NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

SCHOOL OF SCIENCES
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATIONS

MSc THESIS

The Dataset GeoQuestions1089

Maria-Angeliki G. Pollali

Supervisor: **Koubarakis Manolis,** Professor

Co-Supervisor: **Kefalidis Sergios-Anestis,** Associate Researcher

ATHENS

MARCH 2024

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Το σύνολο δεδομένων GeoQuestions1089

Μαρία-Αγγελική Γ. Πολλάλη

Επιβλέπων: **Κουμπαράκης Μανόλης,** Καθηγητής

Συνεπιβλέπων: **Κεφαλίδης Σέργιος-Ανέστης,** Συνεργαζόμενος Ερευνητής

ΑΘΗΝΑ

ΜΑΡΤΙΟΣ 2024

MSc THESIS

The Dataset GeoQuestions1089

Maria-Angeliki G. Pollali
S.N.: 7115112200026

SUPERVISOR: **Koubarakis Manolis,** Professor

COSUPERVISOR: **Kefalidis Sergios-Anestis,** Associate Researcher

THESIS COMMITTEE: **Koubarakis Manolis,** Professor
Ntoulas Alexandros, Professor
Roussou Maria, Professor

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Το σύνολο δεδομένων GeoQuestions1089

Μαρία-Αγγελική Γ. Πολλάλη
Α.Μ.: 7115112200026

ΕΠΙΒΛΕΠΩΝ: **Κουμπαράκης Μανόλης,** Καθηγητής

ΣΥΝΕΠΙΒΛΕΠΩΝ: **Κεφαλίδης Σέργιος-Ανέστης,** Συνεργαζόμενος Ερευνητής

ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ: **Κουμπαράκης Μανόλης,** Καθηγητής
Ντούλας Αλέξανδρος, Επίκουρος Καθηγητής
Ρούσσου Μαρία, Αναπληρώτρια Καθηγήτρια

ABSTRACT

In this thesis, the dataset GeoQuestions1089 is presented for benchmarking geospatial question answering engines. GeoQuestions1089 is the largest such dataset available presently, comprising 1089 questions, their corresponding GeoSPARQL or SPARQL queries, and their answers over the geospatial knowledge graph YAGO2geo. GeoQuestions1089 is used to evaluate the effectiveness and efficiency of geospatial question answering engines, including GeoQA2 (an extension of GeoQA developed by our group), the system of Hamzei et al. (2021) and ChatGPT.

SUBJECT AREA: SEMANTIC WEB

KEYWORDS: Knowledge Graphs, Dataset, SPARQL, Geospatial Question Answering Engines

ΠΕΡΙΛΗΨΗ

Στην παρούσα διπλωματική, παρουσιάζεται το σύνολο δεδομένων GeoQuestions1089 για τη συγκριτική αξιολόγηση μηχανών απάντησης γεωχωρικών ερωτήσεων. Το GeoQuestions1089 είναι το μεγαλύτερο τέτοιο σύνολο δεδομένων που είναι διαθέσιμο επί του παρόντος και περιλαμβάνει 1089 ερωτήσεις, τα αντίστοιχα ερωτήματα GeoSPARQL ή SPARQL και τις απαντήσεις τους πάνω στον γράφο γεωχωρικής γνώσης YAGO2geo. Το GeoQuestions1089 χρησιμοποιείται για την αξιολόγηση της αποτελεσματικότητας και της αποδοτικότητας των μηχανών απάντησης γεωχωρικών ερωτήσεων, συμπεριλαμβανομένων του GeoQA2 (μια επέκταση του GeoQA που αναπτύχθηκε από την ομάδα μας), του συστήματος των Hamzei et al. (2021) και του ChatGPT.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: ΣΗΜΑΣΙΟΛΟΓΙΚΟΣ ΙΣΤΟΣ

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Knowledge Graphs, Dataset, SPARQL, Geospatial Question Answering Engines

ACKNOWLEDGEMENTS

Πρωτίστως θα ήθελα να ευχαριστήσω θερμά τον Καθηγητή του Εθνικού και Καποδηστριακού Πανεπιστημίου Αθηνών κύριο Κουμπαράκη Μανόλη για την ανάθεση της παρούσας διπλωματικής εργασίας καθώς επίσης και για την πολύτιμη καθοδήγηση του διάρκεια των μεταπτυχιακών σπουδών μου.

Τέλος θα ήθελα να εκφράσω τις βαθύτατες ευχαριστίες μου στην οικογένεια μου και ιδιαίτερα στους γονείς μου Γεώργιο Πολλάλη και Στυλιανή Νταντούτη για την υποστήριξη τους καθ' όλη τη διάρκεια των σπουδών μου και την αμέριστη συμπαράσταση που μου πρόσφεραν όλα αυτά τα χρόνια.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Problem Statement

Recognizing the limitations of existing geospatial datasets, the study takes proactive steps to create a new dataset. Following the construction of this new dataset, is evaluated by comparing it against existing geospatial question answering systems and language models.

1.2 Project Scope

The thesis is primarily focused on two key aspects: firstly, the construction of the GeoQuestions1089 dataset, and secondly, the evaluation of three prominent geospatial question answering engines: GeoQA2, the system developed by Hamzei et al., and ChatGPT. The initial emphasis of the thesis lies in meticulously constructing the GeoQuestions1089 dataset. This involves gathering and curating a comprehensive collection of geospatial questions covering a wide spectrum of topics and complexities. Subsequently, the focus shifts to evaluating the performance of the identified QA engines.

1.3 Aim and Objectives

The overarching aim of this study is twofold. Firstly, I aim to construct and meticulously curate the GeoQuestions1089 dataset, which serves as the foundation for evaluating geospatial question answering engines. This involves gathering a diverse range of geospatial questions covering various topics and complexities. Secondly, my aim is to evaluate the performance of three prominent geospatial question answering engines: GeoQA2, the system developed by Hamzei et al., and ChatGPT. To achieve these aims, specific objectives have been outlined, including assessing the engines' proficiency in interpreting and responding to different categories of geospatial questions, identifying their strengths and weaknesses, and providing insights for potential enhancements.

1.4 Thesis Layout

In the Thesis Layout section, I provide an overview of how this study is organized. It starts with the Introduction, where I explain the importance of geospatial question answering systems and outline the study's objectives. Then, the Preliminaries section covers essential background information. Next, the Related Work section discusses previous research in the field, setting the stage for our investigation. The Main Work section focuses on the core contributions, including constructing the GeoQuestions1089 dataset and evaluating key question answering engines. Following this, the Evaluation section presents our findings. Finally, the Conclusions and Future Work section wraps up the study, summarizing our results and suggesting areas for future research in geospatial question answering.

2. PRELIMINARIES

2.1 Answering questions expressed in natural language over knowledge graphs

Users are often interested in posing geospatial questions to search engines, question answering (QA) engines and chatbots. Examples of such geospatial questions are: “Which rivers cross London?”, “Is there a Levi’s store in Athens?” and “Which countries border Greece, have the euro as their currency and their population is greater than the population of Greece?”. In this thesis, I deal with the problem of answering such questions over geospatial knowledge graphs i.e., knowledge graphs (KGs) which represent knowledge about geographic features or simply features in the terminology of GIS systems [25, 27]. Geospatial knowledge in KGs is encoded using latitude/longitude pairs representing the center of features (as e.g., in DBpedia and YAGO2), but also more detailed geometries (e.g., lines, polygons, multipolygons etc.) since these are more appropriate for modeling the geometries of features such as rivers, roads, countries etc. (as in Wikidata [51], YAGO2geo [19], WorldKG [8] and KnowWhereGraph [14]).

The development of the above geospatial KGs has given rise to geospatial QA engines for them. Examples of such systems are the GeoQA engine developed by our group [32, 31] and the systems of [2, 40, 12, 23, 55]. To evaluate the effectiveness and efficiency of these engines, there is currently only one benchmark: the GeoQuestions201 dataset proposed by our group [32] and used in comparing GeoQA with the systems of [11, 12] and [23]. In this thesis we go beyond GeoQuestions201 and make the following original contributions.

AI team and I present the benchmark GeoQuestions1089, which contains 1089 triples of geospatial questions, their answers, and the respective SPARQL/GeoSPARQL queries. GeoQuestions1089 is currently the largest geospatial QA benchmark and it is made freely available to the research community[1]. In addition to simple questions like those present in GeoQuestions201, GeoQuestions1089 contains semantically complex questions that require a sophisticated understanding of both natural language and GeoSPARQL to be answered. Furthermore, it expands the geographical area of interest, by including questions about the United States and Greece. This expanded list of countries of interest introduces additional challenges that QA engines must overcome. In this way, we contribute to a long-term research agenda towards QA systems with geospatial features.

Using GeoQuestions1089, we evaluate the effectiveness and efficiency of geospatial QA engines GeoQA2 and the engine of Hamzei et al. [12] and find that although GeoQA2 emerges victorious, mainly because of its disambiguation component, neither engine is able to process complex questions caused by both a limited vocabulary of geospatial relations and a template-based approach to query generation. We stress here that the competitor engine of Hamzei et al. has been designed to target YAGO2geo and therefore cannot answer questions such as “What is the length of the Awali river?” because the entity `yago:Awali_(river)` appears in YAGO2 but not in YAGO2geo meaning that it is lacking detailed geospatial information which is expected by the query generator of the engine.

We show that the pre-computation and materialization of entailed, but not stored explicitly, topological relations between entities in geospatial KGs can lead to substantial savings in geospatial query processing time. We show experimentally that this can speed up question answering for both engines studied.

[1] https://github.com/AI-team-UoA/GeoQuestions1089

2.1.1 The GeoQA2 pipeline

GeoQA2 takes as input a question in natural language (currently only English is supported) and the union of KGs YAGO and YAGO2geo, and produces one or more answers.[2] Question answering is performed by translating the input question into a set of SPARQL/GeoSPARQL queries, ranking these queries, and executing the top ranked query over the YAGO2geo endpoint[3]. In Figure 2.1 we illustrate the conceptual view of the GeoQA2 pipeline, which contains the following components:

- Dependency and constituency parse tree generator
- Concept identifier
- Instance identifier
- Geospatial relation identifier
- Property identifier
- Query generator

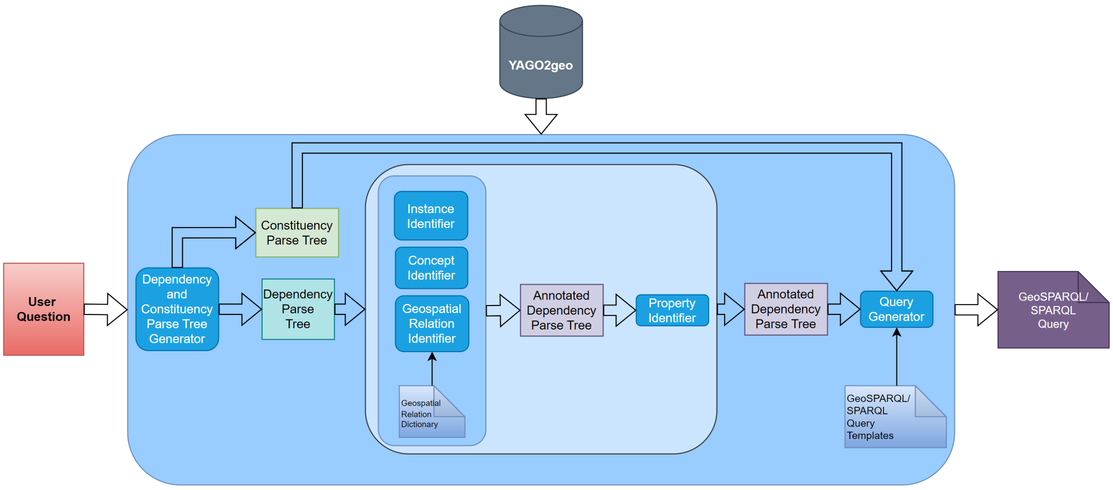
YAGO2geo
User Question
Dependency and Constituency Parse Tree Generator
Constituency Parse Tree
Dependency Parse Tree
Instance Identifier
Concept Identifier
Geospatial Relation Identifier
Geospatial Relation Dictionary
Annotated Dependency Parse Tree
Property Identifier
Annotated Dependency Parse Tree
Query Generator
GeoSPARQL/ SPARQL Query Templates
GeoSPARQL/ SPARQL Query

Figure 2.1: The conceptual architecture of the GeoQA2 system

The order in which these components are called in the pipeline is important because some components use the output generated from other components to perform their task. The dependency parse tree generator must be the first in the pipeline as all the other components annotate the respective nodes of the dependency parse tree. The functionality of the concept, instance, and geospatial relation identifiers does not depend on any other component in order to perform their tasks, thus they can be called in any order in the pipeline. The property identifier uses the outputs from the concept and instance identifiers, thus it must be called only after these two components. The query generator uses

[2] To avoid being tedious, this section will only refer to YAGO2geo when referring to the union of YAGO2 and YAGO2geo would have been more appropriate. Later on, when we present the dataset GeoQuestions1089 and evaluate GeoQA2 using it, we will distinguish these two KGs since doing so will be important for our discussion.

[3] `http://pyravlos2.di.uoa.gr:8080/yago2geo`

the outputs from all the other components in order to generate queries so it is the last one in the pipeline. Below we present each one of these components in detail.

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 toolkit [26]. The dependency parse tree is produced in CoNLL-U format [29].

Concept identifier. This component identifies the *types of features (concepts)* present in the input question and maps them to the corresponding classes of the YAGO2geo ontology. These concepts are identified by the elements of the question that are tagged as nouns (NN, NNS, NNP, NNPS) by the dependency parse tree generator. Then, these elements are mapped to the ontology classes using n-grams.

Instance identifier. This component identifies the *features* (*instances*) present in the input question. These can be, for example, the country Ireland or the Corfu island or lake Loch Ness or County Mayo. The features are identified by the elements of the question that are tagged as (proper) nouns (NN, NNS, NNP) by the dependency parse tree generator. Then, these elements are mapped to YAGO2geo resources using an entity recognition and disambiguation tool.

Geospatial relation identifier. Similarly to the previous modules, this module first identifies the geospatial relations in the input question, based on the POS tags {VB, IN, VP, VBP, VBZ}, generated by the dependency parse tree. Then, it maps them (or their synonyms) to the respective spatial function of the GeoSPARQL or stSPARQL vocabulary.

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 the corresponding properties in YAGO2geo. For instance, for the question “Which village in Rhodes has the biggest population¿‘, the “population“ attribute of the type of feature “village“ is required.

Query generator. This module generates the formal query using handcrafted query patterns, templates, and the outputs of the previous modules. In particular, the query generator reformulates the annotated (by the previous components of the pipeline) dependency parse tree and parses it in traversal order. From this process, it identifies the pattern of the question and, then, the respective template. Finally, the GeoSPARQL or SPARQL queries are generated from the templates and the resources identified from the previous modules of the pipeline. If the user question does not match any of the patterns, a message is passed to the query executor that no query has been generated.

2.2 Summary

The thesis will present the benchmark GeoQuestions1089, comprising 1089 triples of geospatial questions, answers, and their respective SPARQL/GeoSPARQL queries. Notably, GeoQuestions1089 stands as the largest geospatial QA benchmark available to the research community, and its data is freely accessible[4].

Distinguishing itself from its predecessors like GeoQuestions201, GeoQuestions1089 incorporates not only simple questions but also semantically complex queries demanding a nuanced understanding of both natural language and GeoSPARQL for resolution. Additionally, it broadens the geographical scope by including inquiries about the United States and Greece. This expanded coverage introduces new challenges for QA engines, thereby

[4] https://github.com/AI-team-UoA/GeoQuestions1089

contributing to a progressive research agenda aimed at enhancing QA systems with geo-spatial capabilities.

3. RELATED WORK

3.1 Encyclopedic Question Answering Datasets

The datasets WebQuestions [1] (6K questions), SimpleQuestions [3] (100K questions), both targeting Freebase[1], were the first considerably large datasets that appeared in the literature. WebQuestions was created in a forward manner: 100K questions were randomly selected by using the Google Suggest API and, then, by manually keeping the ones that could be answered by Freebase. SimpleQuestions, on the other hand, was created in a backward manner: a set of facts from Freebase were shortlisted and, then, manually annotated with relevant questions by English speakers. In terms of structural complexity, both datasets were simple, containing only factoid questions i.e., questions with a unique answer that can be derived from a single fact (triple) in Freebase. In 2016, WebQuestionsSP [54] (5K questions) was generated from WebQuestions, by providing SPARQL queries for the questions that the annotators could fully process to find the answers (SP stands for Semantic Parsing). Then, WebQuestionsSP was used to generate the benchmark ComplexWebQuestions [42] (35K questions) by sampling question-query pairs and automatically creating more complex SPARQL queries. From these queries, a set of questions was generated automatically by using 687 templates, and, then, the resulting questions were manually reformulated. ComplexWebQuestions contains composition questions, superlatives, and comparatives.

Three other significant benchmarks that contain complex questions are the LC-QuAD [45], LC-QuAD 2.0 [46] and QALD-9 [47] datasets. The LC-QuAD dataset (5K questions) targets DBpedia. Similarly to SimpleQuestions, it was created in a backward manner: the queries were generated semi-automatically by extracting sub-graphs containing triples within a 2-hop distance from a seed entity. The generation of the questions was facilitated automatically, using templates, and, then, refined manually. LC-QuAD was later extended to form LC-QuAD 2.0 (30K questions), which contains questions, their paraphrases, and their corresponding SPARQL queries. QALD-9 was generated manually as part of the latest QALD challenge[2]. It targets DBpedia 2016-10 and contains 558 manually created questions with counts, superlatives, comparatives, and temporal aggregators. The questions are available in 11 different languages and each question is annotated with a manually specified SPARQL query and its output.

[4] presents KQA-Pro, a dataset for complex knowledge base question answering including around 120K natural language questions. In KQA-Pro, the Knowledge-oriented Programming Language (KoPL) was defined to describe the reasoning process for solving complex questions. A KoPL program is composed of symbolic functions which define the basic, atomic operations on a KG. In this way the KoPL provides a more explicit reasoning process, making human understanding easier, by dividing the question into multiple steps. The questions of the dataset are paired with both KoPL programs and SPARQL queries. The dataset was produced by following the pipeline of [52]. First, a large number of $\langle canonical\_questions,\ KoPL,\ SPARQL \rangle$ triples was synthesized, and then the canonical questions were paraphrased to natural language questions via crowd-sourcing. The knowledge graph was built by taking the entities of FB15k-237 [44] as seeds and aligning them with Wikidata via Freebase IDs. For the question generation, there are 2 stages: 1) the locating stage where a single entity or an entity set with various restrictions

[1] SimpleQuestions were later reformulated to target also Wikidata [7].

[2] http://qald.aksw.org/

are described, and 2) the asking stage where specific information about the target entity is queried. For every stage there are some pre-defined strategies and each strategy is paired with a specific template. The last step of this pipeline is paraphrasing manually the question. This task was assigned to workers in Amazon Mechanical Turk.

[17] proposed ParaQA, a question-answering dataset with paraphrase responses for single-turn conversations. ParaQA contains 5000 question-answer pairs with a minimum of two and a maximum of eight unique paraphrased responses. ParaQA was built using a semi-automated framework for generating multiple paraphrase responses for each question using back-translation. The dataset generation workflow consists of 6 modules. The framework as input requires at least one available verbalized answer per question. To cover this need, VQuAnDa [18] is used as the first step to generate the initial responses. Then, 3 modules are used to provide new verbalized sentences: 1) a named entity recognition model is used to classify named entities into predefined categories, for instance, persons, organizations, locations, etc., and replace them with different words such as “the organization”, “the person”, “the country” etc. 2) a gender identification module which replaces the question entity with their corresponding pronouns e.g., “he, she, him, her”. 3) a verbalization template that interchanges the head and tail triple information to generate more diverse responses. After assembling sufficient answers for each question, a back-translation [34] transformer-based model is used to paraphrase the given answer. The last step is to rectify and rephrase the answers to sound more natural and fluent which is done through a peer-review process. In this way, the answers’ grammatical correctness is ensured.

[36] proposed a machine reading comprehension system for question answering over documents about climate change, as well as a climate change dataset CCMRC. Climate Bot applies machine reading comprehension over climate change documents to expand the benefits of question-answering interfaces to this area. CCMRC dataset is a manually annotated, publicly available resource for training question answering and machine reading comprehension applications, having 21k question-answer pairs and 7.400 paragraphs, extracted from trusted data sources. The climate-bot consists of a retriever which is a dense passage retriever [20] capable to retrieve documents relevant to the user’s question, a reader which is an ALBERT [21] model responsible for extracting the text span from the document that answers the user question, and a user interface where the user can ask questions and receive the most relevant documents along with highlights of the answer to the question inside the document. The CCMRC climate change dataset was formed by taking documents from various data sources and asking Amazon Mechanical Turk workers to manually write questions and highlight the corresponding answer inside that document.

[16] studies the generalizability of question answering over KGs. This kind of generalizability was introduced by [10] where three levels of generalization were defined for KGQA. The first type of generalization is independent and identically distributed (i.i.d.) where the question follows schema items that have been seen before in training data. The second type is the compositional generalization consisting of compositions of schema items seen in training data. Lastly, the zero-shot generalization consists of schema items and even domains not seen before inside the training dataset. The author investigates existing KGQA datasets and their ability to generalize. To achieve the previous task, a novel method was developed for cost-effective re-splitting datasets to effectively train the generalization ability of KGQA systems. To evaluate the generalization of a dataset, three different subsets have to be created for testing i.i.d., compositional and zero-shot generalization. This can be done by sampling candidate questions for each level of generalization in a des-

cending way, from the highest level (zero-shot) to the lowest level (i.i.d.). TeBaQA [50], BART [22] and HGNet [5] models were used to examine 25 popular KGQA datasets and their generalization ability and the results verified the assumption that KGQA datasets are not sufficient to train KGQA systems for higher levels of generalizability.

3.2 Leaderboards

[30] presents an extensive evaluation analysis of the state of the research in KGQA. 100 papers and 98 systems were evaluated on 4 datasets focusing on LC-QUAD [45] and QALD [48] series. A central and open leaderboard was proposed for KGQA benchmark datasets as a focal point for the community along with an up-to-date overview of all available demos or Web services for KGQA. The analysis shows that the evaluations presented in the papers were overwhelmingly coherent and lack of open-source implementations.

[49] introduced GERBIL QA a novel benchmarking platform for QA systems. This platform relies on the foundations of GERBIL [35] framework for benchmarking named entity recognition and entity linking systems. GERBIL QA offers 8 metrics for benchmarking QA systems as well as 6 novel sub-experiment types. There were 6 existing QA systems integrated into the platform as well as 22 QA datasets to evaluate those systems (QALD-1 to QALD-6 and NLQ).

3.3 Geospatial Question Answering Datasets

All aforementioned datasets contain encyclopedic questions, while some of them contain also geospatial questions. However, as the datasets are too large, NLP techniques are required to extract them from the full dataset, and, then, check manually that the extracted questions are indeed geospatial (for instance, the question “when was Washington elected” may, falsely, be identified as a geospatial question as Washington is both a state and a person), which is a considerably time-consuming process. A dataset marginally relevant to the geospatial domain is POIReviewQA[3] (20K questions), which contains questions about POIs. It was created by retrieving questions from the “Ask the community” service of Yelp business pages[4] and, then, by manually filtering the ones for which answers are identified in the respective reviews. The dataset contains only pairs of questions with their answers.

Currently, the only datasets focusing on the geospatial domain are GeoQuery [43], GeoAnQu [53] and GeoQuestions201 [33]. GeoQuery [43] contains 880 handcrafted factoid questions about the U.S. Geography in natural language paired with the corresponding queries in a formal query language (Prolog). GeoAnQu is a corpus of 429 geo-analytic complex and non-factoid questions manually extracted from research papers containing GIS analysis, and GIScience textbooks.

GeoQuestions201, created for the evaluation of GeoQA [33] by the UoA group participating in DA4DTE, targets the linked geospatial dataset built from DBpedia and the parts of the datasets GADM and OSM restricted to the United Kingdom and the Republic of Ireland. It contains 201 manually crafted factoid, simple and complex questions, with the respective stSPARQL/GeoSPARQL/SPARQL queries and answers. GeoQuestions201 was used

[3] http://stko.geog.ucsb.edu/poireviewqa/

[4] https://blog.yelp.com/news/qa/

as basis for the construction of GeoQuestions733. It has also been used to evaluate the engines proposed by hamzei2021place,DBLP:conf/www/HamzeiT022 and by [23].

The benchmark GeoQuestions733 [5] was more recently developed by our group too. It is the largest QA benchmark focusing on the geospatial domain over a specific KG (YAGO2geo), that contains factoid, simple and complex questions that may contain aggregates and superlatives. Additionally, it is the only one that contains, also, the respective queries and answers. For these reasons, its development is vital for the development and evaluation of geospatial QA systems and we expect that it will be eagerly taken up by other researchers.

3.4 Temporal Question Answering Datasets

[6] presents Event-QA, a dataset that contains 1000 semantic queries and the corresponding English, German and Portuguese verbalizations for Event-KG, an event-centric KG with more than 970 thousand events. What makes this dataset unique compared with the others is the focus on temporal expressions. The authors proposed an approach for automatically generating an event-centric QA dataset containing complex and diverse queries given a KG. For each query to be generated, the Event-QA pipeline includes the following steps: 1) random query type selection (i.e., ASK, SELECT, or COUNT), 2) Event extraction by selecting an event node from the KG at random, 3) Seed relation selection from the list of all relations involving the previously selected event, 4) Query graph generation in the form of a sub-graph of the knowledge graph, 5) Semantic query generation combining query type, query graph and optionally temporal constrains, and 6) query verbalization for each SPARQL query (manual annotations).

[15] proposed a benchmark called TempQuestions consisting of 1271 temporal questions with gold-standard answers. The authors define temporal questions as questions that contains a temporal expression, a temporal signal, or whose answer is of temporal nature. To create the temporal-questions-only benchmark, temporal questions from existing KG-QA datasets (Free917, WebQuestions, ComplexQuestions) were collected. The creation method followed a two-stage strategy. First, a combination of existing taggers (SUTime, HeidelTime, Standford CoreNLP), dictionaries, and lexicosyntactic patterns were used for automatic temporal question detection on the datasets. Then, manual inspection took place by a human expert who went over each question to remove non-temporal questions. In addition, the human expert verified whether existing gold answers were incorrect or noisy.

[38] introduced CRONQuestions a large Temporal KG-QA dataset that consists of both temporal KG and accompanying natural language questions requiring temporal reasoning. The temporal KG was constructed by taking all facts with temporal annotations from the WikiData subset. The final KG was formed by filtering out some instances, adding some important history events and converting timestamps to years. The authors ended up with a KG of 328k facts, 125k entities and 203 relations. To generate the QA dataset, a set of templates was created for temporal reasoning. At first, 30 templates were created and then by using annotators and the monolingual paraphraser [13] they produced 654 templates. Those templates then were filled up with entities from WikiData to generate automatically 410k unique question-answer pairs.

[5] https://figshare.com/s/3fc3e8c04c0c2bdeb584

Table 3.1: QA benchmark comparison overview

QA Benchmarks	Questions	Domain	Question Types[6]	Knowledge Base	Formal Language	Answers	Para-phrases	Generation Method
WebQuestionsSP [1]	4,737	Encyclopedic	Simple Factoid	Freebase	SPARQL	✓	✗	Manually
ComplexWeb Questions [42]	34,689		Complex	Freebase	SPARQL	✓	✗	Semi-automatically
QALD-9 [47]	408		Complex Non-factoid	DBpedia	SPARQL	✓	✗	Manually
KQA-Pro [4]	120,000		Simple Complex	Freebase Wikidata	SPARQL KoPL	✓	✓	Semi-automatically
ParaQA [17]	5,000		Factoid Non-factoid Complex	Wikidata DBpedia	SPARQL	✓	✓	Semi-automatically
CCMRC [36]	21,000		Paragraph-based Questions	Trusted climate-based data sources	-	✓	✗	Manually
LC-QuAD 2.0 [46]	30,000		Factoid Non-factoid Complex	Wikidata DBpedia	SPARQL	✗	✓	Semi-automatically
GeoQuery [43]	880	Geospatial	Simple Factoid	Geoquery Database	Prolog	✗	✗	Manually
GeoAnQu [53]	429		Non-factoid Complex	na	-	✗	✗	Manually
GeoQuestions201 [32]	201		Factoid Complex	GADM OSM DBpedia	SPARQL stSPARQL GeoSPARQL	✓	✗	Manually
GeoQuestions733	733		Factoid Complex	YAGO2geo	SPARQL stSPARQL GeoSPARQL	✓	✓	Manually
Event-QA [6]	3,000	Temporal	Multilingual Simple Complex	EventKG	SPARQL	✓	✗	Automatically-Manual translations
TempQuestions [15]	1,271		Simple Complex	Freebase	SPARQL	✓	✗	Semi-automatically
CRONQuestions [38]	410,000		Simple Complex	Wikidata	-	✓	✓	Semi-automatically

3.5 Complex Sequential QA Benchmarks

[37] introduced the Complex Sequential QA (CSQA) dataset containing 200k dialogs with a total of 1.6M turns. Unlike existing QA datasets which contains simple questions that can be answered from a single KG triple, the questions in CSQA dataset require larger subgraphs of the KG. The dataset was created through a semi-automatic process involving in-house and crowdsource workers. First annotators were asked to come up with questions that can be answered from a single tuple in the knowledge graph. Then, based on the initial pilot, workers on Amazon Mechanical Turk created subject and object based questions for each of the relation in KG. The next step was to identify types of questions which require logical, comparative and quantitative reasoning over a subgraph in the KG. For each one of these types of questions, several templates were identified in order to modify the simple question and create logical, comparative and quantitative questions. In-house annotators created templates for converting simple or complex questions to conversational questions. The authors also proposed a model for CSQA task which is a cross between a state of the art hierarchical conversation model [39] and a key value based memory network model for QA [28].

3.5.1 VQA datasets

[24] introduce a large-scale, remote sensing VQA dataset named RSVQAxBEN and built from the Sentinel-2 images and land cover classes of the BigEarthNet dataset.[6] In addition to the larger number of samples, the dataset introduces new objects of interest (land cover classes) with a new form of complexity (logical formulas).

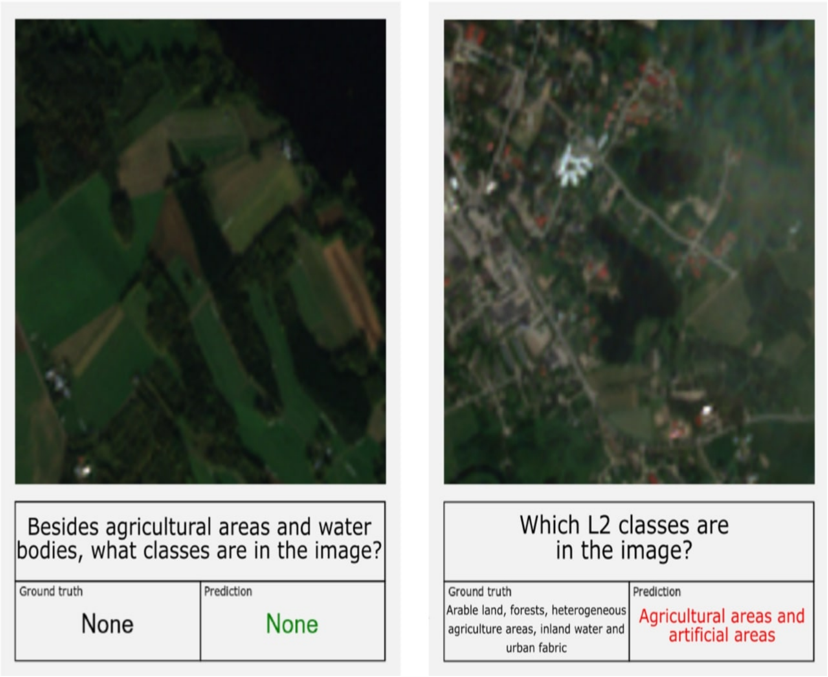
Besides agricultural areas and water bodies, what classes are in the image?
Ground truth
None
Prediction
None
Which L2 classes are in the image?
Ground truth
Arable land, forests, heterogeneous agriculture areas, inland water and urban fabric
Prediction
Agricultural areas and artificial areas

Figure 3.1: Example of type of questions (yes/no and land cover questions)

To create a baseline for the dataset, they used a VQA model that contains a feature extractor for the image (ResNet-152 pre-trained on ImageNet) and one for the question

[6] https://rsvqa.sylvainlobry.com/

(skip-thoughts architecture, pre-trained on the BookCorpus dataset). Each feature extractor produces a 1,200 dimensional feature vector, and the two vectors are then merged with a point-wise multiplication and passed to a multi-layer perceptron for the prediction of the most probable answer. The accuracy for evaluation of the dataset is defined as the ratio between the number of correct answers and the number of questions for the three classes. Specifically, it can be seen that, while the performance on yes/no questions is 79.92%, land cover questions show a poor 20.57% and global questions have a 69.83%.

BigEarthNet is a well-known remote sensing dataset developed by the TUB group participating in DA4DTE. [41] present the latest version (BigEarthNet-MM[7]) which contains 590,326 pairs of Sentinel-1 and Sentinel-2 image patches. BigEarthNet-MM makes a significant advancement for the use of deep learning in remote sensing. For example, the F_2 score obtained for the Agro-forestry areas class when transfer learning from IMAGENET is applied was 2.13% but when direct learning from BigEarthNet-MM for multi-modal multi-label image classification was applied was 71.87%. BigEarthNet-MM is suitable to assess deep learning methods for: i) learning from class-imbalanced multi-modal data (since the land cover/land use classes are not equally represented in BigEarthNet-MM); ii) transfer learning (since BigEarthNet- MM currently contains only pairs of images from a small number of European countries); and iii) also on unsupervised, self-supervised and semi-supervised multi-modal learning for information discovery from big data archives.

[7] https://bigearth.net/

4. THE GEOQUESTIONS1089 DATASET

4.1 GeoQuestions1089 dataset

The GeoQuestions1089 dataset consists of two parts, which I will refer to as GeoQuestions$_C$ (1017 questions) and GeoQuestions$_W$ (72 questions) both of which target the union of YAGO2 and YAGO2geo. GeoQuestions$_C$ is the union of the datasets GeoQuestions$_T$ and GeoQuestions$_F$.

To develop GeoQuestions$_T$, AI team asked each M.Sc. student of the 2020-2021 Knowledge Technologies course of our department to formulate 21 question-query-answers triples targeting YAGO2geo. AI team asked students to include in their questions one or more features and various kinds of geospatial relations: distance relations (e.g., near, at most 2km from), topological relations (e.g., in, borders, crosses) or cardinal directions (e.g., east of, northeast of). Also, they were asked to have questions for all four countries covered with official data by YAGO2geo: USA, Greece, United Kingdom and Ireland. Finally, one more constraint was that the generated GeoSPARQL queries for three of their questions should be with one, two and three aggregate functions, respectively. In at least one of these three cases, the students were asked to provide a question which can be mapped to an advanced GeoSPARQL expression like a nested query or a not-exists filter. In this way, we wanted to target questions that were more complex than the ones in GeoQuestions201. To obtain the answers, the students were asked to run their GeoSPARQL queries in a YAGO2geo endpoint that we provided. The questions gathered were factoid, simple/complex and, in some cases, with aggregations (e.g., counting), comparatives, or superlatives. The resulting dataset contained 615 questions targeting YAGO2geo.

To develop GeoQuestions$_F$, we asked third-year students of the 2020-2021 AI course in the same department to write 50 questions targeting the subset of OSM and the infoboxes of Wikipedia, imagining scenarios related to traveling or to generating geography questionnaires for students or TV games. The only constraint was that simple but also complex questions should be produced (examples of simple questions from GeoQuestions201 and complex questions from GeoQuestions$_T$ were given). In total, we gathered 9,335 questions. From this set, we randomly chose 1200 questions, for which we hired six M.Sc. students of the same course to clean them and translate them into SPARQL or stSPARQL/GeoSPARQL using YAGO2geo. Because this crowdsourcing effort was less restrictive than that of GeoQuestions$_T$, some questions didn't have answers in YAGO2geo alone. However, they could be answered using the union of YAGO2 and YAGO2geo KGs. After this, the students ran the queries in the YAGO2geo endpoint and stored the answers, when these existed. The resulting dataset contained 402 questions, 280 questions targeting YAGO2geo and 122 questions targeting the union of YAGO2 and YAGO2geo.

The dataset GeoQuestions$_C$ was checked by the authors of this paper. Each question (query) was checked both grammatically and syntactically, using Grammarly ([1]) and QuillBot ([2]). When necessary, and because some queries required exorbitant compute resources to be answered in reasonable time, we rerun the queries against the endpoint using materialized relations. The resulting set contained 1017 question-query-answer triples.

GeoQuestions$_W$ consists of the elements of GeoQuestions$_C$ whose questions originally

[1] https://www.grammarly.com/

[2] https://quillbot.com/

had spelling, grammar or syntax mistakes. In GeoQuestions$_W$, we include the original, incorrect questions with the end goal of benchmarking how capable QA engines are at handling incorrect input.

Extending the categorization of [32], we can see that the questions of dataset GeoQuestions1089 fall under the following categories:[3]

A. Asking for a thematic or a spatial attribute of a feature, e.g., *"Where is Loch Goil located?"*. In GeoQA2, these questions can be answered by posing a SPARQL query to YAGO2geo. Google and Bing both can also answer such questions precisely.

B. Asking whether a feature is in a geospatial relation with another feature or features, e.g., *"Is Liverpool east of Ireland?"*. The geospatial relation in this example question is a cardinal direction one (east of). Other geospatial relations in this category of questions include topological ("borders") or distance ("near" or "at most 2km from"). In GeoQA2, these questions are answered by querying YAGO2geo using the detailed geometries of features for evaluating the geospatial relation of the question. Google and Bing both cannot answer such factoid questions, but can only return a list of relevant Web pages. The recently deployed chat feature of Bing gives more information by saying that "Liverpool ... is located on the eastern side of the Irish Sea".

C. Asking for features of a given class that are in a geospatial relation with another feature. E.g., *"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 this set of questions include cardinal or distance (as in the second example question). In GeoQA2, these questions can be answered by using the detailed geometries of features from YAGO2geo for evaluating the geospatial relations. Google and Bing can also answer such questions precisely in many but not all cases (e.g., they can answer the first question but not the second).

D. Asking for features of a given class that are in a geospatial relation with any features of another class, e.g., *"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.

E. Asking for features of a given class that are in a geospatial relation with an unspecified feature of another class, and either one or both, is/are in another geospatial relation with a feature specified explicitly. E.g., *"Which churches are near a castle in Scotland?"* or *"In Greece, which beaches are near villages?"*. Google and Bing both cannot answer such questions precisely.

F. As in categories C, D and E above, plus more thematic and/or geospatial characteristics of the features expected as answers, e.g., *"Which mountains in Scotland have height more than 1000 meters?"*. Google and Bing both give links to pages with lists of mountains of Scotland with their height.

G. Questions with quantities and aggregates, e.g., *"What is the total area of lakes in Monaghan?"* or *"How many lakes are there in Monaghan?"*. Google and Bing both

[3] For comparison purposes, for each question category, we comment whether the search engines Google and Bing can answer such questions after having tried a few examples.

can answer precisely the second question but not the first. For the first question both return pages with lists of lakes in Monaghan. The chat component of Bing attempts to answer the first question but fails.

H. Questions with superlatives or comparatives, e.g., *"Which is the largest island in Greece?"* or *"Is the largest island in France larger than Crete?"*. Google answers the first question accurately but Bing does not and instead gives a list of links to related pages. The chat component of Bing can answer the first question precisely (Crete). Both engines cannot answer the second question; they only give links to relevant Web pages. The chat component of Bing is able to answer the second question precisely. (Corsica is larger than Crete).

I. Questions with quantities, aggregates, and superlatives/comparatives, e.g., *"Which city in the UK has the most hospitals?"* or *"Is the total size of lakes in Greece larger than lake Loch Lomond in Scotland?"*. Google can answer the first question precisely but Bing fails and returns a list of best hospitals in cities of the UK. Both engines cannot answer the second question.

Table 4.1 describes GeoQuestions1089 giving numbers per type of question.

4.2 Comparison to GeoQuestions201.

GeoQuestions201 contains mostly simple questions that can be answered with simple queries. For that reason, the state of the art geospatial QA engines are able to answer a significant portion of it correctly, as was shown in [12] and confirmed by our own experience while developing GeoQA2.

GeoQuestions1089 includes numerous complex questions that require both solid natural language understanding and advanced SPARQL features (nested queries, not-exists filters, arithmetic calculations) to be answered. For example: *"How many times bigger is the Republic of Ireland than Northern Ireland?"* or *"What is the population density of the municipality of Thessaloniki?"* or *"How much of the UK is woodland?"* or *"Is Belfast closer to the capital of the Republic of Ireland or the capital of Scotland?"* or *"Which islands don't have any lakes but have forests?"*. Additionally, GeoQuestions1089 is targeted on YAGO2geo, enabling easier comparison of engines that target this KG. Furthermore, because YAGO2geo also includes data about the United States and Greece, new challenges arise that must be dealt with by a good QA engine. For instance, some Greek entities lack English labels, which makes disambiguation more difficult. All in all, GeoQuestions1089 is a more varied and more challenging dataset that uses a much wider array of SPARQL functionality in its queries compared to GeoQuestions201.

Table 4.1: GeoQuestions1089 statistics

Category	KG	Count in GeoQuestions$_C$	Combined in GeoQuestions$_C$	Count in GeoQuestions$_W$	Combined in GeoQuestions$_W$
A	YAGO2geo	144	175	14	17
	YAGO2geo + YAGO2	31		3	
B	YAGO2geo	134	139	11	11
	YAGO2geo + YAGO2	5		0	
C	YAGO2geo	155	178	12	14
	YAGO2geo + YAGO2	23		2	
D	YAGO2geo	25	25	0	0
	YAGO2geo + YAGO2	0		0	
E	YAGO2geo	134	135	7	7
	YAGO2geo + YAGO2	1		0	
F	YAGO2geo	21	24	1	2
	YAGO2geo + YAGO2	3		1	
G	YAGO2geo	146	174	8	11
	YAGO2geo + YAGO2	28		3	
H	YAGO2geo	114	142	7	8
	YAGO2geo + YAGO2	28		1	
I	YAGO2geo	22	25	2	2
	YAGO2geo + YAGO2	3		0	
All	YAGO2geo	895	1017	62	72
	YAGO2geo + YAGO2	122		10	

5. EVALUATION

The dataset GeoQuestions1089 is used to benchmark the QA engines GeoQA2, the one by Hamzei et al. [12] and ChatGPT. The exact versions of the engines used are available in the repository of GeoQuestions1089. We ran the experiments on a machine with the following specifications: Intel Xeon E5-4603 v2 @2.20GHz, 128 Gb DDR3 RAM, 1.6 TB HDD (RAID-5 configuration).

5.1 Methodology and metrics.

The question answering engine that is being evaluated attempts to generate a query for each natural language question in the dataset. If the generation is successful, the query is then processed by the transpiler that rewrites the query using materialized relations, and it is then sent to a geospatial RDF store that executes the query over our knowledge graph. The result is compared to the gold result included in GeoQuestions1089. To accept an answer as correct, it must match the gold result exactly. We do not consider partially correct answers (e.g., when computed answers are a proper subset of the ones in the gold set) as correct. Likewise, we do not consider a superset of the answers in the gold set as correct. We chose to not use F-score because the correct number of returned answers/entities for each query varies greatly, which biases the metric towards certain kinds of questions.

5.2 GeoQA2

To evaluate GeoQuestions1089 in GeoQA2 we set up three Strabon endpoints. In the first two we store YAGO2 and YAGO2geo respectively. These endpoints are required by GeoQA2 to generate queries. In the third endpoint, which we use for retrieving the answers to our generated queries, we store YAGO2, YAGO2geo and its materialization.

Tables 5.1 and 5.2 show the results of the evaluation. The column “Generated Queries” gives the percentage of questions for which GeoQA2 was able to generate a query. The column “Correct Answers” gives the percentage of questions for which the query that was generated was able to retrieve the correct set of answers. Finally, the column “Correct Answers*” shows the same percentage computed over the set of questions for which a query was generated.

We observe that the complexity of the structure of the question affects significantly the performance of the system. For instance, GeoQA2 performed decently in answering rather simple questions (i.e., geospatial relation between two features), while it has difficulties in answering more structurally complex questions (i.e., questions with a combination of superlatives and quantities, questions with more sophisticated syntax or vocabulary). In addition, we see that GeoQA2 is a robust engine, meaning that it loses only a small percentage of its effectiveness when the input questions contain spelling, grammar or syntax mistakes.

Table 5.1: Evaluation of GeoQA2 over GeoQuestions$_C$.

Category	Generated Queries	Correct Answers	Correct Answers*
A	84%	47.42%	56.45%
B	76.25%	58.99%	77.35%
C	79.21%	44.38%	56.02%
D	56%	12%	21.42%
E	80%	31.85%	39.81%
F	66.66%	16.66%	25%
G	74.13%	32.18%	43.41%
H	71.12%	26.05%	36.63%
I	84%	20%	23.80%
Total	76.99%	38.54%	50.06%

Table 5.2: Evaluation of GeoQA2 over GeoQuestions$_W$.

Category	Generate Questions	Correct Answers	Correct Answers*
A	82%	47.05%	57.14%
B	81.81%	54.54%	66.66%
C	85.71%	57.14%	66.66%
D	50%	33%	66.66%
E	88%	0.00%	0.00%
F	36.36%	0.00%	0%
G	50.00%	0.00%	0.00%
H	100.00%	0.00%	0.00%
I	50%	50%	100.00%
Total	72.22%	34.72%	48.07%

5.3 Hamzei et al

In a similar vein to the evaluation of GeoQA2, the generated queries of the engine are processed by our transpiler before being sent to the Apache Jena Fuseki endpoint whose answer is compared to that included in GeoQuestions1089. To communicate with the Fuseki endpoint we use Apache Jena's own SPARQL-OVER-HTTP scripts to make sure that queries are sent and results are returned correctly. Tables 5.3 and 5.4 show the results of the evaluation.

We make three main observations. First, we see that as questions become more complex, the effectiveness of the engine drops dramatically, as was the case in our evaluation of GeoQA. The more complex the question, the less likely it is that the query generator is able to construct the proper GeoSPARQL query, with the most extreme example being questions of type I. Second, the system severely underperforms in questions of Category A, which is one of the simpler categories. This is caused by the lack of a dedicated step for named entity disambiguation. For example, if given the input question "*Where is Dublin located?*" the engine of Hamzei et al. [12] will return the location of every place named "Dublin" in the KG, instead of the location of the capital of the Republic of Ireland. This leads to an explosive increase of returned answers. Moreover, there is no mechanism for ranking the returned answers in accordance to their relevance, so even taking the first 3 answers as candidates doesn't significantly change the picture. Instead of a dedicated disambiguation step, the engine relies on the automatic resolution of disambiguation during query execution, which is an approach that works well for category B questions. In the original evaluation of their system, the authors disregarded toponym disambiguation, but we consider it a core part of question answering. Third, the system can handle spelling, grammar, and syntax mistakes without performance loss.

The main weakness of the engine of [12] is the lack of a dedicated disambiguation step. This leads to answers that contain numerous irrelevant results, i.e., the system is lacking precision. The other significant weakness is the rule-based approach to query generation that is unable to deal with complex queries.

Table 5.3: Evaluation of the system of Hamzei et al. [12] over GeoQuestions$_C$. Because the query generator of the engine was not designed to work with entities that do not have detailed geometries, we also provide statistics for the subset of questions that target YAGO2geo only.

Category	GeoQuestions$_C$			GeoQuestions$_C$ without YAGO2 Questions		
	Generated Queries	Correct Answers	Correct Answers*	Generated Queries	Correct Answers	Correct Answers*
Type-A	89.71%	10.85%	12.10%	88.88%	12.50%	14.06%
Type-B	95.68%	53.23%	55.63%	95.52%	55.22%	57.81%
Type-C	97.75%	30.33%	31.03%	97.41%	32.90%	33.77%
Type-D	100%	12%	12.00%	100%	12%	12.00%
Type-E	99.25%	7.40%	7.46%	99.25%	7.46%	7.51%
Type-F	79.16%	4.10%	5%	76.19%	4.76%	6%
Type-G	98.27%	11.49%	11.69%	97.94%	13.01%	13.28%
Type-H	97.18%	7.74%	7.97%	96.49%	7.89%	8.18%
Type-I	92%	0%	0.00%	95%	0%	0.00%
Total	95.77%	18.97%	19.81%	95.53%	20.67%	21.63%

Table 5.4: Evaluation of the system of Hamzei et al. [12] over GeoQuestions$_W$

Category	GeoQuestions$_W$		
	Generated Queries	Correct Answers	Correct Answers*
A	88.23%	17.64%	20.00%
B	100.00%	54.54%	54.54%
C	100.00%	35.71%	35.71%
D	100.00%	0.00%	0.00%
E	87.50%	0.00%	0.00%
F	90.90%	0.00%	0.00%
G	100.00%	0.00%	0.00%
H	100.00%	0.00%	0.00%
I	100.00%	0.00%	0.00%
Total	94.44%	19.44%	20.58%

5.4 ChatGPT

Finlay, in this section we would like to evaluate whether search engines like Google and Bing could answer questions like the ones of the dataset GeoQuestions1089. Given the popularity of chatbots like ChatGPT, Gemini and Copilot, it is also interesting to consider how these chatbots perform on the same task; this is what we do in this section. For our detailed study, we selected ChatGPT since it is currently the most popular of these chatbots. Our evaluation is done through the OpenAI API utilizing the GPT-3.5-turbo model.

Table 5.5 display the results of the evaluation. ChatGPT was given each question three times and the most precise answer of the three was selected manually and compared with the gold answer in GeoQuestions1089. The table column “Correctly Answered” presents the percentage of questions that are answered correctly i.e., their answer is the same as the gold one. The table column “Partially Correctly Answered” indicates the percentage of questions that are answered partially correctly i.e., the response closely approximates the correct answer, yet does not achieve full alignment with the gold standard.

During the evaluation of ChatGPT, several key findings emerged regarding its performance. ChatGPT demonstrated excellent proficiency in answering simple “yes or no” questions (e.g., the Category B question“Is Kythira within Attica?”). However, due to the inherent limitations of the OpenAI API, which cannot provide precise coordinates, the majority of the answers generated by ChatGPT for other categories of GeoQuestions1089 were only approximate. For example, for the question “Where is Kilkenny located?”, the ChatGPT answer was “Kilkenny is located in the southeast of Ireland. It is the county town of County Kilkenny and is situated on the River Nore.”. In Table 5.5, we label this answer as partially correct because it lacks precision. The correct answer in GeoQuestions1089 is the polygonal geometry of Kilkenny, something that we cannot get from ChatGPT since this chatbot does not know about geometries.

Another example is queries containing vague terms such as “near” e.g., “Which bays are near Doolin?”. In these cases ChatGPT struggles to interpret the exact criteria for proximity leading to partially correct responses. In this example, while it identifies some bays in the vicinity, not all relevant ones are included in the answer.

More importantly, when ChatGPT is faced with complex questions, particularly those requiring nuanced understanding or synthesis of information, it often falters in providing accurate responses, highlighting its limitations in handling intricate questions and underscoring the need for further refinement to enhance its performance. An example of such a complex query is “How many lakes overlap with Greek municipalities?”. For this query, ChatGPT replies “To accurately answer your question, I would need access to up-to-date geographical data regarding Greek municipalities and lakes. Unfortunately, as a text-based AI, I do not have real-time data capabilities.” Naturally, the answer to this question in GeoQuestions1089 is the exact number of such lakes. Another example of a complex question is “How many canals in England are west of villages in Camrbridgeshire?”. For this query, ChatGPT replies “To determine how many canals in England are west of villages in Cambridgeshire, I need more specific information such as the name of the villages. Could you please provide the names of the villages in Cambridgeshire that you are referring to?”. Naturally, the answer to this question in GeoQuestions1089 is the exact number of canals.

If we examine Table 5.5, we will see that as we move from simpler question categories to more complex ones, the performance of ChatGPT on the question answering task becomes worse. An exception is Category H (questions with superlatives or comparatives) which are easy to understand and ChatGPT has the required knowledge to answer them correctly.

Finally, we observed that in questions regarding population statistics, ChatGPT provided inconsistent answers in each call of the API, indicating variability in its response generation process (e.g., for the question “What is the population of Alabama?”).

Table 5.5: Evaluation of ChatGPT over GeoQuestions1089

Category	Correctly Answered	Partially Correctly Answered
Category A	15.4%	59.4%
Category B	61.4%	0%
Category C	31.3%	51.1%
Category D	23.8%	52.4%
Category E	18.8%	63%
Category F	4%	8%
Category G	6.3%	3.4%
Category H	38.7%	0%
Category I	4%	0%
Total	22.6%	26.4%

6. MY CONTRIBUTION

My input to the GeoQuestions1089 dataset is extensive and diverse. Above all, I rearranged every question in the completed version on my own initiative, making sure that each was carefully checked and arranged before being added to the dataset. To guarantee the correctness and applicability of each question, this task required close attention to detail. I was able to provide a strong basis for the dataset by supervising this procedure, which laid the framework for further research and assessment.

I also contributed significantly to the writing of the queries required to get responses from the YAGO2geo endpoint. Not only did creating these searches need in-depth knowledge of the dataset and the underlying knowledge network, but it also required proficiency with SPARQL and GeoSPARQL. Because of my experience, the questions were both effective.

In addition to writing the queries, I was in charge of executing them at the endpoint and storing the responses. Technical expertise and meticulousness were required for this undertaking to guarantee precise and seamless operation. Through the implementation of this stage, I enabled scholars and practitioners in the area to assess and analyze the dataset by providing access to its answers.

Ultimately, my participation in the assessment stage is a noteworthy component of my input to the thesis. My contribution to the benchmarking of the GeoQA2 and Hamzei et al. engines was significant. To evaluate the performance of these engines, a thorough evaluation method has to be meticulously planned and carried out. I also took responsibility for conducting all the tests and getting the responses from ChatGPT to make sure the evaluation was completed correctly and completely. My work in this area were essential in advancing the research overall by offering insightful information about the strengths and weaknesses of the individual question-answering systems.

7. CONCLUSIONS AND FUTURE WORK

I presented the dataset GeoQuestions1089 and evaluated the QA engines GeoQA2, Hamzei et al. [12] and ChatGPT using it.

AI team of NKUA and I plan to extend the dataset by utilizing semi-automatic techniques as it has been done e.g., in LC-QuAD 2.0 [9]. This will allow us to train geospatial QA engines using deep learning techniques with the hope that they will be more effective than the ones evaluated.

Additionally, we are actively engaged in advancing GeoQA by addressing several key areas: making GeoQA able to handle spatiotemporal questions, utilizing Large Language Models to improve query generation and natural language understanding, and utilizing state-of-the-art Entity Linking systems. By pursuing these avenues of research and development, we anticipate significant advancements in the effectiveness and usability of geospatial QA systems like GeoQA. Our ongoing efforts underscore our commitment to pushing the boundaries of geospatial information retrieval and enabling more intuitive and efficient access to geospatial knowledge.

ABBREVIATIONS - ACRONYMS

RDF	Resource Description Framework
SPARQL	SPARQL Protocol and RDF Query Language
OWL	Web Ontology Language
LLMs	Large Language Models

APPENDIX A. FIRST APPENDIX

BIBLIOGRAPHY

[1] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1533–1544. ACL, 2013.

[2] Mohammad Kazemi Beydokhti, Matt Duckham, Yaguang Tao, Maria Vasardani, and Amy L. Griffin. Qualitative spatial reasoning over questions (short paper). In Toru Ishikawa, Sara Irina Fabrikant, and Stephan Winter, editors, *15th International Conference on Spatial Information Theory, COSIT 2022, September 5-9, 2022, Kobe, Japan*, volume 240 of *LIPIcs*, pages 18:1–18:7. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.

[3] Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. Large-scale simple question answering with memory networks. *CoRR*, abs/1506.02075, 2015.

[4] Shulin Cao, Jiaxin Shi, Liangming Pan, Lunyiu Nie, Yutong Xiang, Lei Hou, Juanzi Li, Bin He, and Hanwang Zhang. KQA pro: A dataset with explicit compositional programs for complex question answering over knowledge base. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 6101–6119. Association for Computational Linguistics, 2022.

[5] Yongrui Chen, Huiying Li, Guilin Qi, Tianxing Wu, and Tenggou Wang. Outlining and filling: Hierarchical query graph generation for answering complex questions over knowledge graph. *CoRR*, abs/2111.00732, 2021.

[6] Tarcísio Souza Costa, Simon Gottschalk, and Elena Demidova. Event-qa: A dataset for event-centric question answering over knowledge graphs. In Mathieu d'Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux, editors, *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, pages 3157–3164. ACM, 2020.

[7] Dennis Diefenbach, Thomas Pellissier Tanon, Kamal Deep Singh, and Pierre Maret. Question Answering Benchmarks for Wikidata. In *ISWC Posters & Demos*, 2017.

[8] Alishiba Dsouza, Nicolas Tempelmeier, Ran Yu, Simon Gottschalk, and Elena Demidova. WorldKG: A world-scale geographic knowledge graph. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, pages 4475–4484. ACM, 2021.

[9] Mohnish Dubey, Debayan Banerjee, Abdelrahman Abdelkawi, and Jens Lehmann. LC-QuAD 2.0: A Large Dataset for Complex Question Answering over Wikidata and DBpedia. In *ISWC*, 2019.

[10] Yu Gu, Sue Kase, Michelle Vanni, Brian M. Sadler, Percy Liang, Xifeng Yan, and Yu Su. Beyond I.I.D.: three levels of generalization for question answering on knowledge bases. In Jure Leskovec, Marko Grobelnik, Marc Najork, Jie Tang, and Leila Zia, editors, *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 3477–3488. ACM / IW3C2, 2021.

[11] Ehsan Hamzei. *Place-related question answering: From questions to relevant answers*. PhD thesis, 2021.

[12] Ehsan Hamzei, Martin Tomko, and Stephan Winter. Translating place-related questions to GeoSPARQL queries. In *Proceedings of the Web Conference (WWW)*, 2022.

[13] J. Edward Hu, Huda Khayrallah, Ryan Culkin, Patrick Xia, Tongfei Chen, Matt Post, and Benjamin Van Durme. Improved lexically constrained decoding for translation and monolingual rewriting. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 839–850. Association for Computational Linguistics, 2019.

[14] Krzysztof Janowicz, Pascal Hitzler, Wenwen Li, Dean Rehberger, Mark Schildhauer, Rui Zhu, Cogan Shimizu, Colby K. Fisher, Ling Cai, Gengchen Mai, Joseph Zalewski, Lu Zhou, Shirly Stephen, Seila Gonzalez Estrecha, Bryce D. Mecum, Anna Lopez-Carr, Andrew Schroeder, Dave Smith, Dawn J. Wright, Sizhe Wang, Yuanyuan Tian, Zilong Liu, Meilin Shi, Anthony D'Onofrio, Zhining Gu, and Kitty Currier. Know, know where, knowwheregraph: A densely connected, cross-domain knowledge graph and geo-enrichment service stack for applications in environmental intelligence. *AI Mag.*, 43(1):30–39, 2022.

[15] Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. Tempquestions: A benchmark for temporal question answering. In Pierre-Antoine Champin, Fabien Gandon, Mounia Lalmas, and Panagiotis G. Ipeirotis, editors, *Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018*, pages 1057–1062. ACM, 2018.

[16] Longquan Jiang and Ricardo Usbeck. Knowledge graph question answering datasets and their generalizability: Are they enough for future research? In Enrique Amigó, Pablo Castells, Julio Gonzalo, Ben Carterette, J. Shane Culpepper, and Gabriella Kazai, editors, *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 3209–3218. ACM, 2022.

[17] Endri Kacupaj, Barshana Banerjee, Kuldeep Singh, and Jens Lehmann. Paraqa: A question answering dataset with paraphrase responses for single-turn conversation. In Ruben Verborgh, Katja Hose, Heiko Paulheim, Pierre-Antoine Champin, Maria Maleshkova, Óscar Corcho, Petar Ristoski, and Mehwish Alam, editors, *The Semantic Web - 18th International Conference, ESWC 2021, Virtual Event, June 6-10, 2021, Proceedings*, volume 12731 of *Lecture Notes in Computer Science*, pages 598–613. Springer, 2021.

[18] Endri Kacupaj, Hamid Zafar, Jens Lehmann, and Maria Maleshkova. Vquanda: Verbalization question answering dataset. In Andreas Harth, Sabrina Kirrane, Axel-Cyrille Ngonga Ngomo, Heiko Paulheim, Anisa Rula, Anna Lisa Gentile, Peter Haase, and Michael Cochez, editors, *The Semantic Web - 17th International Conference, ESWC 2020, Heraklion, Crete, Greece, May 31-June 4, 2020, Proceedings*, volume 12123 of *Lecture Notes in Computer Science*, pages 531–547. Springer, 2020.

[19] Nikolaos Karalis, Georgios M. Mandilaras, and Manolis Koubarakis. Extending the YAGO2 knowledge graph with precise geospatial knowledge. In Chiara Ghidini, Olaf Hartig, Maria Maleshkova, Vojtech Svátek, Isabel F. Cruz, Aidan Hogan, Jie Song, Maxime Lefrançois, and Fabien Gandon, editors, *The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part II*, volume 11779 of *Lecture Notes in Computer Science*, pages 181–197. Springer, 2019.

[20] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6769–6781. Association for Computational Linguistics, 2020.

[21] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[22] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics, 2020.

[23] Haonan Li, Ehsan Hamzei, Ivan Majic, Hua Hua, Jochen Renz, Martin Tomko, Maria Vasardani, Stephan Winter, and Timothy Baldwin. Neural factoid geospatial question answering. *Journal of Spatial Information Science*, 23:65–90, 2021.

[24] Sylvain Lobry, Begüm Demir, and Devis Tuia. RSVQA meets bigearthnet: A new, large-scale, visual question answering dataset for remote sensing. In *IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2021, Brussels, Belgium, July 11-16, 2021*, pages 1218–1221. IEEE, 2021.

[25] Paul A. Longley, Michael F. Goodchild, David J. Maguire, and David W. Rhind. *Geographic Information Science and Systems, 4th edition*. John Wiley and Sons, 2015.

[26] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[27] Manolis Koubarakis, editor. *Geospatial data science: a hands-on approach based on geospatial technologies*. ACM Books, 2023.

[28] Alexander H. Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. Key-value memory networks for directly reading documents. In Jian Su, Xavier Carreras, and Kevin Duh, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1400–1409. The Association for Computational Linguistics, 2016.

[29] Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D. Manning, Ryan T. McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. Universal dependencies v1: A multilingual treebank collection. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *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.

[30] Aleksandr Perevalov, Xi Yan, Liubov Kovriguina, Longquan Jiang, Andreas Both, and Ricardo Usbeck. Knowledge graph question answering leaderboard: A community resource to prevent a replication crisis. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Thirteenth Language Resources and Evaluation Conference, LREC 2022, Marseille, France, 20-25 June 2022*, pages 2998–3007. European Language Resources Association, 2022.

[31] Dharmen Punjani, Markos Iliakis, Theodoros Stefou, Kuldeep Singh, Andreas Both, Manolis Koubarakis, Iosif Angelidis, Konstantina Bereta, Themis Beris, Dimitris Bilidas, Theofilos Ioannidis, Nikolaos Karalis, Christoph Lange, Despina-Athanasia Pantazi, Christos Papaloukas, and Georgios Stamoulis. Template-based question answering over linked geospatial data. *CoRR*, abs/2007.07060, 2020.

[32] Dharmen Punjani, Kuldeep Singh, Andreas Both, Manolis Koubarakis, Iosif Angelidis, Konstantina Bereta, Themis Beris, Dimitris Bilidas, Theofilos Ioannidis, Nikolaos Karalis, Christoph Lange, Despina-Athanasia Pantazi, Christos Papaloukas, and George Stamoulis. Template-based question answering over linked geospatial data. In Ross S. Purves and Christopher B. Jones, editors, *Proceedings of the 12th Workshop on Geographic Information Retrieval, GIR@SIGSPATIAL 2018, Seattle, WA, USA, November 6, 2018*, pages 7:1–7:10. ACM, 2018.

[33] Dharmen Punjani, Kuldeep Singh, Andreas Both, Manolis Koubarakis, Iosif Angelidis, Konstantina Bereta, Themis Beris, Dimitris Bilidas, Theofilos Ioannidis, Nikolaos Karalis, Christoph Lange, Despina-Athanasia Pantazi, Christos Papaloukas, and George Stamoulis. Template-based question answering over linked geospatial data. In Ross S. Purves and Christopher B. Jones, editors, *Proceedings of the 12th Workshop on Geographic Information Retrieval, GIR@SIGSPATIAL 2018, Seattle, WA, USA, November 6, 2018*, pages 7:1–7:10. ACM, 2018.

[34] Chris Quirk, Chris Brockett, and William B. Dolan. Monolingual machine translation for paraphrase generation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing , EMNLP 2004, A meeting of SIGDAT, a Special Interest Group of the ACL, held in conjunction with ACL 2004, 25-26 July 2004, Barcelona, Spain*, pages 142–149. ACL, 2004.

[35] Michael Röder, Ricardo Usbeck, and Axel-Cyrille Ngonga Ngomo. GERBIL - benchmarking named entity recognition and linking consistently. *Semantic Web*, 9(5):605–625, 2018.

[36] Md. Rashad Al Hasan Rony, Ying Zuo, Liubov Kovriguina, Roman Teucher, and Jens Lehmann. Climate bot: A machine reading comprehension system for climate change question answering. In Luc De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pages 5249–5252. ijcai.org, 2022.

[37] Amrita Saha, Vardaan Pahuja, Mitesh M. Khapra, Karthik Sankaranarayanan, and Sarath Chandar. Complex sequential question answering: Towards learning to converse over linked question answer pairs with a knowledge graph. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 705–713. AAAI Press, 2018.

[38] Apoorv Saxena, Soumen Chakrabarti, and Partha P. Talukdar. Question answering over temporal knowledge graphs. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 6663–6676. Association for Computational Linguistics, 2021.

[39] Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C. Courville, and Joelle Pineau. Building end-to-end dialogue systems using generative hierarchical neural network models. In Dale Schuurmans and Michael P. Wellman, editors, *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, pages 3776–3784. AAAI Press, 2016.

[40] Giorgos Stoilos, Nikos Papasarantopoulos, Pavlos Vougiouklis, and Patrik Bansky. Type linking for query understanding and semantic search. In Aidong Zhang and Huzefa Rangwala, editors, *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, pages 3931–3940. ACM, 2022.

[41] Gencer Sumbul, Arne de Wall, Tristan Kreuziger, Filipe Marcelino, Hugo Costa, Pedro Benevides, Mario Caetano, Begüm Demir, and Volker Markl. Bigearthnet-mm: A large scale multi-modal multi-label benchmark archive for remote sensing image classification and retrieval. *CoRR*, abs/2105.07921, 2021.

[42] Alon Talmor and Jonathan Berant. The web as a knowledge-base for answering complex questions. In Marilyn A. Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 641–651. Association for Computational Linguistics, 2018.

[43] Lappoon R. Tang and Raymond J. Mooney. Using multiple clause constructors in inductive logic programming for semantic parsing. In Luc De Raedt and Peter A. Flach, editors, *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*, pages 466–477. Springer, 2001.

[44] Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. Representing text for joint embedding of text and knowledge bases. In Lluís Màrquez, Chris Callison-Burch, Jian Su, Daniele Pighin, and Yuval Marton, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1499–1509. The Association for Computational Linguistics, 2015.

[45] Priyansh Trivedi, Gaurav Maheshwari, Mohnish Dubey, and Jens Lehmann. Lc-quad: A corpus for complex question answering over knowledge graphs. In Claudia d'Amato, Miriam Fernández, Valentina A. M. Tamma, Freddy Lécué, Philippe Cudré-Mauroux, Juan F. Sequeda, Christoph Lange, and Jeff Heflin, editors, *The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference, Vienna, Austria, October 21-25, 2017, Proceedings, Part II*, volume 10588 of *Lecture Notes in Computer Science*, pages 210–218. Springer, 2017.

[46] Priyansh Trivedi, Gaurav Maheshwari, Mohnish Dubey, and Jens Lehmann. Lc-quad: A corpus for complex question answering over knowledge graphs. In Claudia d'Amato, Miriam Fernández, Valentina A. M. Tamma, Freddy Lécué, Philippe Cudré-Mauroux, Juan F. Sequeda, Christoph Lange, and Jeff Heflin, editors, *The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference, Vienna, Austria, October 21-25, 2017, Proceedings, Part II*, volume 10588 of *Lecture Notes in Computer Science*, pages 210–218. Springer, 2017.

[47] Ricardo Usbeck, Ria Hari Gusmita, Axel-Cyrille Ngonga Ngomo, and Muhammad Saleem. 9th challenge on question answering over linked data (QALD-9) (invited paper). In Key-Sun Choi, Luis Espinosa Anke, Thierry Declerck, Dagmar Gromann, Jin-Dong Kim, Axel-Cyrille Ngonga Ngomo, Muhammad Saleem, and Ricardo Usbeck, editors, *Joint proceedings of the 4th Workshop on Semantic Deep Learning (SemDeep-4) and NLIWoD4: Natural Language Interfaces for the Web of Data (NLIWOD-4) and 9th Question Answering over Linked Data challenge (QALD-9) co-located with 17th International Semantic Web Conference (ISWC 2018), Monterey, California, United States of America, October 8th - 9th, 2018*, volume 2241 of *CEUR Workshop Proceedings*, pages 58–64. CEUR-WS.org, 2018.

[48] Ricardo Usbeck, Ria Hari Gusmita, Axel-Cyrille Ngonga Ngomo, and Muhammad Saleem. 9th challenge on question answering over linked data (QALD-9) (invited paper). In Key-Sun Choi, Luis Espinosa Anke, Thierry Declerck, Dagmar Gromann, Jin-Dong Kim, Axel-Cyrille Ngonga Ngomo, Muhammad Saleem, and Ricardo Usbeck, editors, *Joint proceedings of the 4th Workshop on Semantic Deep Learning (SemDeep-4) and NLIWoD4: Natural Language Interfaces for the Web of Data (NLIWOD-4) and 9th Question Answering over Linked Data challenge (QALD-9) co-located with 17th International Semantic Web Conference (ISWC 2018), Monterey, California, United States of America, October 8th - 9th, 2018*, volume 2241 of *CEUR Workshop Proceedings*, pages 58–64. CEUR-WS.org, 2018.

[49] Ricardo Usbeck, Michael Röder, Michael Hoffmann, Felix Conrads, Jonathan Huthmann, Axel-Cyrille Ngonga Ngomo, Christian Demmler, and Christina Unger. Benchmarking question answering systems. *Semantic Web*, 10(2):293–304, 2019.

[50] Daniel Vollmers, Rricha Jalota, Diego Moussallem, Hardik Topiwala, Axel-Cyrille Ngonga Ngomo, and Ricardo Usbeck. Knowledge graph question answering using graph-pattern isomorphism. In Mehwish Alam, Paul Groth, Victor de Boer, Tassilo Pellegrini, Harshvardhan J. Pandit, Elena Montiel-Ponsoda, Víctor Rodríguez-Doncel, Barbara McGillivray, and Albert Meroño-Peñuela, editors, *Further with Knowledge Graphs - Proceedings of the 17th International Conference on Semantic Systems, SEMANTiCS 2017, Amsterdam, The Netherlands, September 6-9, 2021*, volume 53 of *Studies on the Semantic Web*, pages 103–117. IOS Press, 2021.

[51] Denny Vrandecic and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, 2014.

[52] Yushi Wang, Jonathan Berant, and Percy Liang. Building a semantic parser overnight. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1332–1342. The Association for Computer Linguistics, 2015.

[53] H. Xu, E. Hamzei, E. Nyamsuren, H. Kruiger, S. Winter, M. Tomko, and S. Scheider. Extracting interrogative intents and concepts from geo-analytic questions. *AGILE: GIScience Series*, 1:23, 2020.

[54] Wen-tau Yih, Matthew Richardson, Christopher Meek, Ming-Wei Chang, and Jina Suh. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*. The Association for Computer Linguistics, 2016.

[55] Eman M. G. Younis, Christopher B. Jones, Vlad Tanasescu, and Alia I. Abdelmoty. Hybrid geo-spatial query methods on the semantic web with a spatially-enhanced index of DBpedia. In Ningchuan Xiao, Mei-Po Kwan, Michael F. Goodchild, and Shashi Shekhar, editors, *Geographic Information Science - 7th International Conference, GIScience 2012, Columbus, OH, USA, September 18-21, 2012. Proceedings*, volume 7478 of *Lecture Notes in Computer Science*, pages 340–353. Springer, 2012.