. Both, AGDISTIS - graph-based disambiguation of named entities using linked data, in: P. Mika, T. Tudorache, A. Bernstein, C. Welty, C. A. Knoblock, D. Vrandecic, P. T. Groth, N. F. Noy, K. Janowicz, C. A. Goble (Eds.), The Semantic Web - ISWC 2014 - 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I, volume 8796 of *Lecture Notes in Computer Science*, Springer, 2014, pp. 457–471. URL: https://doi.org/10.1007/978-3-319-11964-9_29. doi:10.1007/978-3-319-11964-9_29.

[36] A. Moro, A. Raganato, R. Navigli, Entity linking meets word sense disambiguation: a unified approach, Trans. Assoc. Comput. Linguistics 2 (2014) 231–244.

[37] M. J. Egenhofer, R. D. Franzosa, Point set topological relations, International Journal of Geographical Information Systems 5 (1991) 161–174. URL: `https://doi.org/10.1080/02693799108927841`. doi:`10.1080/02693799108927841`.

[38] A. U. Frank, Qualitative spatial reasoning about distances and directions in geographic space, J. Vis. Lang. Comput. 3 (1992) 343–371. URL: `https://doi.org/10.1016/1045-926X(92)90007-9`. doi:`10.1016/1045-926X(92)90007-9`.

[39] S. Skiadopoulos, M. Koubarakis, Composing cardinal direction relations, in: C. S. Jensen, M. Schneider, B. Seeger, V. J. Tsotras (Eds.), Advances in Spatial and Temporal Databases, 7th International Symposium, SSTD 2001, Redondo Beach, CA, USA, July 12-15, 2001, Proceedings, volume 2121 of *Lecture Notes in Computer Science*, Springer, 2001, pp. 299–320.

[40] E. Clementini, P. D. Felice, A model for representing topological relationships between complex geometric features in spatial databases, Inf. Sci. 90 (1996) 121–136. URL: `https://doi.org/10.1016/0020-0255(95)00289-8`. doi:`10.1016/0020-0255(95)00289-8`.

[41] M. J. van Kreveld, I. Reinbacher, Good news: Partitioning a simple polygon by compass directions, Int. J. Comput. Geometry Appl. 14 (2004) 233–259. URL: `https://doi.org/10.1142/S0218195904001469`. doi:`10.1142/S0218195904001469`.

[42] B. Regalia, K. Janowicz, S. Gao, VOLT: A provenance-producing, transparent SPARQL proxy for the on-demand computation of linked data and its application to spatiotemporally dependent data, in: H. Sack, E. Blomqvist, M. d'Aquin, C. Ghidini, S. P. Ponzetto, C. Lange (Eds.), The Semantic Web. Latest Advances and New Domains - 13th International Conference, ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Proceedings, volume 9678 of *Lecture Notes in Computer Science*, Springer, 2016, pp. 523–538. URL: `https://doi.org/10.1007/978-3-319-34129-3_32`. doi:`10.1007/978-3-319-34129-3\_32`.

[43] A. R. B. M. Shariff, M. J. Egenhofer, D. M. Mark, Natural-language spatial relations between linear and areal objects: The topology and metric of english-language terms, International Journal of Geographical Information Science 12 (1998) 215–245.

[44] D. M. Mark, M. J. Egenhofer, Topology of prototypical spatial relations between lines and regions in english and spanish, in: AUTOCARTO-CONFERENCE-, 1995, pp. 245–254.

[45] M. P. Dube, M. J. Egenhofer, An ordering of convex topological relations, in: N. Xiao, M. Kwan, M. F. Goodchild, S. Shekhar (Eds.), Geographic Information Science - 7th International Conference, GIScience 2012, Columbus, OH, USA, September 18-21, 2012. Proceedings, volume 7478 of *Lecture Notes in Computer Science*, Springer, 2012, pp. 72–86. URL: `https://doi.org/10.1007/978-3-642-33024-7_6`. doi:`10.1007/978-3-642-33024-7\_6`.

[46] M. Stocker, A. Seaborne, A. Bernstein, C. Kiefer, D. Reynolds, SPARQL basic graph pattern optimization using selectivity estimation, in: J. Huai, R. Chen, H. Hon, Y. Liu, W. Ma, A. Tomkins, X. Zhang (Eds.), Proceedings of the 17th International Conference on World Wide Web, WWW 2008, Beijing, China, April 21-25, 2008, ACM, 2008, pp. 595–604. URL: `https://doi.org/10.1145/1367497.1367578`. doi:`10.1145/1367497.1367578`.

[47] Z. Jia, A. Abujabal, R. S. Roy, J. Strötgen, G. Weikum, TEQUILA: temporal question answering over knowledge bases, in: A. Cuzzocrea, J. Allan, N. W. Paton, D. Srivastava, R. Agrawal, A. Z. Broder, M. J. Zaki, K. S. Candan, A. Labrinidis, A. Schuster, H. Wang (Eds.), Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018, ACM, 2018, pp. 1807–1810.