Ontology-based Question Answering Systems over Knowledge Bases: A Survey

Wellington Franco[1], Caio Viktor[1], Artur Oliveira[1], Gilvan Maia[1], Angelo Brayner[1], V. M. P. Vidal[1], Fernando Carvalho[1] and V. M. Pequeno[2]

[1]*Departamento de Computação, Federal University of Ceará, Fortaleza, Ceará, Brazil*

[2]*TechLab, Departamento de Ciências e Tecnologias, Universidade Autónoma de Lisboa Luís de Camões, Portugal*

wellington@crateus.ufc.br, caioviktor@alu.ufc.br, {arturoliveira, gilvanmaia}@virtual.ufc.br, {vvidal, brayner, carvalho}@lia.ufc.br, vpequeno@autonoma.pt

Keywords: Question Answering Systems, Ontology, Knowledge Bases, Literature Survey.

Abstract: Searching relevant, specific information in big data volumes is quite a challenging task. Despite the numerous strategies in the literature to tackle this problem, this task is usually carried out by resorting to a Question Answering (QA) systems. There are many ways to build a QA system, such as heuristic approaches, machine learning, and ontologies. Recent research focused their efforts on ontology-based methods since the resulting QA systems can benefit from knowledge modeling. In this paper, we present a systematic literature survey on ontology-based QA systems regarding any questions. We also detail the evaluation process carried out in these systems and discuss how each approach differs from the others in terms of the challenges faced and strategies employed. Finally, we present the most prominent research issues still open in the field.

1 INTRODUCTION

The advent of *Natural Language Interfaces* (NLI) secured the interest of *Natural Language Processing* (NLP) researchers for practical applications. In particular, *Question Answering* (QA) systems are closely related to Computational Linguistics (Herzog and Rollinger, 1991; Wilensky et al., 1988). Ontologies (Daconta et al., 2003) are fundamental *Knowledge Bases* (KBs), such as DBPedia (Bizer et al., 2009), YAGO (Suchanek et al., 2007), YAGO2 (Hoffart et al., 2013), and Freebase (Bollacker et al., 2008). KBs perform information collection on a regular time basis from open, constantly expanding resources such as Wikipedia (Hakimov et al., 2013). Many challenges surround the field of KBs, such as base coverage, commonsense, rules, and socio-cultural aspects (Weikum et al., 2019).

Regarding a QA system, consulting a KB requires a thorough mastery of elaborate, formal concepts regarding ontologies and their underlying technologies. A knowledge base QA system aims to retrieve the information requested by users in natural language, but in terms of automatic inferences or queries operating over KBs. Actual access to a KB demands queries to be written in formal, complex languages such as SPARQL (Seaborne and Prud'hommeaux, 2008; Diefenbach et al., 2018). Resorting to sophisticated user interfaces is time-consuming and also demands customization efforts for different domains and users.

In this work we present a literature survey on ontology-based QA systems that operate over Knowledge Bases. The remaining of this manuscript is organized as follows: Section 2 initially introduces the main types of QA systems. Sections 3 and 4 contain a discussion about related works and golden standards for evaluation, respectively; Section 5 is devoted to the discussion of works found in this literature survey; Section 6 presents the top research challenges identified in this survey; and the conclusions and main considerations about this investigation are developed throughout Section 7.

2 QA SYSTEM TYPES

Given the full range of topics involved in QA system, this paper focuses on an existing classification based on the type of response expected to be found (Latifi, 2018): *Closed-Domain Question Answering* (QADR), *QA for Comprehension Reading* (QACR), *Community Question Answering* (QAC), *QA Over Domain Ontologies, Ontology-Based QA* (QASOBO), and *Linked Open Data Question Answering* (QALOD).

QADR: question and search space are restricted to a particular domain, so users often show high expectations for appropriate responses, i.e., having no answer is preferable to reporting wrong answers (Weikum et al., 2019). QADR systems are usually applied to specific tasks and use lexical, terminologies, knowledge bases, ontologies, and other domain-restricted lexical-conceptual resources (Latifi, 2018).

QACR: questions are closely related to a given document, user's ability to understand the matters presented in that document can be assessed. Richardson et al. (2013) proposed MCTest[1], a machine text comprehension dataset formulated based on different aspects of 660 stories.

QAC: social QA network is based on interactions within a virtual community, such as *Quora* or *Stack Overflow*. QAC starts when a user posts an initial consultation formulated using natural language, thus triggering a line of interventions by other community members.

QASOBO: answers are not sought directly in plain, unstructured text documents, but in ontologies (Diefenbach et al., 2018), so systems can take advantage of its linguistic and terminological data, its relations, properties, and inference capabilities.

QALOD: there is a substantial growth of available resources in the Semantic Web over the past few years, regarding both quantity and complexity, including the *Linked Open Data* initiative.

3 RELATED SURVEYS

Cimiano and Minock (2009) carried out qualitative analyzes of the underlying problems and challenges regarding the construction of NLI, resulting in 11 challenges identified for QA systems, such as question types, ambiguities, spatial prepositions, modifiers and superlatives, aggregation, comparison, and negation. Freitas et al. (2012) focused in the challenges of building effective query mechanisms for large-scale data, exemplifying the differences between *Information Retrieval (IR)*, SPARQL queries, and QA systems. (Lopez et al., 2013) listed and discussed the main forms of assessment and *golden standards* applicable to QA systems, their limitations and key questions for future works. (Mishra and Jain, 2016) proposed 8 classification criteria for QA systems: application domain; question types; question analysis types; data source type; correspondence function types; characteristics of data sources; techniques used; and forms of responses generated. They also presented advantages, disadvantages, and representative systems.

(Höffner et al., 2017) coined the term *Semantic Query Answering* (SQA) in their survey: users ask questions in NL using their terminology for which receive a concise answer, generated by querying a KB *Resource Description Framework* (RDF)[2]. They also analyzed and categorized methods that address specific problems of SQA. (Diefenbach et al., 2018) published a *suvery* on SQA focusing on the *"standard architecture"* of existing systems, which subdivides the problem into four steps/modules: Question Analysis, Sentence Mapping, Disambiguation, Query Building, and Queries over Distributed Knowledge Bases. (Soares and Parreiras, 2018) presented a systematic literature review of studies published from 2000 to 2017. They focused on identifying QA techniques and tools with particular attention to the relationship between QA systems and NLP. A similar study was published by (Tasar et al., 2018) to identify and analyze the main methods, datasets, and venues of works published between 2010 and 2017 for QALOD. From an initial universe of 843 articles, the authors selected 53 studies as primary studies from which methods were analyzed and gaps between approaches were identified. (Wohlgenannt et al., 2019) also analyzed frameworks for QALOD, but comparing visual diagrammatic approaches to querying data with existing NL-based systems. Using the (QALD7) dataset, they assessed that visual methods iteratively until the answer is found and showed some benefits in relation to 4 NL-based systems, such as better data exploration and better performance.

Most systematic reviews on SQA research focus on evaluation (Lopez et al., 2013), specific domains (Freitas et al., 2012; Soares and Parreiras, 2018), categorization (Mishra and Jain, 2016; Höffner et al., 2017) or general approaches (Diefenbach et al., 2018) and techniques (Soares and Parreiras, 2018). Our research aims to analyze the SQA works dealing with both specific and general domains under the light of how ontologies can be employed to assist or improve QA systems.

[1]http://research.microsoft.com/mct

[2]Definition made by (Hirschman and Gaizauskas, 2001)

4 GOLDEN STANDARDS AND EVALUATION

There is an interesting diversity of standard data sets available to evaluate QA systems. Such *Golden Standards* (GSs) establish reference problems for evaluation. Given the enormity of the datasets, typical evaluations using a GS include manual checks on reduced scenarios to ensure the quality of the data and results. These are main GS referenced in literature: *Question Answering over Linked Data (QALD)* (Cimiano et al., 2013; Cimiano and Minock, 2009), *WebQuestions* (Berant et al., 2013), *Stanford Question Answering Dataset (SQuAD)* (Rajpurkar et al., 2016), and *SimpleQuestions* (Bordes et al., 2015).

QALD's questions are prepared for an annual challenge held at CLEF (*Conference and Labs of the Evaluation Forum*), ESWC (European Semantic Web Conference), and ISWC (International Semantic Web Conference) conferences. Questions are usually answered using up to three binary relationships, but typically resorting to modifiers such as *order by* and *count*. The training set is qualitative but quite small, with about 50 to 250 questions, leaves by itself little to no space for supervised approaches such as deep learning (LeCun et al., 2015). WebQuestions contains about 5,810 questions drawn from Freebase (Bollacker et al., 2008). Of these, about 97% can be answered using a single reified statement with potentially few constraints (type, temporal, etc).

SQuAD addresses reading comprehension and contains 100,000 questions about articles from Wikipedia collected by Amazon Mechanical Turk (AMT) (Paolacci et al., 2010). SQuAD' questions were validated using DBPedia (Bizer et al., 2009), so each answer is text passage or an extension of the corresponding reading passage. However, some questions are impossible to answer. Finally, SimpleQuestions contains 108,442 questions built from Freebase that, due to their factual nature, can be answered using a binary relation. (Petrochuk and Zettlemoyer, 2018) showed that about 33.9% of these questions are unanswerable due to problems concerning the nature of the underlying data.

$$\mathrm{Acc(Q)} = \frac{\text{\# correct answers for Q}}{\text{\# questions in Q}} \qquad (1)$$

The assessment of QA systems is usually performed based on the following metrics: accuracy; *recall*, also known as sensitivity; and *F-score*. *Accuracy* (*Acc*) is most widely used since it indicates the fraction of questions that are answered correctly for a set Q (Equation 1). *recall* (*Rec*) is similar to accuracy, but only considers the subset of the answerable questions and does not consider wrong, irrelevant responses. *F-score* considers both false cases as a combination of precision and *recall*. F_1 is typically used for small data sets that usually have unbalanced classes.

5 COMPARING QA APPROACHES

We adopted the following search criteria to select works discussed in this paper: an initial search was carried out for QA works that resort to ontologies in the process of building their strategy for solving the problem; both open-domain QA and closed domain systems were contemplated in this search; finally, we intend to find material that allows us to discuss the main challenges and point out future works on QA systems built on top of ontologies.

***Ontology Natural Language Interaction*, ONLI$^+$** (Mithun et al., 2007)**:** NLP is used as *front-end* for the Racer reasoner (Haarslev and Möller, 2001) and nRQL *new Racer Query Language* (Haarslev et al., 2004), which augments and extends the functional API to query a knowledge base using tuples in using assertions from Description Logic (Baader et al., 2003), so the system can retrieve all individuals matching a query concept. Experimental evaluation shows that ONLI$^+$ does not lose performance in terms of transforming NL questions into nRQL queries while its approach increases users' expressiveness. ***Portable nAtural laNguage inTerface to Ontologies*, PANTO** (Wang et al., 2007)**:** accepts form inputs in NL to generate SPARQL queries by performing a mapping between an ontology's concepts, instances, and relations and NL using a syntactical analysis tree built by the Stanford Parser and Stanford CoreNLP (De Marneffe and Manning, 2008). PANTO also uses WordNet (Miller, 1995) and string similarity (Cohen et al., 2003) to increase mapping quality. The result is converted to the query triple form in OntoTriples, i.e., *Ontology Triples* compatible with some ontology statements in the form of $<$*subject, predicate, object*$>$ and represented as entities. Finally, OntoTriples are interpreted as SPARQL. The main problem with PANTO is the heavy reliance on the Stanford Parser.

AquaLog (Lopez et al., 2007)**:** a portable system that receives queries expressed in NL and ontology as input, extracting responses from one or more KBs while it learns user jargon to improve experience over time. Two language models are used to (a) convert NL queries into query triple format and (2) to transform query triple to triple in ontology format. The underlying data model consists of RDF triples. AquaLog has poor performance relative to its ability to answer complex questions.

***Question Answering System Applied to the Cinema Domain*, QACID** (Ferrández et al., 2009)**:** its main component are ontology, data, a dictionary, user query collections, and the linking mechanism. QACID depends on manually constructed ontologies and does not feature a partial matching strategy for dealing with cases in which no exact match is found. QACID was tested in the Spanish language using ontological modeling for the Cinema domain. QACID heavily relies on the domain, so its coverage is limited both in terms of answerable questions and spatiotemporal parameters.

***Question-based Interface to Ontologies*, QuestIO** (Tablan et al., 2008)**:** an NLI to access structured, domain-independent information, with no training step required. QuestIO automatically converts short conceptual NL queries into formal queries that can be executed over virtually any semantic repository. This approach is efficient for small, domain-specific ontologies, and it also performs relatively well for poorly formed questions. However, QuestIO may partially or even not answer complex questions, lacks user interaction for improving results, and has no handling capable of resolving the ambiguity of searches using keywords.

***Feedback, Refinement and Extended Vocabulary Aggregation*, FREyA** (Damljanovic et al., 2010)**:** improves QuestIO concerning a deeper understanding of the semantic meaning of questions to better deal with ambiguity when ontologies are spanning multiple domains. FREyA allows users to enter any queries and resorts to a parse tree for identifying the type of question answers that are capable of providing more concise answers. Similar to AquaLog, FREyA uses user feedback to improve performance over time.

***Question Answering System for YAGO Ontology*, QASYO** (Hogan et al., 2011)**:** integrates NLP, ontologies, and information retrieval technologies into approach at one sentence-level in four steps, i.e., question classification, a language component, query generator, and query processor. QASYO provides answers extracted from the available semantic markings for queries expressed in NL and YAGO ontology (Suchanek et al., 2007, 2008). Semantic analysis extracts keywords from questions for both use in queries and to detect the expected response type. However, QASYO strongly tied to YAGO and the use of keyword severely limits expressiveness of queries to simple ones.

***Ontology-based Question Answering on the Semantic Web*, Pythia** (Unger and Cimiano, 2011)**:** builds compositional representations of meaning using a vocabulary alignment for an ontology based on deep linguistic analysis, which allows the construction of formal queries even for complex NL questions involving quantification and superlatives. Pythia can translate representations of meaning into formal queries by resorting to a grammar, and also uses an interface to perform a lexical-ontological specification that explains possible linguistic links of ontology concepts. However, Pythia is limited to small knowledge bases and cannot be used as a general solution.

***Deep Web Extraction for Question Answering*, DEQA** (Lehmann et al., 2012)**:** is a conceptual framework that combines semantic technologies with effective data extraction. DEQA performs web data extraction for real estate site offerings, where there is no structured user interface for the users, given the case of all Oxford real estate agencies. Data is integrated to link extracted data with prior knowledge, such as geospatial information about relevant points of interest. Then, DEQA maps NL questions into SPARQL patterns, which are quite limited as the coverage of questions, especially the complex ones. Moreover, DEQA does not support complex operators such as "*less than*" and "*greater than*". **QAAL** (Kalaivani and Duraiswamy, 2012): results from a comparison of different input types, query processing methods, and the input and output formats of various systems. The authors also analyzed and discussed different performance metrics in addition to their limitations. QAAL uses a graph matching algorithm to associate the query with the answer (Collins and Loftus, 1975), thus improving its results in terms of generated SPAQRL queries. Although QAAL uses NLP for improving its accuracy, the adoption of keywords limits the search scope, which detracts QAAL's applicability for complex questions. Also, QUAAL is limited to closed domain and does not handle ambiguity.

***Question Answering wiKiframework-based System*, QAKiS** (Cabrio et al., 2012)**:** a QALOD that addresses the problem of question interpretation as correspondence with relationship-based ontology, in which the fragments of the question correspond to the binary relations in the ontology. QAKiS first tries to match fragments with textual templates automatically collected from Wikipedia. QAKiS's relationship-based mapping for question interpretation allows to convert user questions into a query language, e.g., SPARQL. However, QAKiS is heavily tied to Wikipedia and DBpedia. **PARALEX** (Fader et al., 2013)**:** an interactive open domain QA system that maps questions into simple queries over extractions made by an open information extraction system (Banko et al., 2007). PARALEX executed on an extracted KB and used sentences extracted from

WikiAnswers to learn a query function for carrying out queries over a KB.

SINA (Shekarpour et al., 2015)**:** a scalable search system for answering NL queries by transforming user-supplied keywords or queries into SPARQL queries over a set of interconnected data sources. Internally, SINA uses a Markov Chain Model to determine, from different datasets, which resources are best suited to respond to each query. The key advantage of this approach is its independency from ontology data schema: SINA generates a set of predefined templates that scale to large knowledge bases in an easy-to-use manner. **DEANNA** (Yahya et al., 2012, 2013)**:** as far as it was possible to investigate, the only QA system that approaches QALD from a formal Integer Linear Programming (ILP) perspective. DEANNA simultaneously addresses the problems of question decomposition and disambiguation using an optimization model that combines phrase selection and mapping into semantic targets. Constraints ensure that sentences are selected to preserve their phrasal dependencies in the mapping image for semantic destinations. However, there is no clear evidence that this promising approach can be easily adapted to other datasets or domains.

Hybrid Deep Relation Extraction for Question Answering on Freebase, HybQA (Mohamed et al., 2017)**:** proposes a QA strategy focused on extracting the relationship in a hybrid way over the Freebase dataset, which consists of using state-of-the-art deep neural networks to capture the type of relationship between a question and the expected answer. This relationship is verified using Wikipedia to choose the best relationship. Evaluation using HybQA over the WebQuestions dataset showed an improvement over existing models in terms of accuracy, which is 57%.

***A Semantic-based Closed and Open Domain Question Answering System*, ScoQAS** (Latifi et al., 2017; Latifi, 2018)**:** this system proposed a hybrid approach, i.e., handles factoid questions for both open and closed domains. ScoQAS adopt QALD as a golden standard for evaluation purposes. The main differential of ScoQAS is using a set of graph inferences for the closed domain. On the other hand, as its main limitation, ScoQAS does not address complex questions.

Figure 1 compares strategies in terms of accuracy. Initial approaches were domain-specific, so these adopt a well-formed *schema* for a better accuracy when compared to the latest approaches proposed for open domain. ONLI (Mithun et al., 2007), QAAL (Kalaivani and Duraiswamy, 2012), and PARALEX (Fader et al., 2013) do not adopt accuracy.

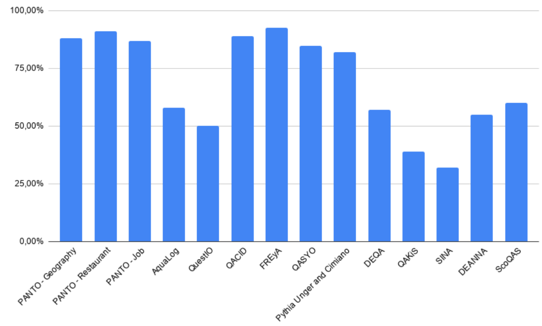

Figure 1: Accuracy for each approach found in this review.

6 RESEARCH CHALLENGES

Evidence found in the literature show that, whilst domain-specific QA can be addressed with enough manual effort, there is no single system capable of answering complex questions regarding multiple domains (see Table 1). Due to the breadth of the studies and because they are recently published works, our analysis of the articles shows that the following research challenges are still open (Höffner et al., 2017; Diefenbach et al., 2018).

Lexical Gap (Hakimov et al., 2015). Despite the advances experienced in the field, the same meaning can still be expressed in many different ways using natural language. Queries and KBs many times are not built on top of the same vocabulary, i.e., using synonyms or the same abstraction level. The lexical gap problem still occurs when a vocabulary used in the question differs from that used when labeling KBs, making it difficult for QA systems to find a link or association between a question and its answer. This seems to be a promising subject for future work in the area, as tackling this challenge has resulted in a significant improvement on the results reported by QA systems (Petrochuk and Zettlemoyer, 2018).

Queries over Multiple Knowledge Bases. If the information that is referenced in a query is represented by distributed RDF resources, then the corresponding information need for formulating a proper answer can be found in multiple bases. Performing a combination of knowledge bases demands both schema-level and entity-level matching to merge partial results or translations between databases. This is required to find dataset entities that are semantically equivalent to a single, “global” entity during the query execution.

Multilingual QA. The knowledge found in the Web and information systems is written in multiple languages. Although RDF resources can be described multilingually by adopting language *tags*, there is no single language that is always used in Web docu-

Table 1: Summary of works found in the literature about ontology-based QA systems.

ID	**QA Systems**	**Characteristics**	**Limitations**	**Dataset Adopted**	**Evaluation**
T001	ONLI Mithun et al. (2007)	Queryes in NL with nRQL. Syntactical Analysis. Ontology Mapping. Query interface for the Racer system.	Domain-dependent. Limited analyzed question types.	Own (30 question and respective nRQL query pairs for two genome-related onologies)	MRR (Mean Reciprocal Rank): 0.35
T002	PANTO Wang et al. (2007)	Uses WordNet and string similarity measures in ontology mapping algorithms. NL question as input and outputs a SPARQL query. QueryTriples as intermediary representation. Converts queries into triples in OntoTriples.	Scalability: only works for small ontologies. Does not use data indexing techniques. Constraints the query scope. Weak user interaction.	Mooney (http://www.cs.utexas.edu/users/ml/nldata.html) Geographic basis, restaurant, and work.	Accuracy: 88.05%, 90.87%, 86.12% Accuracy: 85.86%, 96.64%, 89.17%
T003	AquaLog Lopez et al. (2007)	Independent domain grammar. Queries in LN. Uses string similarity algorithms. Uses GATE and WordNet platform. Ontology-based relationship similarity Triple Response Service (RSS).	Lack of adequate reasoning services defined by ontology. Does not understand queries in the format “How much”. Does not explore scope quantifiers: (“each”, “all”, and “some”)	Own base built with 69 pairs of questions and answers. Closed domain.	Accuracy: 58%
T004	QuestIO Tablan et al. (2008)	Open domain application. Translates NL and keywords into SPARQL query by means of linguistic analysis. Ontologic dictionary search. Iterative Transformation until a SeRQL query is obtained.	Lacks user interaction. Session-based. Cannot solve ambiguity from terms of keywords.	36 specific-domain questions.	Accuracy: 50%
T005	QACID Ferrández et al. (2009)	Tested for Spanish over the cinema domain. Query set build by means of *clusters*. Query mapping between NL and knowledge bases using distance metrics,	Expensive due to the domain dependency. Can only be applied with limited coverage. Leaks conscient capacities of temporal and spatial context.	Closed domain. 162 question pairs.	Accuracy: 89%
T006	FREyA Damljanovic et al. (2010)	Open domain. Identification and verification of ontology concepts. SPARQL query generation. Question type identification. Reinforcement learning for improving the suggestion rank. Session-based interaction.	Requires tests with large datasets. Evaluation is not user-centered.	250 question from Mooney Geoquery	Accuracy: 92.4% MRR: 78%
T007	QASYO Hogan et al. (2011)	Queries in NL. YAGO as input. The LN query is translated into a set of intermediate and triple-base representations, query *traps*. Translates into ontology-compatible triples.	Lacks information about the nature and complexity of possibly necessary changes in the ontology and the linguistic component.	-	Accuracy: 84,7%
T008	Pythia Unger and Cimiano (2011)	Handles with a wide range of linguistically complex questions involving quantifiers, numerals, comparisons, superlatives, and negation. Correctly maps NL terms into the corresponding ontology concepts, despite these are superficially different. The domain-specific lexical is built automatically from an specification of linguistic realizations of the concept ontology.	Portability (requires a new LexInfo model for a new domain to be built); Requires not negligible effort for larger domains (DBpedia, for example)	880 questions from Mooney Geoquery	Accuracy: 82%
T009	DEQA Lehmann et al. (2012)	Application for the Web of Data. Uses the TBSL algorithm. Compreensive deep Web QA system. Web extractions using OXPath. Usses LIMES to compute complex links for specification.	Needs to cover more question types. Does not support complex operators. Does not support multiple languages.	100 domain-specific questions	Accuracy: 57%
T010	QAAL Kalaivani and Duraiswamy (2012)	Uses concept graph matching. Adopts SPARQL. Uses NLP for QA analysis. Diffuse activation algorithm.	Normal keyword search model. Cannot answer to complex questions in ambiguous cases. Closed domain.		Mean Accuracy Distribution
T011	QAKiS Cabrio et al. (2012)	Open domain. QA structured over the knowledge base. Relevant information in unstructured form.	Cannot handle boolean and n-related questions. Cannot perform Procedural, Temporal or Spatial analyzes.	QALD-2 (DBPedia).	Accuracy: 39%
T012	PARALEX Fader et al. (2013)	Open domain. Transforms text into tuple. Learning oriented to paragraph interpreting questions. No manual model must be created.	Not able to work with complex questions. Lack of “answerability”.	WebQuestions (2,032). TREC (517). WikiAnswers (7,310).	Accuracy: 77%
T013	SINA Shekarpour et al. (2015)	Open domain. Tries to reduce the lexical gap. Uses Hidden Markov Chain. *Template-based*.	Cannot perform Procedural, Temporal or Spatial analyzes. Does not support complex operations. Does not support multiple languages.	QALD-3.	Accuracy: 32% MRR: 0.8
T014	DEANNA Yahya et al. (2012, 2013)	Integer Linear Programming (ILP) approach. Query extension for SPO and SPOX. Query relaxation when no results are found.	Cannot perform Procedural, Temporal, or Spatial analyzes. Does not support complex operations. Does not support multiple languages. Does not adopt templates.	QALD-1. NAGA.	Accuracy: 55%
T015	HybQA Mohamed et al. (2017)	Hybrid approach. Open domain. Relation Extraction	Does not handle complex questions;	WebQuestions	Accuracy: 57%
T016	ScoQAS Latifi et al. (2017); Latifi (2018)	Hybrid approach. Open and Closed domain. Graph inferences,	Does not handle complex questions;	QALD-2. QALD-3. QALD-4.	Accuracy: 60%

ments, for example. For a QA system, the big challenge is mediating between the user's need for information in her own language and the available semantic data avoiding idiosyncrasies, expression gaps, and other limitations relevant to machine translation.

7 CONCLUSION

This survey analyzed approaches for QA systems that resort to ontologies for answering questions in both open and closed domains. There is a relatively large number of approaches following this for building QA systems on top of ontologies. In this paper, we analyzed the 15 main strategies found in the literature, by (1) comparing their strengths and weaknesses, (2) highlighting the importance of ontologies in the QA process, (3) characterizing each system in terms of its fundamentals and choices, and (4) distinguishing the differences between existing systems.

A strong point of this type of system is the use of the ontology schema to enrich the query construction, the lack of data for training, and richer query construction. The main weaknesses are the need for prior knowledge of the *schema* used in the ontology and the manual steps usually performed in the process, such as mapping elements, etc.

As future works, we are already developing efforts over interesting topics arising from this investigation: to train new architectures on this problem; investigate other factual question/knowledge bases and their respective influence on the accuracy of the models; and address specific research challenges outlined in Section 5, such as Queries over Multiple Knowledge Bases and answering Complex Questions..

REFERENCES

Baader, F., Calvanese, D., McGuinness, D., Patel-Schneider, P., and Nardi, D. (2003). *The description logic handbook: Theory, implementation and applications*. Cambridge university press.

Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., and Etzioni, O. (2007). Open information extraction from the web. In *Ijcai*, volume 7, pages 2670–2676.

Berant, J., Chou, A., Frostig, R., and Liang, P. (2013). Semantic parsing on freebase from question-answer pairs. In *EMNLP*, pages 1533–1544.

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., and Hellmann, S. (2009). Dbpedia-a crystallization point for the web of data. *Web Semantics: science, services and agents on the world wide web*, 7(3):154–165.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *ACM SIGMOD*, pages 1247–1250. AcM.

Bordes, A., Usunier, N., Chopra, S., and Weston, J. (2015). Large-scale simple question answering with memory networks. *arXiv preprint arXiv:1506.02075*.

Cabrio, E., Cojan, J., Aprosio, A. P., Magnini, B., Lavelli, A., and Gandon, F. (2012). Qakis: an open domain qa system based on relational patterns. In *ISWC 2012*.

Cimiano, P., Lopez, V., Unger, C., Cabrio, E., Ngomo, A.-C. N., and Walter, S. (2013). Multilingual question answering over linked data (qald-3): Lab overview. In *CLEF*, pages 321–332. Springer.

Cimiano, P. and Minock, M. (2009). Natural language interfaces: what is the problem?–a data-driven quantitative analysis. In *nldb*, pages 192–206. Springer.

Cohen, W. W., Ravikumar, P., Fienberg, S. E., et al. (2003). A comparison of string distance metrics for name-matching tasks. In *IIWeb*, volume 2003, pages 73–78.

Collins, A. M. and Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological review*, 82(6):407.

Daconta, M. C., Obrst, L. J., and Smith, K. T. (2003). *The Semantic Web: a guide to the future of XML, Web services, and knowledge management*. John Wiley & Sons.

Damljanovic, D., Agatonovic, M., and Cunningham, H. (2010). Natural language interfaces to ontologies: Combining syntactic analysis and ontology-based lookup through the user interaction. In *ESWC*, pages 106–120. Springer.

De Marneffe, M.-C. and Manning, C. D. (2008). The stanford typed dependencies representation. In *Coling 2008: proceedings of the workshop on cross-framework and cross-domain parser evaluation*, pages 1–8. Association for Computational Linguistics.

Diefenbach, D., Lopez, V., Singh, K., and Maret, P. (2018). Core techniques of question answering systems over knowledge bases: a survey. *Knowledge and Information systems*, 55(3):529–569.

Fader, A., Zettlemoyer, L., and Etzioni, O. (2013). Paraphrase-driven learning for open question answering. In *51st ACL*, volume 1, pages 1608–1618.

Ferrández, O., Izquierdo, R., Ferrández, S., and Vicedo, J. L. (2009). Addressing ontology-based question answering with collections of user queries. *Information Processing & Management*, 45(2):175–188.

Freitas, A., Curry, E., Oliveira, J. G., and O'Riain, S. (2012). Querying heterogeneous datasets on the linked data web: Challenges, approaches, and trends. *IEEE Internet Computing*, 16(1):24–33.

Haarslev, V. and Möller, R. (2001). Racer system description. In *IJCAR*, pages 701–705. Springer.

Haarslev, V., Möller, R., and Wessel, M. (2004). Querying the semantic web with racer + nrql. In *ADL'04*, volume 24.

Hakimov, S., Tunc, H., Akimaliev, M., and Dogdu, E. (2013). Semantic question answering system over linked data using relational patterns. In *EDBT/ICDT 2013 Workshops*, pages 83–88. ACM.

Hakimov, S., Unger, C., Walter, S., and Cimiano, P. (2015). Applying semantic parsing to question answering over linked data: Addressing the lexical gap. In *NLDB*, pages 103–109. Springer.

Herzog, O. and Rollinger, C.-R. (1991). *Text understanding in LILOG: integrating computational linguistics and artificial intelligence: final report on the IBM Germany LILOG-Project*. Springer.

Hirschman, L. and Gaizauskas, R. (2001). Natural language question answering: the view from here. *natural language engineering*, 7(4):275–300.

Hoffart, J., Suchanek, F. M., Berberich, K., and Weikum, G. (2013). Yago2: A spatially and temporally enhanced knowledge base from wikipedia. *Artificial Intelligence*, 194:28–61.

Höffner, K., Walter, S., Marx, E., Usbeck, R., Lehmann, J., and Ngonga Ngomo, A.-C. (2017). Survey on challenges of question answering in the semantic web. *Semantic Web*, 8(6):895–920.

Hogan, A., Harth, A., Umbrich, J., Kinsella, S., Polleres, A., and Decker, S. (2011). Searching and browsing linked data with swse: The semantic web search engine. *Web semantics: science, services and agents on the world wide web*, 9(4):365–401.

Kalaivani, S. and Duraiswamy, K. (2012). Comparison of question answering systems based on ontology and semantic web in different environment. In *Journal of Computer Science*. Citeseer.

Latifi, M. (2018). *Using natural language processing for question answering in closed and open domains*. PhD thesis, Universitat Politècnica de Catalunya, Barcelona.

Latifi, M., Rodríguez Hontoria, H., and Sànchez-Marrè, M. (2017). Scoqas: A semantic-based closed and open domain question answering system. *Procesamiento del Lenguaje Natural*.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436.

Lehmann, J., Furche, T., Grasso, G., Ngomo, A.-C. N., Schallhart, C., Sellers, A., Unger, C., Bühmann, L., Gerber, D., Höffner, K., et al. (2012). Deqa: deep web extraction for question answering. In *ISWC*, pages 131–147. Springer.

Lopez, V., Unger, C., Cimiano, P., and Motta, E. (2013). Evaluating question answering over linked data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 21:3–13.

Lopez, V., Uren, V., Motta, E., and Pasin, M. (2007). Aqualog: An ontology-driven question answering system for organizational semantic intranets. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):72–105.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Mishra, A. and Jain, S. K. (2016). A survey on question answering systems with classification. *Journal of King Saud University-Computer and Information Sciences*, 28(3):345–361.

Mithun, S., Kosseim, L., and Haarslev, V. (2007). Resolving quantifier and number restriction to question owl ontologies. In *SKG 2007*, pages 218–223. IEEE.

Mohamed, R., El-Makky, N. M., and Nagi, K. (2017). Hybqa: Hybrid deep relation extraction for question answering on freebase. In *KEOD*, pages 128–136.

Paolacci, G., Chandler, J., and Ipeirotis, P. G. (2010). Running experiments on amazon mechanical turk. *Judgment and Decision making*, 5(5):411–419.

Petrochuk, M. and Zettlemoyer, L. (2018). Simplequestions nearly solved: A new upperbound and baseline approach. *arXiv preprint arXiv:1804.08798*.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Seaborne, A. and Prud'hommeaux, E. (2008). Sparql query language for rdf. w3c recommendation. *WWW*, 15.

Shekarpour, S., Marx, E., Ngomo, A.-C. N., and Auer, S. (2015). Sina: Semantic interpretation of user queries for question answering on interlinked data. *Journal of Web Semantics*, 30:39–51.

Soares, M. A. C. and Parreiras, F. S. (2018). A literature review on question answering techniques, paradigms and systems. *Journal of King Saud University-Computer and Information Sciences*.

Suchanek, F. M., Kasneci, G., and Weikum, G. (2007). Yago: a core of semantic knowledge. In *16th WWW*, pages 697–706. ACM.

Suchanek, F. M., Kasneci, G., and Weikum, G. (2008). Yago: A large ontology from wikipedia and wordnet. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(3):203–217.

Tablan, V., Damljanovic, D., and Bontcheva, K. (2008). A natural language query interface to structured information. In *ESWC*, pages 361–375. Springer.

Tasar, C. O., Komesli, M., and Unalir, M. O. (2018). Systematic mapping study on question answering frameworks over linked data. *IET Software*, 12(6):461–472.

Unger, C. and Cimiano, P. (2011). Pythia: Compositional meaning construction for ontology-based question answering on the semantic web. In *NLDB*, pages 153–160. Springer.

Wang, C., Xiong, M., Zhou, Q., and Yu, Y. (2007). Panto: A portable natural language interface to ontologies. In *ESWC*, pages 473–487. Springer.

Weikum, G., Hoffart, J., and Suchanek, F. (2019). Knowledge harvesting: achievements and challenges. In *Computing and Software Science*, pages 217–235. Springer.

Wilensky, R., Chin, D. N., Luria, M., Martin, J., Mayfield, J., and Wu, D. (1988). The berkeley unix consultant project. *Computational Linguistics*, 14(4):35–84.

Wohlgenannt, G., Mouromtsev, D., Pavlov, D., Emelyanov, Y., and Morozov, A. (2019). A comparative evaluation of visual and natural language question answering over linked data. *arXiv preprint arXiv:1907.08501*.

Yahya, M., Berberich, K., Elbassuoni, S., Ramanath, M., Tresp, V., and Weikum, G. (2012). Natural language questions for the web of data. In *2012 EMNLP-IJCNLP*, pages 379–390. Association for Computational Linguistics.

Yahya, M., Berberich, K., Elbassuoni, S., and Weikum, G. (2013). Robust question answering over the web of linked data. In *22nd ACM CIKM*, pages 1107–1116. ACM.