An Italian Question Answering System Based on Grammars Automatically Generated from Ontology Lexica

Gennaro Nolano[1], **Mohammad Fazleh Elahi**[2], **Maria Pia di Buono**[1], **Basil Ell**[2,3] **and Philipp Cimiano**[2]
1. UniOr NLP Research Group, University of Naples "L'Orientale", Italy
2. Cognitive Interaction Technology Center, Bielefeld University, Germany
3. Department of Informatics, University of Oslo
gnolano,mpdibuono@unior.it,
{melahi,bell,cimiano}@techfak.uni-bielefeld.de

Abstract

The paper presents an Italian question answering system over linked data. We use a model-based approach to question answering based on an ontology lexicon in lemon format. The system exploits an automatically generated lexicalized grammar that can then be used to interpret and transform questions into SPARQL queries. We apply the approach for the Italian language and implement a question answering system that can answer more than 1.6 million questions over the DBpedia knowledge graph.

1 Introduction

As the amount of linked data published on the Web keeps increasing, there is an expanding demand for multilingual tools and user interfaces that simplify the access and browsing of data by end-users, so that information can be explored in an intuitive way. This need is what motivated the development of tools such as Question Answering (QA) systems, whose main aim is to make users be able to explore complex datasets and an ever growing amount of data in an intuitive way, through natural language.

While the default approach for many NLP tasks has recently been represented by machine learning systems, the use of such approaches (Chakraborty et al., 2019) for QA over RDF data suffers from lack of controllability, making the governance and incremental improvement of the system challenging, not to mention the initial effort of collecting and providing training data for a specific language.

An alternative is the so-called model-based approach to QA, in which a model is first used to specify how concepts and relations are realized in natural language, and then this specification is employed to interpret questions from users. One such system is the one proposed by (Benz et al., 2020), which makes use of a lexicon in lemon format (McCrae et al., 2011) to specify how the vocabulary elements of an ontology or knowledge graph (e.g., entities and relations from a Knowledge Graph) are realized in natural language.

The previous work on this approach shows how, leveraging on lemon lexica, question answering grammars can be automatically generated, and those can, in turn, be used to interpret questions and then parse them into SPARQL queries. A QA web application developed in previous work (Elahi et al., 2021) has further shown that such QA systems can scale to large numbers of questions and that the performance of the system is practically real-time from an end-user perspective.

In this work we describe the extension to the Italian language of the model-based approach described in (Benz et al., 2020) and the QA system described in (Elahi et al., 2021). By doing so, we develop a QA system that can answer more than 1.6 million Italian questions over the DBpedia knowledge graph[1].

2 Related Work

Besides the goal of creating QA systems that are robust and have high performance, an important goal is also to develop systems that can be ported to languages other than English. The interest in other languages is, for example, explicitly stated in the Multiple Language Question Answering Track at CLEF 2003 (Magnini et al., 2004), that includes Italian among others.

One of the earlier attempts in this regard has been the DIOGENE model (Magnini et al., 2002; Tanev et al., 2004), which exploits linguistic tem-

[1] https://www.dbpedia.org/

plates and keyword recognition to answer questions over document collections. Other efforts have been made in the QALL-ME project (Cabrio et al., 2007; Cabrio et al., 2008; Óscar Ferrández et al., 2011), where a system was created for the tourism domain through an instance-based method, that is by clustering together similar question-answer pairs.

More recently, the QuASIt model (Pipitone et al., 2016), makes use of the *Construction Grammar* and an abstraction of cognitive processes to account for the inherent fluidity of language, while exploiting linguistic and domain knowledge (in the form of an ontology) to answer essay and multiple choice questions. Similarly, the authors of (Leoni et al., 2020) built a system to answer questions regarding a specific domain using IBM Watson services and online articles as source of information.

These kind of systems, built to answer questions using textual information, have been largely growing in recent years, especially since the availability of large QA datasets such as the Stanford Question Answering Dataset (SQuAD)[2], which allows to train complex deep learning models with millions of parameters (Rajpurkar et al., 2016; Rajpurkar et al., 2018). While the performance shown by these models is impressive, they suffer from major drawbacks: first of all, they need an extremely large dataset to be trained on, making the porting of such a system to another language extremely demanding;[3] furthermore, they show a lack of controllability in the sense that it is unclear which new examples are to be added to make a new question answerable. This makes systems opaque and difficult to maintain.

The MULIB system (Siciliani et al., 2019) tackles the problem of answering questions in Italian over structured data. The system is based on a modified version of the automaton developed for CANaLI (Mazzeo and Zaniolo, 2016), but it employs a Word2Vec model (Mikolov et al., 2013) to allow for more flexibility in language use. In contrast to these trained approaches, we present a model that generates (i) a deeper interconnection of semantic and syntactic information through the integration of a lemon lexicon with the DBpedia ontology, and (ii) the focus on Linked Open Data as a source of knowledge.

[2] `https://rajpurkar.github.io/SQuAD-explorer/`

[3] The Italian translation for SQuAD, for example, has been described in Croce et al. (2018)

3 Methodology

The architecture consists of two components: (i) the grammar generator and (ii) the QA component. The approach to grammar generation for different syntactic frames according to LexInfo (Cimiano et al., 2011) for the English language was described in a previous work (Benz et al., 2020). In this paper we show that, through a simple language adaptation, we are able to adjust the system so that the system also accepts questions in Italian language.

In a nutshell, the grammar generation approach relies on a mapping between syntactic constructions and classes and properties from a given ontology and/or knowledge graph. This generation process makes use of several *frames*, each describing the linguistic realizations of specific properties that might appear in questions. Thus, the frames employed in this work are: *NounPPFrame*, *TransitiveFrame*, *IntransitivePPFrame*, *AdjectiveAttributive* and *AdjectiveGradable*.

For example, the (lexicalized) construction for the NounPPFrame *'the capital of X'*, can be regarded as expressing the DBpedia property `dbo:capital`, with Country as domain and City as range. This would lead to the generation of the following questions:

- What is the **capital** of X (Country)?
- Which city is the **capital** of X (Country)?

Similar grammar generation rules exist for transitive constructions (TransitiveFrame) as well as constructions involving an intransitive verb with a prepositional complement (IntransitivePPFrame) as well as adjective constructions in attributive (AdjectiveAttributive) and predicate form (AdjectiveGradable).

In the context of this work, we adapted the generation of rules to the Italian language, without extending or modifying the existing types of constructions[4].

In adapting the grammar generation to Italian, we had to accommodate for the following language-specific properties:

- Sentence order, e.g., in sentence starting with interrogative pronouns the subject has to be

[4] The code for our grammar generation for Italian is available at `https://github.com/fazleh2010/question-grammar-generator`

placed at the end of the sentence, e.g., *Dove si trova Vienna?* (Where is Vienna?)

- The presence of auxiliary verbs, either *avere* (have) or *essere* (be), in compound tenses;
- Interrogative pronoun rules, e.g., *chi* (who) is invariable and refers only to people;
- The use of interrogative adjectives, e.g., *quale* (which);
- The use of different prepositions, either simple or articulated, on the basis of range/domain semantics (e.g., toponyms might require different prepositions);
- The presence of a determiner/articulated preposition on the basis of range/domain semantics (e.g., toponyms are preceded by a determiner when the noun refers to a country).

```

:lexicon_en a lemon:Lexicon ;
  lemon:language "it" ;
  lemon:entry :capital_of ;
  lemon:entry :di .

:capital_of a lemon:LexicalEntry ;
  lexinfo:partOfSpeech lexinfo:noun ;
  lemon:canonicalForm :capital_form ;
  lemon:synBehavior :capital_of_nounpp ;
  lemon:sense :capital_sense1 .

:capital_form a lemon:Form ;
  lemon:writtenRep "capitale"@it .

:capital_of_nounpp a lexinfo:NounPPFrame ;
  lexinfo:copulativeArg :arg1 ;
  lexinfo:prepositionalAdjunct :arg2 .

:capital_sense1 a lemon:OntoMap, lemon:LexicalSense ;
  lemon:ontoMapping :capital_sense1 ;
  lemon:reference dbo:capital ;
  lemon:subjOfProp :arg2 ;
  lemon:objOfProp :arg1 ;
  lemon:condition :capital_condition .

:capital_condition a lemon:condition ;
  lemon:propertyDomain dbo:Country ;
  lemon:propertyRange dbo:City .

:arg2 lemon:marker :di .

:di a lemon:SynRoleMarker ;
  lemon:canonicalForm [ lemon:writtenRep "della"@it ] ;
  lexinfo:partOfSpeech lexinfo:preposition .
```

Figure 1: Lemon entry for the relational noun *'capitale della'*

Consider the lemon lexical entry in Figure 1[5] for the relational noun *'capitale della'*. The entry states that the canonical written form of the entry is *"capitale"*. It states that the entry has a NounPPFrame as syntactic behaviour, that is it corresponds to a copulative construction *X è la capitale della Y* with two arguments, where `copulativeArg` corresponds to the copula subject X and the *prepositional adjunct* corresponds to the prepositional object Y.

We give examples for the different syntactic frames below to illustrate the behaviour of the Italian grammar generation.

NounPPFrame Assuming that in the corresponding lemon lexicon we model the connection between the NounPP construction *capitale della* (capital of) as referring to the property `dbo:capital` with domain Country and range City, we can generate questions automatically such as:

1. *Qual è la capitale della* (What is the capital of) (X—Country_NP)?
2. *Quale città è la capitale della* (Which city is the capital of) (X—Country_NP)?

where *X* is a placeholder allowing to fill in a particular country, e.g. *Germania* (Germany), or a noun phrase, e.g., *paese dove si parla tedesco* (the country where German is spoken).

TransitiveFrame Assuming that the lemon lexicon captures the meaning of the construction X *'scrive'* (write) Y as referring to the property `dbp:author`, with Song as domain and Person as range, the following questions would then be covered by an automatically generated grammar:

1. *Chi ha scritto* (Who wrote) (X—Song_NP)?
2. *Quale cantante ha scritto* (Which singer wrote) (X—Song_NP)?
3. *Quale* (Which) (X—Song_NP) *è stata scritta da* (was written by) (Y—Person_NP)?

IntransitivePPFrame Assuming that the lemon lexicon captures the meaning of the construction *'X pubblicare nel Y'* (*'X published in Y'*) as representation of the property `dbp:published`, with Song as its domain and Date as its range, the following questions would be generated:

1. *Quando è stata pubblicata (X—Song_NP)?* (When was (X—Song_NP) published?),
2. *Quale (X—Song_NP) è stata pubblicata nel (Y—date)?* (Which (X—Song_NP) was published in (Y—date)?
3. *In quale data è stata pubblicata* (In which date was) (X—Song_NP)?

[5] In this paper we abbreviate URIs with the namespace prefixes `dbo`, `dbp`, `lemon`, and `lexinfo` which can be expanded into `http://dbpedia.org/ontology/`, `http://dbpedia.org/property/`, `https://lemon-model.net/lemon#`, and `http://www.lexinfo.net/ontology/2.0/lexinfo#`, respectively.

LexInfo Frame	Syntactic Pattern	Question Sample
NounPP	WDT/WP V* DT [noun] IN DT [domain] WDT `dbo:range` V* DT [noun] IN [domain]? WDT/WP V* DT [noun] in [domain] [range] V* DT [noun] IN (DT) [domain]	*Qual è la **capitale della** Germania?* *Quale città è la **capitale della** Germania?* *Chi era la **moglie di** Abraham Lincoln?* *Rita Wilson è la **moglie di** Tom Hanks?*
AdjectiveAttributive	WDT V* DT `dbo:range` [adjective] [domain] VB (DT) [adjective]	*Chi era un vescovo cristiano **spagnolo**?* *Barack Obama è un **democratico**?*
AdjectiveGradable	WRB V* [adjective] DT [domain] WDT V* DT [domain] JJS IN (DT) [range]	*Quanto è **lungo il Barguzin?*** *Qual è la montagna **più alta** della Germania?*
Transitive	WP V* [domain] WDT `dbo:range` V* [domain] WP V* DT [domain] WDT `dbo:range` V* DT [domain] [domain] V* [range]	*Chi **ha scritto** Ziggy Stardust?* *Quale cantante **ha scritto** Ziggy Stardust?* *Chi **ha fondato** C&A?* *Quale persona **ha fondato** C&A?* *Socrate **ha influenzato** Aristotele?*
IntransitivePP	WRB VB [domain] IN WDT `dbo:domain` VB [range] WDT `dbo:domain` VB IN [range] [domain] V* IN [range]	*Quando **è iniziata** l'operazione Overlord?* *In quale data **è iniziata** l'operazione Overlord?* *Quale libro **è stato pubblicato** nel 1563?* *Il libro dei martiri di Foxe **è stato pubblicato** nel 1563?*

Table 1: Italian Patterns and Questions

Frame type	#Entries	#Grammar rules	#Questions
NounPPFrame	113	226	1,010,234
TransitiveFrame	41	124	595,854
IntransitivePPFrame	58	116	52,040
AdjectiveAttributiveFrame	29	130	10,025
AdjectiveGradable	8	24	3,123
Total	**249**	**620**	**1,671,276**

Table 2: Frequencies of entries with a certain frame type. The entries are created manually; the rules and questions are generated automatically.

AdjectiveAttributive and AdjectiveGradable Assuming that the lemon lexicon would capture the meaning of the (gradable) adjective *lungo* (long) as referring to the ontological property `dpb:length`, the grammar generation approach would generate the following types of questions:

1. *Quanto è lungo il* (How long is the) (X—River_NP)?
2. *Qual è il fiume più lungo (del mondo, del Kentucky)?* (What is the longest river in (*the world*, *Kentucky*)?).

The rules implemented for the generation of Italian questions are shown in further detail in Table 1. In particular, we use the tagset[6] from the Penn Treebank Project (Marcus et al., 1993), with `V*` defining all possible forms of a given verb, words in brackets defining nouns/verbs/adjectives that realize a specific property, and `dbo:range`/`dbo:domain` defining the possible labels that may represent classes (e.g., `dbo:Country` might be represented by either *paese* or *stato*).

4 Results

We apply our system to the DBpedia dataset and manually created a lemon lexicon comprising of 249 lexical entries[7]. Table 2 shows the number of grammar rules and questions generated for each syntactic type. Altogether, the approach generates 620 grammar rules and about 1.6 million questions. The web-based demonstration is available online[8].

We used the training set of multilingual QALD-

[6] https://www.sketchengine.eu/english-treetagger-pipeline-2/

[7] https://scdemo.techfak.uni-bielefeld.de/quegg-resources/

[8] https://webtentacle1.techfak.uni-bielefeld.de/quegg/

7[9] to evaluate our approach. QALD-7 contains a total of 214 questions over linked data, covering for more relations than the ones we considered so far. In order to overcome this issue, a total of 109 entries were added to our system (22 NounPPFrame, 41 TransitiveFrame, 41 IntransitiveFrame, 1 AdjectiveAttributiveFrame and 4 AdjectiveGradable).

Precision	0.485
Recall	0.224
F-Measure	**0.307**

Table 3: Evaluation results against QALD-7

The results of the evaluation process (Table 3) show a quite satisfying precision, but a low recall. The main reason behind such results is related to the presence of different types of questions in QALD. Indeed, besides single-triple questions, QALD presents also complex questions referring to more than one triple, e.g., *A quale movimento artistico apparteneva il pittore de I tre ballerini?* (What was the artistic movement of the author of The Three Dancers?), which are not covered yet by our model. Nevertheless, when taking into account all the questions in QALD-7, our system recognizes 46.98% (101 questions) of the total set of questions.

5 Conclusion and Future Work

We presented an approach to developing Italian QA systems over linked data that relies on the automatic generation of grammars from corresponding lemon lexica describing how elements of the dataset are realized in natural language. The approach is controllable, since the introduction of a lexical entry increases the question coverage in a fully predictable way. Our proof-of-concept implementation over DBpedia covers 1.6 million questions generated from 249 lemon entries.

In future work, we intend to further automatize grammar generation by using LexExMachina (Ell et al., 2021), which induces lexicon entries bridging the gap between ontology and natural language from a corpus in an unsupervised manner.

Acknowledgments This work has been funded by the European Commission under grant 825182 (Prêt-à-LLOD) as well as Nexus Linguarum Cost Action. M.P. di Buono has been partially supported by Programma Operativo Nazionale Ricerca e Innovazione 2014-2020 - Fondo Sociale Europeo, Azione I.2 "Attrazione e Mobilità Internazionale dei Ricercatori" Avviso D.D. n 407 del 27/02/2018. B. Ell has been partially supported by the SIRIUS centre: Norwegian Research Council project No 237898.

[9] `https://github.com/ag-sc/QALD`

References

Viktoria Benz, Philipp Cimiano, Mohammad Fazleh Elahi, and Basil Ell. 2020. Generating Grammars from lemon lexica for Questions Answering over Linked Data: a Preliminary Analysis. In *NLIWOD workshop at ISWC*, volume 2722, pages 40–55.

Elena Cabrio, Bonaventura Coppola, Roberto Gretter, Milen Kouylekov, Bernardo Magnini, and Matteo Negri. 2007. Question answering based annotation for a corpus of spoken requests. In *Proceedings of the workshop on the Semantic Representation of Spoken Language*, volume 31.

Elena Cabrio, Milen Kouylekov, Bernardo Magnini, Matteo Negri, Laura Hasler, Constantin Orasan, David Tomás, Jose Luis Vicedo, Guenter Neumann, and Corinna Weber. 2008. The QALL-ME benchmark: a multilingual resource of annotated spoken requests for question answering. In *LREC'08*.

Nilesh Chakraborty, Denis Lukovnikov, Gaurav Maheshwari, Priyansh Trivedi, Jens Lehmann, and Asja Fischer. 2019. Introduction to Neural Network based Approaches for Question Answering over Knowledge Graphs. *CoRR*, abs/1907.09361.

Philipp Cimiano, Paul Buitelaar, John P. McCrae, and Michael Sintek. 2011. LexInfo: A declarative model for the lexicon-ontology interface. *JWS*, 9(1):29–51.

Danilo Croce, Alexandra Zelenanska, and Roberto Basili. 2018. Neural learning for question answering in italian. In *AI*IA 2018*, pages 389–402.

Mohammad Fazleh Elahi, Basil Ell, Frank Grimm, and Philipp Cimiano. 2021. Question Answering on RDF Data based on Grammars Automatically Generated from Lemon Models. In *SEMANTiCS Conference, Posters and Demonstrations*.

Basil Ell, Mohammad Fazleh Elahi, and Philipp Cimiano. 2021. Bridging the Gap Between Ontology and Lexicon via Class-Specific Association Rules Mined from a Loosely-Parallel Text-Data Corpus. In *LDK 2021*, pages 33:1–33:21.

Chiara Leoni, Ilaria Torre, and Gianni Vercelli. 2020. ConversIAmo: Improving Italian Question Answering Exploiting IBM Watson Services. In *Text, Speech, and Dialogue*, pages 504–512.

Bernardo Magnini, Matteo Negri, Roberto Prevete, and Hristo Tanev. 2002. Mining Knowledge from Repeated Co-Occurrences: DIOGENE at TREC 2002.

Bernardo Magnini, Simone Romagnoli, Alessandro Vallin, Jesús Herrera, Anselmo Peñas, Víctor Peinado, Felisa Verdejo, and Maarten de Rijke. 2004. The Multiple Language Question Answering Track at CLEF 2003. In *Comparative Evaluation of Multilingual Information Access Systems*, pages 471–486.

Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Comput. Linguist.*, 19(2):313—-330.

Giuseppe M. Mazzeo and Carlo Zaniolo. 2016. Answering controlled natural language questions on RDF knowledge bases. In *EDBT*, pages 608–611.

John P. McCrae, Dennis Spohr, and Philipp Cimiano. 2011. Linking Lexical Resources and Ontologies on the Semantic Web with Lemon. In *ESWC Conference*, pages 245–259.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26.

Arianna Pipitone, Giuseppe Tirone, and Roberto Pirrone. 2016. QuASIt: A Cognitive Inspired Approach to Question Answering for the Italian Language. volume 10037, pages 464–476.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. *CoRR*, abs/1606.05250.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know What You Don't Know: Unanswerable Questions for SQuAD. *CoRR*, abs/1806.03822.

Lucia Siciliani, Pierpaolo Basile, Giovanni Semeraro, and Matteo Mennitti. 2019. An italian question answering system for structured data based on controlled natural languages. In *CLiC-it*.

Hristo Tanev, Matteo Negri, Bernardo Magnini, and Milen Kouylekov. 2004. The DIOGENE question answering system at CLEF-2004. volume 3491, pages 435–445.

Óscar Ferrández, Christian Spurk, Milen Kouylekov, Iustin Dornescu, Sergio Ferrández, Matteo Negri, Rubén Izquierdo, David Tomás, Constantin Orasan, Guenter Neumann, Bernardo Magnini, and Jose Luis Vicedo. 2011. The QALL-ME Framework: A specifiable-domain multilingual Question Answering architecture. *Journal of Web Semantics*, 9(2):137–145.