Marie Skłodowska-Curie Actions (MSCA)
Innovative Training Networks (ITN)
H2020-MSCA-ITN-2014

WDAqua
"Answering Questions Using Web Data"
ETN, Grant no. 642795

http://wdaqua.informatik.uni-bonn.de

Dissemination level	Public
Type of document	Report
Contractual date of delivery	M9
Actual date of delivery	2015-10-31
Deliverable number	D6.1
Deliverable name	Web & Data Science Techniques for Question Answering: A State of the Art
Deliverable leader	UJMS
Work package	WP6
Tasks	T6.1
Coordinator (name, affiliation, contact)	Pierre Maret (UJMS; pierre.maret@univ-st-etienne.fr)
Other contributors	See page 2
EC project officer	Thierry Colin

Web & Data Science Techniques for Question Answering: A State of the Art

Alessandro Piscopo[1], Ashwini Jaya Kumar[2], Christoph Lange[3], Christoph Schmidt[2], Denis Lukovnikov[3], Dennis Diefenbach[4], Hady Elsahar[4], Harsh Thakkar[3], José M. Giménez-García[4], Kamal Singh[4], Pierre Maret[4], Kemele M. Endris[3], Kuldeep Singh[2], Nikolaos Liappas[1], Pavlos Vougiouklis[1], and Saeedeh Shekarpour[3]

[1]SOTON , {A.Piscopo, N.Liappas, P.Vougiouklis}@soton.ac.uk
[2] FRAUNHOFER , {ashwini.jaya.kumar, christoph.andreas.schmidt, kuldeep.singh}@iais.fraunhofer.de
[3]UBO , {langec, lukovnik, hthakkar, endris, shekarpour}@uni-bonn.de
[4]UJMS, {dennis.diefenbach, hady.elsahar, jose.gimenez.garcia, kamal.singh, pierre.maret}@univ-st-etienne.fr

October 2015

Contents

1 Introduction

Question Answering (QA) has become a very active research field in the last years due to the advent of the Semantic Web [108]. While the "traditional" World Wide Web is a web of documents, where information retrieval systems have to parse them in order to extract information, the Semantic Web is a web of interrelated data encoded with semantic meaning. This makes possible for automatic agents to query and process the data, allowing to answer potentially very complex questions.

The Semantic Web provides standardized formats to represent and query this new type of data. RDF [154] allows to represent semantic data in a graph conceptual model, while RDFS [24] and OWL [72] enable the creation of vocabularies and ontologies, and to make use of inference and reasoning. SPARQL [80] is defined as standard query language to enquire datasets in RDF format. The Linked Open Data[1] project is an initiative to put in practice the Semantic Web principles and openly expose interlinked datasets. Since its inception, the LOD Cloud[2] has been growing at a dramatic pace and contains information about media, publications, geography, life-science and more.

The process of QA consists of a number of steps such as voice recognition, understanding the question, finding the data to answer the question, generating queries to retrieve the answer and presenting the answer. While a lot of research has been done in each of this steps, the QA process as a whole has been rarely addressed by a research project. IBM Watson [125], Wolfram Alpha[3] and Siri[4] are examples of these projects, but they are based on proprietary ontology formats and there is no easy way to extend the underlying knowledge. In addition, they are not accessible to the research community.

The WDAqua project[5] aims to advance the State of the Art on data-driven QA, focusing on developing an ecosystem of tools able to find, link and integrate ontologies, understand a question, query many different sources, and present an answer. The overall architecture will be domain-independent, scalable, cross-lingual and designed so that new QA systems can be set up using data published on the web with minimum effort.

This document portrays the State of the Art for Web and Data Science techniques for QA, a key cornerstone to exploit semantic data evolution. We describe each step involved in a data-driven QA system, explore the main challenges and, when possible, relevant proposed solutions. Section 2 deals with extracting valuable data from the web, characterizing the challenges regarding the collection and maintenance of data, including how to extract data from unstructured sources, and dealing with data quality, organization and confidence. Section 3 delves into the task of querying this data using natural language, from speech-to-text recognition to query generation, including text analysis, evaluation and the need of a general architecture. Finally, section 4 provides an overview on relevant State of the Art regarding user interaction and text generation.

2 Collecting and Maintaining Web Data

This section describes how to gather, process and maintain data to make it fit for QA. Source data can be either in RDF or in unstructured text format. For the latter, data extraction is a first step to be performed (subsection 2.1); for the former, data can be directly reused, but attention must be paid to its characteristics in order to better profit from it (2.2). Quality assessment of a dataset (subsection 2.3), together with provenance and trust handling (subsection 2.4) are necessary tasks to select the more adequate datasets for QA. Then, quality improvement through cleaning and enrichment (subsection 2.5) improves accuracy and completeness for the chosen datasets. Finally, data needs to be indexed such that it can be found and retrieved fast (subsection 2.6).

2.1 Information Extraction from unstructured text

Information Extraction is the process of extraction of structured information from free unstructured text. This structured information can be later utilized for many purposes such as knowledge base population, question answering, document summarization and machine translation. Many tasks lies under the umbrella of Information Extraction and they can be categorized according to the type of Relations being extracted.

[1] `http://linkeddata.org`
[2] `http://lod-cloud.net`
[3] `http://www.wolframalpha.com`
[4] `http://www.apple.com/ios/siri/`
[5] `http://wdaqua.informatik.uni-bonn.de/`

In free text there are many types of relations that occur between nominals. Some of these relations have only two arguments which are the most basic yet common form of relations and usually referred to as **Binary relations**, while **N-ary relations** are when the relation includes more than one argument, for example an agreement can include relations between two or more parties, the thing being agreed on, the agreement time, and conditions of the agreement. Additionally relations can be categorized as first order relations which are flat **first order** relations for example: *arrive(john, school)*, or as a **higher order** relation which might include further modifiers, for example *believe(Jessica, arrive(john, school))* . Other form of categorization as well is the syntactic position of the relation in the text, in which, some relations are represented as direct verbs or nouns occurring between nominals, other relations can be hidden in compound nouns such as: "embassy driver" or "Italian restaurant".

These different categorizations resulted in different open tasks for information extraction, most of them are concerned only with one or two specific categories of these relations under specific conditions. In this section we will discuss some of these open tasks and the state-of-the-art techniques used in solving them.

Relation Extraction The common form of relation extraction in the literature, is a list of tasks that are concerned with extraction of relations between entities in the text given a list of predefined relations beforehand to choose from, and the arguments of the relation before hand. Thus, the problem is often formalized as a classification problem and sometimes referred to as relation classification. Most of the research work in this topic is divided into two section, the first one is attempting to create a comprehensive set of relations that can include every relation in text. The other type of research work is concerned with developing algorithms for detection of these relations in text. The first set of algorithms for relation extraction is using patterns either induced or hand crafted, Hearst et al. [83] was one of the pioneers in this field by introducing a set of *lexico-syntactic patterns* to capture hyponym relations (is-a relations). Although hand crafted patterns have proved to have high precision, they result in a very low recall and often require a lot of effort and domain experts to hand craft them. Supervised techniques for relation extraction have proved to be the most successful and the most applied within the literature. These techniques usually rely on a set of features extracted from sentences, a learning algorithm and a set of labeled instances of each relation for training. Some of the features are related to the arguments [175], such as the syntactic position of the arguments in the sentence, the semantic class of the arguments, list of synonyms using dictionaries or knowledge bases (e.g. Wordnet [55]). Others are relation centric [175], such as the path between the arguments of the relation, in terms of words, part of speech tags, constituency or dependency paths in the parse tree. For learning algorithms, one of the methods that proved to have high efficiency are Kernel Methods. The idea of these methods is for each candidate relation, the similarity between the features of a training example and the new example to be classified can be calculated in a high dimensional feature space without the need to enumerate the dimension of that space. Kernel methods such as support vector machines has proved efficiency in many of the machine learning tasks. Applying that for NLP tasks usually require a certain mechanism to extract feature from parse trees and convert them to feature vectors. Thus, several modifications of these kernels has been developed for linguistic structures. Some of them are TreeKernels [199, 39] and convolution kernels [38]. For benchmarking and evaluation of the task of relation classification, the most well known benchmarks are Task 4 in semeval 2007 [66] and Task 8 in semeval 2010 [86]. For the latter benchmark the best performing system during the time of the competition was developed by Rink et al. [140] scoring an F1 score of 82.2%. The system relied on combining lexical and semantic resources. A later improvement on the performance on same benchmark was done by Dos Santos et al. [50] by utilizing a convolution neural network and a large corpus only without any semantic resources scoring an F-score of 84.1%.

Template Filling One of the other forms of Information Extraction from free text is the task of Template Filling. Many unstructured text contains reports of stereotypical situations or events such for example as report of a sports game in the news. Template filling or slot filling is the task of extraction of a predefined set of attributes, referred to as slots. These arguments or the slot-fillers may be in the form of text segments extracted directly from the text, or they may require additional processing such as normalization, grounding or conversion to standard form. One of the well known benchmarks for slot filling task is the ATIS dataset [132] and it's based on the Air Travel Information System. Listing 1 shows sample entries in the ATIS dataset which was basically developed for the purpose of spoken language understanding through extraction of specific target information from human queries. Listing 2 shows a target template with arguments being filled from an unstructured text.

Slot filling is typically modeled as a sequence classification problem in which the input is a sequence of words, and the output is a sequence of slot IDs. The main approaches for solving the slot filling problem included different sequence labelers such as Hidden Markov Models [187] and conditional random fields (CRF) [188]. Recently, DeepLearning techniques have been applied to solve this problem. A recent work by Mesnil et al. [117] [116] compared different kind of standard Recurrent Neural Network (RNN) architectures, elman-type [52] and jordan-type [94] and a bidirectional variant of them for the slot filling task on the ATIS dataset. These models have set a new State of the Ert for the ATIS benchmark.

Many slot filling techniques have reached a very high performance on the ATIS dataset with less than 5% error rate. While this might give the indication that the task is already solved, a study by Tur et al. [171] indicated the necessity of having a more naturally spoken data sets and employing more linguistically motivated features in order to prevent overfitting of the task. Other challenging problem in template filling is the extraction of the information without the necessity of having predefined patterns beforehand [33].

Listing 1: example from ATIS dataset

```
> please list the earliest lunch flight from Dublin to London
> Show me all the nonstop flights between Atlanta and Philadelphia.
> leaving Sunday after twelve noon
> flights from Liverpool to Dublin arriving before five p.m.
> show me the fares from Dublin to London
```

Listing 2: example for a typical slot filling template

```
AIRLINE: UNITED AIRLINES
DEPART: US
ARRIVAL: CA
DEPART-TIME: 2015-09-24-16:00
DURATION: 3:00
```

Open Information Extraction Traditional Relation extraction is concerned with determination of the relation between arguments from a set of predefined relations beforehand. As a result, applying traditional relation extraction techniques to new domains often requires defining new target relations and sometimes hand crafting training examples for them. For this reason open information extraction (Open IE) was first introduced by Banko et al. [11]. Open IE was a slightly different modification to the Relation Extaction task, in which it isn't required to define the target relations before hand. This has enabled large scale extraction of relations over millions of documents from the Internet in multiple domains. Open information extraction is often referred to in the literature as: open relation extraction, unsupervised relation extraction, webscale relation extraction or extraction of emergent relations.

One of the earliest developed Open IE systems was TextRunner [11]. TextRunner extracts relations by "running" over a large corpus of text and examining the text between noun phrases in a large corpus, utilizing in this a part of speech (POS) tagger and noun phrase chunker. ReVerb [54] is a later developed OpenIE system that seeks to alleviate the large number of noisy relations extracted by previous systems by adding some predefined constraints. One type of constraints were syntactic constraints through restricting the extracted relation into a set of part of speech patterns and the other was lexical constraints by restricting the extracted relations to only those which have many different arguments in a large corpora. These two constraints although very simple, have proved to be very effective. Having said that, this type of systems still suffered from problems such as: inability to handle relations expressed in nominal rather than verbs, inability to handle far dependencies between arguments of the relation. Instead of relying on patterns and heuristics another, Banko et al. [12] modeled the Open IE task as a sequence labeling problem. O-crf used conditional random field sequence labeler to label each word in a sentence with either being argument of the relation of the relation word itself. O-crf was trained using heuristics over the penn tree bank and syntactic features including part of speech, regular expressions and word position in the sentence. Having an automatic sequence labeling algorithm rather than hand crafted patterns can yield in higher recall, yet because each assignment in the sequence is uncertain, the likelihood that the extracted relation phrase is flawed increases with the length of sentence and far dependencies between the relation arguments, let alone the need of creation of a training data.

In addition to all of the previous limitations, most of the previous systems have focused only on extraction of binary relations from text and ignored the contextual information that might exist with the relations. This contextual information can hold additional information such as the time and the location of a relation. In other cases the contextual information can totally change the validity of a relation, this often occurs when the relation is modified with a belief, attribution, hypothetical or any other conditional context. OLLIE (Open Language Learning for Information Extraction) [152], by Mausam et al. was the first Open IE system to address the context of extracted relations. OLLIE is bootstrapped by high confidence relations extracted from ReVerb. These extractions are annotated with the dependency parse information and patterns were Learned for future extractions. The confidence of every relation was calculating using a logistic regression classifier and a set of features including the existence of contextual modifiers. Accordingly if a sentence is modified by any context which might affect its validity, it will be automatically learnt and a lower confidence will be given to this relation. Listing 3 shows a sample output of OLLIE and how contextual information can affect the confidence of the triple.

Listing 3: sample extractions of OLLIE Open IE system

```
> Earth was flat.
0.714: (earth; was; flat)

> People believed that earth was flat.
0.596: (earth; was; flat)
```

In this section,we have discussed different open problems for information extraction from free unstructured text. We categorized each of the tasks according to the type of relations they are extracting from the text and discussed the most successful algorithms in the literature for each task. Although Traditional **Relation classification** has been very useful in many tasks such as knowledge base population, ontology creation, and question answering, there hasn't been any consensus on a list of relations fit for all purposes and all domains, let alone that relation classification is only concerned only with binary relations. Thus, other open research problems have emerged which handles n-ary relations and emergent relations such as **template filling** and **open information extraction**. Having said that, many of the features and techniques that proven to be effective in one of these problems can be easily utilized and ported to other problems, for example, Xu et al. [193] utilized TreeKernels, which one of the most successful methods in relation extraction in the problem of open relation extraction and proved to be more efficient than other pattern based open ie systems. Many of resources developed for different purposes can be utilized int he task of information extraction either in training or bootstrapping of models. This includes Knowledge bases such as DBpedia [23], Wikidata [53] Wordnet [55]. Other resources such as PropBank [98], FrameNet [10] which were originally crafted for the task of Semantic Role Labeling contains information about semantic frames, which can be utilized in extraction of n-ary relations. Within the context of Question Answering we can find that almost all of these previous tasks can fit for preparation of data for Question Answering, depending on the domain of application. However, an additional work will be included. As an example, by examining the output of the most Open Information retrieval systems we can notice it is still an open research problem to define a standard output to be easily queried later by QA systems. On the other hand, these standard formats shall fully represent the information and not loose any information which might happen for example when grounding n-ary triples to a list of binary ones, or when dealing with numerical attributes for relations.

2.2 Utilising structured data sources – the case of Wikidata

For publishing structured data on the Web and making them as reusable as possible – for QA and for any other purpose – it is generally advised to follow the 5-star deployment scheme for open data[6]. For the purpose of QA in the WDAqua project, we will therefore prefer data sources that are natively available as Linked Data – open, but also, in enterprise scenarios, with access restrictions. A source which is not natively available as Linked Data – but that has nonetheless recently been encoded in RDF and connected to the Linked Data Web [53] – is Wikidata. This is a highly relevant data source which has been not thoroughly exploited yet for QA, notwithstanding some attempts in that sense. An example is the *Platypus project*[7], which is still in an alpha version, though. In the following, we review the characteristics of the Wikidata project, highlighting the ones that are relevant for its use in a QA system.

Wikidata is a project of the Wikimedia Foundation, launched in October 2012, which aims primarily at building a free knowledge base to provide structured data to support Wikipedia and other projects, either internal or external to Wikimedia[8]. In the first phase of its deployment, it was intended to replace the links

[6]`http://5stardata.info/`, consulted on October 28, 2015.
[7]`http://askplatyp.us/`, consulted on October 30, 2015.
[8]`http://www.wikidata.org/wiki/Wikidata:Introduction`, consulted on October 27, 2015.

connecting different language versions of Wikipedia articles with a centralised system. The first version of Wikipedia to make use of this interlanguage linking system was the Hungarian one, in January 2013[9], while it was extended in March of the same year to all the other languages[10]. A following phase aimed at enabling all Wikipedia language editions to access Wikidata, in order to populate their infoboxes (i.e. the structured tables that provide facts about entities of certain types, such as cities) and enrich their articles. A third phase, currently ongoing, involves the creation of enhanced querying capabilities and lists of data stored in Wikidata.

Wikidata is a free and collaborative project, in that its data is entered, edited and managed by users. The existence of an already broad community, such as the Wikimedia one, ensures that data is continuously curated and updated. In addition, it has allowed to gather up to the present day data about more than 15M entities[11], related both to general and specialised knowledge. This data can be freely reused and shared, as it is released under CC0 licence [183], an open licence that allows to use, modify and distribute it, without requiring any permission[12]. Furthermore, it can be accessed through web services in several formats, such as JSON and RDF. Wikidata is inherently multilingual: non-language specific data, e.g. numeric data, dates or relationships between entities, entered in a language is immediately available in all other featured languages. To make an example, if we state that the entities *Italy* and *Rome* are connected by the relationship *has capital*, this information would be available to users in any language, with the difference that each one will visualise their labels – i.e. their "human understandable" names – in her preferred language. This is made possible by the fact that Wikidata uses URIs for its entities. Moreover, these are divided into *Items* and *Properties*, with Classes being instances of the first type. A network of relationships between Properties and Classes, i.e. Items, constitutes the structure of Wikidata, which has therefore been defined as "rather schemaless" [91], since it does not rely on any formal ontology.

A particular data model is used by Wikidata. This describes entities by means of statements composed of a property-value pair – called *claim* – and optional qualifiers and references. The first ones can be added to a property-value pair to add contextual information and relationships [183]. The latter ones are used to store sources for a statement. The presence of qualifiers and references in statements required an *ad-hoc* approach to encode this knowledge base into RDF, in order to make it available as Linked Data. The solution employed involved the creation of auxiliary individuals representing the statements to which a qualifier or a reference was applied [53]. In a QA system, when available, references can be used to provide provenance details for the data used and to help users identifying trustworthy information. Moreover, the possibility to add sources and qualifiers to claims enables to report contradictory assertions about the same fact. This is of particular interest for QA systems, as it allows to provide diversified answer about debated topics. As an example[13], one user might pose the question "is Kosovo a country?" Although a big part of the EU and UN has recognised Kosovo as an independent state, some countries consider it as a part of Serbia. Wikidata allows to include both positions, each one possibly supported by a source and by a qualifier specifying the date in which it declared its independence. A system relying of Wikidata could therefore provide the user with an answer including both these points of view, each one justified by some third source and enriched with additional contextual information.

Finally, we would like to summarise the reasons that make Wikidata a highly valuable resource for QA systems. First, it is a huge source of information, with general and specific coverage. Second, this information is curated by a vast community of users, which makes it on average fairly accurate – it has been claimed that Wikidata should have higher data quality than DBpedia, due to manual curation [91] – and updated – an entity describing the October 2015 Ankara bombings was created by users shortly after this happened[14]. Moreover, data is exportable through several web services and is available on the Linked Data Web, with outgoing and ingoing links to other knowledge bases, such as DBpedia. In addition, entities have multilanguage labels, which may help interpreting and disambiguating queries in different languages. Finally, due to a data model that allows the addition of qualifiers to statements, Wikidata can be used to provide answers reflecting different points of view, including provenance information for each of these.

[9] `http://blog.wikimedia.de/2013/01/14/first-steps-of-wikidata-in-the-hungarian-wikipedia/`, consulted on October 27, 2015.

[10] `http://blog.wikimedia.de/2013/03/06/wikidata-now-live-on-all-wikipedias/`, consulted on October 27, 2015.

[11] `https://www.wikidata.org/wiki/Wikidata:Main_Page`, consulted on October 30, 2015.

[12] `https://creativecommons.org/publicdomain/zero/1.0/`, consulted on October 30, 2015

[13] This example follows the one made by the former project director of Wikidata, Denny Vrandecic, in a blogpost introducing some Wikidata features: `http://blog.wikimedia.de/2013/02/22/restricting-the-world/`, consulted on October 30, 2015.

[14] `https://www.wikidata.org/wiki/Q21083277`, consulted on October 30, 2015.

2.3 Assessment of dataset quality

Recent advancements in the fields of Web of Data and Data Science have led to an outburst of standards related to structured data[15] such as RDF(a), Linked Data, Schema.org, etc., to an increasing amount of such data, and to a wide range of tools to produce, manage and consume such data. To be available for ready consumption, especially in open question answering systems, any such data sources should meet a certain level of quality, e.g., defined by benchmarks. Quality can generally be defined as "fitness for use", but there are a lot of concrete factors that influence a dataset's fitness for use in question answering[16] settings and in specific application domains. Recently, a number of research activities have been concerned with automating the assessment of linked data quality. Debattista, who has developed one such tool (Luzzu [43]), provides an overview of other state-of-the-art tools [42], including one by Flemming [60], as well as Sieve [115], RDF Unit [99], Triple Check Mate [197], LinkQA [73], and LiQuate [143] and Luzzu [43]. In this section, we summarise the concrete criteria by which the quality of linked data can be assessed, with a special focus on those criteria that are relevant to question answering.

In a comprehensive review of literature and systems, [198] have identified the dimensions of linked data quality and categorised them as follows:

- Accessibility dimensions: This category covers aspects related to retrieving and accessing data, which includes full or partial access and different technical means of access (e.g. the possibility to download a data dump vs. the availability of a SPARQL endpoint, i.e. a standardised query interface). Examples include availability and interlinking.
  - Availability is generally defined as the ease of access with which particular information is obtainable or rapidly retrievable for readily consumption. In a linked data context, availability can be referred to as the accessibility of a SPARQL endpoint or RDF dumps or dereferenceable URIs.
  - Interlinking is relevant as it refers to the data integration and interoperability. RDF triples that provide a link, between the entities recognised by the subjects and those recognised by the objects, crucial for interlinking.
  - Intrinsic dimensions: This category covers aspects that are independent of the user's context, or the out of the application context – such as accuracy and consistency.
  - Accuracy refers to the degree of a dataset correctly representing the captured real world facts and figures in the form of information with high precision. Consistency refers to the independence from logical, formal or representational contradictions of a dataset with respect to others.
- Trust dimensions: This category is concerned with the perceived security and reliability or trustworthiness of the data and its source. For example Verifiability.
  - Verifiability refers to the authenticity and correctness of the dataset. This primarily consists of verifying the authenticity and correctness by either an unbiased third party such as a provenance vocabulary.
- Dataset dynamicity dimensions: This category is concerned with the freshness of data over time or timeliness, i.e. the regularity of updates or merges and so on.
- Contextual dimensions: This category is concerned with the context of the task which is being pursued – such as completeness and security.
  - Completeness is referred to as the degree to which information in the dataset is complete or not missing. The dataset should have all the required objects or values for a given task in order to be considered as complete. Thus, arguing intuitively, completeness is one of the concrete metrics for linked data quality assessment.
  - Security denotes the degree to which a particular dataset is resistant to misuse or alteration without appropriate user access rights.

[15] The amount not only of structured, but also of semi-structured and unstructured data available online is also steadily increasing; however, for the purpose of our work we assume that such data has first been translated to the RDF data model using standard tools, e.g. from the Linked Data Stack [9]

[16] In this section, we do not abbreviate "question answering" as "QA" to avoid confusion with "quality assessment".

- Representational dimensions: This category is concerned with the design and representation of the data and its schema. For instance, understandability and interpretability.
  - Understandability can be achieved by providing appropriate human readable annotations to a dataset and its entities, and by consistently following a certain regular expression as a pattern for forming entity URIs.
  - Interpretability refers to adhering to the standard practice of representing information using appropriate notations, symbols, units and languages.

Data quality dimensions in all of these categories can be relevant in question answering scenarios. Our next step is to identify more systematically what dimensions of data quality are specifically relevant in the typical application domains of question answering, or sufficient for determining a dataset's "fitness" for question answering. Having identified such dimensions, we have two goals: identifying datasets that are suitable for question answering at all, and then, for those datasets that are, identifying more specifically what quality problems they still suffer from. This leads to the questions of what concrete quality metrics for the relevant quality dimensions can be computed on such datasets in a reasonable way, given, e.g., the expressiveness of their schemas, and, secondly, whether implementations are available to effectively and efficiently compute these metrics on the given datasets. Regarding implementation, we expect that the Luzzu linked data quality assessment framework provides a sufficient number of fully implemented quality metrics that are ready to use in a question answering setting, that further existing implementations of metrics in Luzzu can be specifically adapted to make them suitable for quality assessment related to question answering, and that, finally, Luzzu's flexible extensibility even enables us to implement new metrics that may be required. In summary, our near-future work will be concerned with defining a generally and flexibly applicable framework for automating the process of rigorously assessing the quality of linked datasets for question answering by identifying, formalising and implementing the required metrics.

2.4 Handling trust and provenance of Web data

Generation and consumption of semantic data is increasing each day from many different publishers, including science data (such as in astronomy or gene fields), open government initiatives or the Linked Open Data project. Data trustworthiness is, however, an important factor hampering the evolution data web technologies. In an environment of information published independently by many different actors, data veracity and quality is usually uncertain, and there is always the risk of consuming misleading data. How to asses the trustworthiness of given data and search authoritative data, in order to decrease this risk, has been a concern since the inception of the Semantic Web, but it has not been deeply explored and is still a challenge. Nonetheless, it is a topic cross-pollinated by other fields [7], such as Artificial Intelligence [144], the World Wide Web [68], or Multi-agent Systems [131]. Trust is a wide topic with different aspects such as data attribution and provenance, trust representation, and trust assessment. It is important to keep in mind that trust depends on both objective and subjective criteria. That is, while a number of metrics can be used to measure a variety of trust aspects, different consumers may have disparate trust criteria or tolerances for diverse operations.

Data attribution and provenance deals with the origin of the data, be it *who* generated or published it, *when* the data was created or what is its temporal validity, or *how* it was produced (particularly useful when reasoning is involved). Initial attempts to use reification of RDF statements to include meta-data shown that it was not an optimal solution [30], and currently the common approach is to *annotate* the additional information, attaching it to the statements. Early works suggested the usage of quads (i.e. tuples of four elements) to add this information to a triple [111, 51], and in 2005 Carroll et al. consolidated the idea for using *Named Graphs* [31]. This idea is further extended by Flouris et al. [61] with semantics for combinations of different named graphs when inferring new triples. RDF 1.1 N-Quads [28] is a current W3C Recommendation for encoding RDF using quads to include the graph name on each triple.

However, quads are not enough when dealing with more than one type of provenance. On the one hand, a few works have tried to approach the idea of reification with a different angle, such as Provenance Context Entities [145], Singleton Properties [126], or object reification [53], but they still have not gained traction. On the other hand, some works propose different solutions to attach the desired information on the triples [71, 49, 149, 202], with a special emphasis on developing semantic extensions for meta-data, allowing to compute their value when reasoning with triples from different sources, and to include meta-data in the queries.

While not focused on Semantic Data annotation, it is also worth mentioning *PROV* [102], the W3C recommendation for inter-operable interchange of provenance information in heterogeneous environments. The PROV-O ontology allows to record information about the *Agents*, who create or modify *Entities* by means of *Activities*. This standard has been implemented to model dynamic provenance on RDF graphs by Halpin and Cheney [77, 78], allowing to store, update and retrieve provenance meta-data as part of SPARQL sentences.

Those solutions provide means to annotate and manage provenance information, but they assume the validity of these annotations (i.e. they trust the provenance information). Some authors handle with the possibility of impostors trying to add deceiving information about semantic data provenance using cryptographic techniques [29, 147, 148, 87].

Trust representation focuses in modelling and using ontologies for trust information and relations between different entities in a system. Early works propose simple ontologies or vocabulary extensions to existing ones. Golbeck et al. [69] extend the `FOAF` ontology by adding new specific vocabularies to express levels of trust on persons and/or subjects; Heath and Motta [84] propose an ontology based on `FOAF` and `SKOS` to model trust on the expertise for different people and topics. Later works propose full-fledged ontologies. Some of them are designed with a focus on specific domains. Sherchan et al. [157] introduces a reputation-based *trust service ontology* and a framework for trust management services that considers not only services trust but also trust as a service. Other authors, however, try to model a generic representation for trust. Alnemr et al. [5] presents a *Reputation Object* for achieving reputation interoperability and portability between different domains. In this model reputation is comprised by diverse criteria, each one with different possible values. This ontology is extended by Ceolin et al. [32] to model trust in data, which arguably was not properly addressed. Koster et al. [100], by contrast, define a mathematical framework for trust alignment between agents with dissimilar trust models. This allows the agents to align their subjective trust evaluations and communicate objective information about the interactions these evaluations are based on.

Trust assessment works try to compute trust measures for data or data producers. Those measures can be general or specific for every user. Most of the authors focus on semantic web agents computing trust information on different entities, based on previous experiences and shared information with other agents [46, 138, 69, 48, 67, 92]. Those agents can be automated agents or physical users, while the evaluated entities can be other agents or data (possibly in relation with its provenance).

2.5 Dataset cleaning and enrichment

The Linked Open Data (LOD) Cloud consists of 1,100 datasets, as of 2014 [151]. Although more and more data is being published as linked data, the quality of datasets varies significantly. The LOD Cloud comprises datasets of varying quality, ranging from extensively curated datasets to crowd-sourced data or data extracted from semi-structured or unstructured sources of often relatively low quality. Datasets may contain out of date, incomplete or incorrect data, which might lead to wrong answers to questions. Dataset completeness and accuracy are among the key requirements for QA systems [180].

In order to provide accurate and consistent data, consolidation of different data representations and cleaning of redundant, out-of-date, conflicting, and erroneous information becomes necessary. Data cleaning, also called data cleansing, refers to the process of improving the quality of data by detecting and removing errors and inconsistencies from data [133]. Since data cleaning depends on the application domain, there is no commonly agreed definition of the term. Data cleaning problems can be roughly categorized in to single-source and multi-source problems. Data cleaning is especially required when data is integrated from heterogeneous sources and should be addressed together with other relevant transformations such as mapping data to a common schema. In general, we can distinguish between syntactic and semantic data quality problems. Syntactic quality problems are errors related to the data format including misspelled values, domain format errors, use of undefined abbreviations, and non-standardized formats for values (e.g., using different date format). Semantic errors are violations of the vocabulary (or ontology) such as integrity constraints, duplicated values, use of instances that contradict the domain or range of a property, and contradicting values.

Data cleaning approaches can be categorized into *manual* (crowd-sourced) cleaning, *semi-automatic* cleaning, and *automatic* cleaning. Manual data cleaning is a very tedious task, but results in high quality data if done by experts. Given the typical size of LOD datasets, manual cleaning approaches are infeasible, if not impossible. CROCUS [35], a cluster-based ontology data cleansing framework, provides a semi-automatic approach, for instance level error detection in a dataset. It detects errors in a semi-automatic way, which are afterwards validated by non-expert users called “quality raters”. Automatic cleaning approaches can employ rule-based, query based [99], or machine learning based [2, 130, 19] techniques. Kontokostas et al. [99], presented an approach for test-driven quality assessment of Linked Data using SPARQL query patterns for running data quality tests against RDF knowledge bases. Lehmann et. al. [103] presented an algorithm called DeFacto (Deep Fact Validation) for validating facts by supplying the relevant sources of the fact (provenance information) and confidence scores for the correctness of these facts. The *SDValidate* [130] method exploits the statistical distribution of types and properties to identify possibly wrong statements in a dataset.

Furthermore, we can also classify data cleaning approaches based on the type of data they use as instance based versus schema based data cleaning methods. Instance based cleaning methods do not use the schema of a knowledge base, i.e., they only consider the instances in a dataset. Schema based data cleaning methods use logic-based reasoning methods, such as RDFS or OWL inference, based on the schema of the dataset.

The LOD Cloud comprises many large public datasets in different domains. The heterogeneity of these datasets requires numerous integration steps before they can be used effectively in applications. Data integration frameworks such as the Linked Data Integration Framework (LDIF) [155], the Information Workbench [74], Semantic Web Pipes [123], and ODCleanStore [118] can be used to integrate heterogeneous datasets. LDIF is a framework that translates heterogeneous data from the Web into a homogenized local target. LDIF maintains an integration pipeline starting from data retrieval, continuing with schema mapping, duplicate detection, quality assessment, and concluding with fusion. After retrieving data from the Web, using the R2R mapping language [20], the schema translation phase translates the source vocabularies to a local target vocabulary. Then LDIF uses the SILK [90] identity resolution and linking tool to discover URI aliases and replace them with a single target URI (also adding `owl:sameAs` links to the original sources). The next step is to assess data quality and resolve conflicts from different sources using Sieve [115]. Finally, the output of the integration process as accompanied by provenance information and written in a quads format, i.e., triples are extended by a fourth component that carries the provenance information. ODCleanStore [118] supports the management of Linked Data including tasks such as data cleaning, linking, transformation and quality assessment. As a first step, ODCleanStore carries out data cleaning on each source dataset. ODCleanStore also employs SILK for entity resolution and linking. Unlike LDIF, ODCleanStore uses manually provided trust scores for named graphs and it then computes an aggregated quality score based on the scores of the sources (named graphs). Once duplicate quads have been identified by SILK, ODCleanStore normalizes their subject URIs into a single URI, removes duplicates and then groups quads with conflicts. Then, for each set of conflicting quads, appropriate conflict resolution policies (e.g., MIN, MAX, BEST, AVG, . . .) are applied to a value, considering the trust scores of their source graph. The output of the pipeline is a set of integrated quads accompanied by provenance and quality scores of the source. Bischof et. al. [19] presented a platform for collecting, integrating, and enriching open data about cities. Their Open City Data Pipeline collects, cleans, and integrates various open data sources; then it uses statistical regression methods for predicting and filling in the missing values.

The other main requirement for QA system is completeness of the data, i.e. a complete coverage of the domain. Besides assessing the coverage of a domain (e.g. against a gold standard or reference dataset), coverage quality problems include missing values of entities such as missing values, types, or attribute information. Different enrichment methods can be utilized to complete datasets for the purpose of QA. Enriching datasets can range from extracting new facts from unstructured sources, such as text, to inferring implicit facts from RDF data, to indexing (see Section 2.6) and equipping datasets with linguistic resources.

Gagnon et. al. [62] uses natural language processing techniques to extract RDF triples from literals. The knowledge schema for the representation of extracted knowledge and extraction rules are defined manually. The extractor is based on a set of rules expressed in syntactic patterns and a specification of how RDF triples are to be generated from the dependency trees that will be matched with the patterns. The extraction uses the Stanford Parser[17] to generate the dependency tree from the literals. Manually defining enrichment rules and the sequence of enrichment functions can be a tedious task. Sherif et. al [158] presented a supervised learning approach based on a refinement operator and a self-configuration algorithm that automatically defines RDF dataset enrichment pipelines. Given the pairs of concise bounded descriptions (CBDs) of resources with the enriched version of the same resources, the algorithm learns sequences of atomic enrichment functions that can be applied to generate enriched versions of datasets. SDType [129] can be used for completing missing type information in large knowledge bases automatically extracted from the Web or by crowd-sourcing. These types of datasets contain information that violates the schema definition of the knowledge base or data that do not follow the schema of a dataset. This means, inferencing using traditional reasoning methods may produce

[17] `http://nlp.stanford.edu/software/lex-parser.shtml`

nonsensical results. Like SDValidate, SDType uses the statistical distribution of types and properties to discover the possible type of untyped entity (or additional type for a typed entity). SDType uses links between resources as indicators for types by following a *link-based object classification* approach [65]. SDType employs a link-based weighted voting mechanism to heuristically infer instance types by exploiting other axioms in a knowledge base, in particular links between instances.

Many datasets lack a combination of rich schema and instance data that allows querying and inferencing as well as consistency checking and debugging. Tonon et al. [169] presented a technique for fixing the domain and range of properties using context disambiguation. Bühmann et al. [25] presented a semi-automatic schema construction approach that extracts the most frequently used axiom patterns from existing knowledge bases and converts them to SPARQL-based pattern detection algorithms, which allow for enriching the schemata of the knowledge bases. In the first step, the preparation phase, the frequency of axioms is determined by analyzing several ontology repositories; then, patterns are converted to SPARQL queries. In the execution phase, the actual axiom suggestions are generated. After converting axioms to SPARQL query patterns, entities in the query pattern are replaced with variables, which results in another query pattern. Then, the results are projected to all entities that were replaced by variables, and the frequency of their combination is counted. Finally, the a confidence score of axiom candidates is computed, i.e., the F-measure of each candidate.

Similarly, different data mining approaches, such as association rule mining techniques, can also be used for data and schema construction. In [2], Abedjan and Naumann, introduced "mining configurations", an approach that enable a user to define different rules that helps with predicate suggestion, data enrichment, ontology improvement, and query relaxation. In [3], the authors presented an association rule mining technique for schema and data enrichment. They present two algorithms. The first algorithm suggests new predicates for a given entity by mining predicates. The second algorithm generates new statements for a given dataset by mining predicates and objects, without using external resources.

2.6 Web data indexing

With the recent explosion of data and also non-expert data consumers, the challenges of information retrieval have also increased by leaps and bounds. Systems originally based on keywords matching, nowadays report to return very poor quality results due to the overwhelming outburst of data. An efficient search engine is therefore required to make use of a variety of additional factors for improving performance such as heuristics, linguistic resources and also statistical models.

A typical QA system is empirically only as good as the performance of its indexing module. The performance of indexing serves as an upper bound to the overall output of the QA, since it can process only as much data as is being presented/served to it from the indices. The precision and recall of the system may be good, but if all or most of the top relevant documents are not indexed in the system, the system performance suffers and so does the end user.

Many researchers have compared effectiveness across a variety of indexing techniques. Their studies show improvement if multiple techniques were combined compared to any single individual indexing technique [134]. In the present scenario, information retrieval systems are carefully tailored and optimised to deliver highly accurate results for specific tasks. Over the years, efforts of developing such task specific systems have been diversified based on a variety of factors.

Based on the type of the data and its application settings, a wide range of indexing techniques are deployed. They can broadly be categorized into three categories based on the type of data they index, namely: structured (e.g. RDF, SQL, etc), semi-structured (e.g. HTML, XML, JSON, CSV, etc.) and/or unstructured (e.g. text dumps) data. These are further classified by the type of techniques they use for indexing and/or also by the type of queries which are addressed by a particular technique. These techniques inherently make use of a wide spectrum of underlying fundamental data structures in order to achieve the desirable result.

Most of the systems which deal with unstructured or semi structured data make use of inverted indices and lists for indexing. For structured systems, a variety of data structures such as AVL trees, B-Trees, sparse indices, IR trees, etc., have been developed in the past decades. Many systems combine two or more data structures to maintain different indices for different data attributes. We present a short survey of indexing platforms and data structures used in a wide range of QA systems in the table 1.

System	Data structure used	Platform used for indexing
SWSE/YARS2 [81]	Sparse, Inverted Indices for RDF quads	Lucene
SINDICE [127]	Inverted Index and On-disk persistent storage	Solr
SINA [156]	Bitmap index on RDF quads (toal 5 indices are maintained: 2 full rdf quad indices, 3 partial rdf quad indices)	OpenLink Virtuoso
HAWK [181]	*N/A	*N/A
TBSL [178]	Inverted Index	Solr
EYPHRA [150]	Inverted index	Lemur-Indri
POWER AQUA [106]	Inverted index	Lucene (two indices are prepared taxonomically)
AQUALOG [104]	*N/A	GATE MIMIR architecture (https://gate.ac.uk/sale/tao/index.html), possibly with Lucene
SIG.MA [170]	Inverted Index and On-disk persistent storage	Solr
QUADS [195]	Inverted index	Lucene
MAYA [97]	(key, value) pairs	Traditional index with RDBMS
ESTER [13]	Extended inverted index - inverted index with scores for each word ; combines prefix search and join operations	Proprietary module
QAST [93]	Inverted index with term weighting ((Minimal Span Weighting))	Lucene
FREYA [40]	*N/A	Sesame/OWLIM (aka GraphDB))
QAKIS [26]	*N/A	*N/A
MEANS [14]	Inverted index	Terrier
WATSON/DEEPQA [96, 58]	Persistent disk caching	Watson Explorer Engine XML (VXML)

Table 1: A table comparing the indexing platforms and data structures used by a variety of QA systems. *N/A - to be interpreted as data not available

3 Querying ontologies using natural language

The previous section described how it is possible to find in the Web valuable data and how to make it fit for QA. The *de facto* standard for querying this data is SPARQL. This query language is very powerful but can be used only by experts. For example the question "What is the population of Europe?" can be answered querying DBpedia using the SPARQL query in figure 1.

```
Select ?p where {
      dbr:Europe dbp:populationTotal ?p
      }
```

Figure 1: Example of SPARQL query

In this section we describe how current QA systems make the data, described in the previous section, accessible to the end-user. The idea is to construct a SPARQL query from a question in natural language, whether it is in speech or written format. In the case of speech, a first step is to transform it to text (subsection 3.1); then, text and semantic analysis (subsections 3.2 and 3.3) allow to extract the relevant information from the original string; finally, a query generation stage constructs the query (subsection 3.4). In addition, popular benchmarks for QA (subsection 3.5) and the need for a generic framework to interweave current approaches (subsection 3.7) are also explored.

3.1 Speech recognition

Speech recognition is the process of converting the words spoken by a person and recorded in an audio file or stream to its transcription. In QA systems, the specific task is to recognise the questions uttered by the user. Spoken input can be embedded into a QA system in two ways: one is to interface an Automatic Speech Recognition (ASR) unit with the QA system, considering them as two independent systems; another possibility is to build a unified system where the components of ASR and QA are interdependent. A QA system generally consists of three phases; question processing, passage retrieval and ranking, and answer processing [95]. Since latter two phases do not directly depend on the spoken input, it is not considered in detail here. The goal of the question processing phase is to extract two things from the question: a keyword query and an answer type, a specification of the kind of entity that would constitute a reasonable answer to the question. This involves language understanding tasks such as semantic parsing and applying various query generation rules. A unified system, as discussed above, is usually constructed by combining ASR and language understanding part of QA system, and has similarities with the Spoken Dialog System (SDS). The conceptual model of spoken dialog system [174] consists of i) ASR, as mentioned above, to convert spoken queries to transcriptions, ii) Spoken Language Understanding, which analyses the transcription to extract semantic representations, and iii) a Dialog Manager, which performs semantic interpretation and decides on the best system action according to which further system responses are generated, either as a natural language output, as a result page or as synthesised spoken sentences. The intelligent personal assistants like *Siri*, *Google Now* and *Cortana* follow the architecture of spoken dialog system. The following examples illustrates a QA system and a dialog system. In a dialog system, the context of a conversation is important, because the upcoming user turns might be dependent on the previous turns, whereas in QA systems, the context of a conversation might not play a major role since the task is to generate an answer for a given question.

Example for a QA system

1. user Q1: What is the capital of Germany?
2. spoken QA system: Berlin is the capital of Germany.
3. user Q2: How far is Paris from Bonn?
4. spoken QA system: 510km.

* Q1 and Q2 indicates Question 1 and Question 2 respectively

Example for a Dialog system

1. user T1: book an appointment with Person A at time B
2. personal assistant: Can you tell me the venue?
3. user T2: room C in building D
4. personal assistant: your appointment is booked

* T1 and T2 indicates user Turn 1 and Turn 2 respectively

Conventional speech based QA systems usually concatenate the ASR and QA units [159, 146, 153, 191, 122, 79]. The consequences of this simple concatenation of ASR and QA units are reviewed in [142]. ASR systems are usually built for a specific domain, e.g. for broadcast news or medicine. This implies that the QA system using such a system would also be constrained to a specific domain. The QA system in such a setting is usually based on two paradigms, an information retrieval approach and a knowledge-based approach. Information retrieval means finding unstructured data (i.e. text) that satisfies the user query from large collections of web data. A knowledge-based approach involves a semantic analysis of the query and then the access to structured data [95]. The term "knowledge-based question answering" refers to the idea of answering a natural language question by mapping it to a query over a structured database. The database can be a relational database or follow a simpler data model such as RDF triples. The popular knowledge base DBpedia [21] for example has large numbers of triples derived from Wikipedia infoboxes, the structured tables associated with certain Wikipedia articles. With the evolution of enormous triple-based web data in recent days, an effort is being made to enable knowledge-based spoken dialog systems [85, 110, 173]. On the similar lines of knowledge-based spoken dialog system, a knowledge-based unified QA system can be developed. An interference of knowledge graph helps for a semantic analysis of the recognised query and also to improve the automatic speech recognition accuracy. The semantic analysis involves detection of the domain and intent of the spoken query, named entities in a query and the relation between these named entities. In addition to the knowledge graph, large volumes of multi-domain training data (user query and click logs) exist in web search engines. The breadth of this data is so immense, ranging from simple (weather, finding directions, local events) to more complex (shopping, planning a trip, booking a hotel room). The user actions (e.g. clicks) further provide enough constraints so that the labels (domain, intent, slot) of the query can be inferred by the URIs. The web query click logs and knowledge graph have been employed for domain detection [76], slot filling [172], intent detection [85] and entity extraction [186]. All of these methods make it possible to build multi-domain SDS with less effort on the annotation of data. On the other hand, for statistical modelling we need a manually annotated training and testing dataset.

3.2 Text analysis techniques

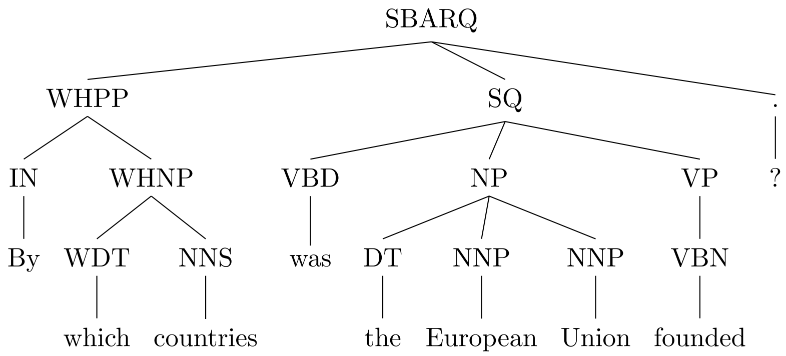
Many QA systems use techniques borrowed from NLP to analyse the natural language questions. We describe some of the typical tools and also discuss their usages.

Named Entity Recognition Named Entity Recognition (NER) tools are tools to identify parts of text that refer to a named entity and classify them in predefined categories like person, organization and location. For example in the question: "When was the European Union founded?" it is important to recognize the phrase "European Union" as one entity. This is for example used in QAKiS [26].

Part-of-speech taggers Many QA systems use Part-of-speech (POS) tagging tools, i.e. tools that assign the corresponding part of speech like verb, noun or determiner to the different words in a given sentence. The observation is that in an ontology, a verb generally corresponds to a relation, a noun to either a subject, object or class and a determiner can often be ignored. The general idea using POS-tags is to obtain some reliable POS tags, which in turn can be used to recognize expressions that refer to relations, subjects, objects and classes. This is for example used in PowerAqua [105] and [17].

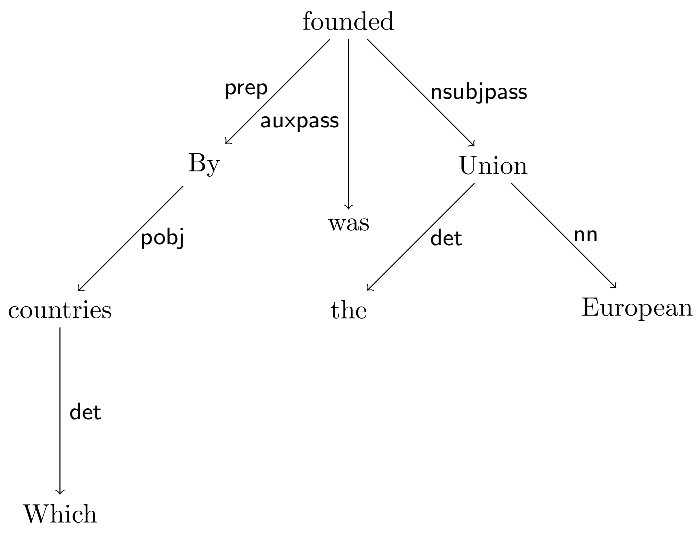
Parsers A parser is based on a formal grammar. A formal grammar consists of symbols (words) and production rules that are used to combine the symbols. Given a sentence, a parser deduces the combination of production rules that generated the sentence according to the underlying grammar. There are mainly two types of grammars that are used in QA systems: phrase structure grammar (PSG) and dependency grammars (DG).

The idea of PSG is to break down a sentence into its constituent parts. The Stanford Parser[18] supports such a grammar. As an example consider the parsing tree of the question ”By which countries was the European Union founded?” returned by the Stanford Parser:



At the bottom of the tree are the words in the question as well as the corresponding POS tags. The tags above denote phrasal categories like: noun phrase (NP), verb phrase (VP), main clause of a wh-question (SQ) and direct question introduced by a wh-phrase (SBARQ). In the phrase structure grammar the production rules define how the POS tags can be combined to form phrasal categories and how to combine phrasal categories to new ones.
These types of trees are used similarly to POS tags, i.e. one tries to find some graph patterns that map to instances, properties or classes with high confidence. Differently from POS tags, one can deduce the dependencies between entities, relations and classes. Parsers of such type are used in the QA systems Intui2 [44], Intui3 [45] and Freya [40]. Another type of PSG is used in TBSL [178].

The idea behind DG is that the words in a sentence depend on each other, i.e. a word ”A” depends on a word ”B”. ”B” is called the head (or governor) and ”A” is called the dependent. Moreover parsers generally also indicate the type of relation between ”A” and ”B”. The Stanford parser supports two types of DG. For one of them the dependency tree of the question ”By which countries was the European Union founded?” is:



There are two main reasons to use dependency parsers. The first is that they make it more easier to extract relations. In the question above the relational phrase “was founded by” is actually scattered in the question making it difficult to extract. This is not the case anymore when using the dependency tree where “was”, “founded” and “by” are connected together. The second reason is that by using the tree it is possible to deduce the relations between the parts of the text, i.e. “was founded by” connects “countries” and “European Union”. There are different strategies to use dependency trees. They are for example used by gAnswer [203] and CASIA [82]. Another type of DG is used for example by Xser[192].

[18] `http://nlp.stanford.edu/software/lex-parser.shtm`

Coreference resolution A problem that can be encountered in questions is that sometimes two or more expressions in a given text can refer to the same entity. For example in the question: "What company did Steve Jobs found after *he* was fired at Apple?" the words "Steve Jobs" and "he" refer to the same entity. In natural language processing, this problem is called Coreference Resolution. A Coreference Resolution tool is for example used by gAnswer [203].

3.3 Semantic analysis

3.3.1 Entity Linking

Given the question "What is the population of Europe?" it is important to identify the concept to which the phrase "Europe" corresponds to. In DBpedia there are for example different possibilities like: dbr:Europe (intended as a continent) and dbr:Europe_(Band) (referring to a rock band called Europe).
To perform this task, a form of "semantic-coherence" is needed, i.e., a measure for the semantic relation between concepts. For example in the above question it is clear for a human that the term "Europe" refers to the continent and not to the band called "Europe". This can be translated by saying that there is no (or a very weak) "semantic-coherence" between the band Europe and population.
There are different approaches to measure the "semantic-coherence" of two concepts A and B. One is to look in the ontology how long is the shortest path between A and B, i.e. the concepts A and B are more related if the shortest path has length 1 than if it is 5. Another approach is to look in a large text corpus how often the concept A appears in the context of B, i.e. for example if A appears in a window of 10 words around B.
The task of identifying named entities in a text and assigning a URI to them, in some ontology, is called entity linking (EL). Tools which do EL to a Wikipedia URI are AIDA [196], DBpedia Spotlight [114] and Bablefy [124]. While these tools map only to the URIs of Wikipedia, the tool AGDISTIS[182] can be configured to link entities in a text to any RDF ontology.
Moreover in QA, one is generally not only interested in mapping entities, but for example also in mapping parts of a question which refer to relations, e.g. map "is the population of" to the URI "dbp:populationTotal". Many different approaches where proposed to solve this problem: the QA system SINA uses a Hidden Markov Model [156], the system DEANNA uses an Integer Linear Program [194] and the system gAnswer uses a subgraph isomorphisem strategy [203].

3.3.2 Semantic Parsing

The purpose of semantic parsing (SP) is to map natural language (NL) input to a formal representation of its meaning. For example, the question *"Who is the wife of Barack Obama"* can be represented as

$$\lambda x.spouse(:\texttt{Barack\_Obama}, x)$$

Such lambda expressions can later be mapped to formal database/knowledge base queries that retrieve the answer to the original question. The task of determining the formal meaning of an NL utterance is a central problem in QA but is also useful in the broader context of human-computer interaction (HCI).

One approach to semantic parsing is to couple syntactic production with semantic composition. In this approach, the semantic representation is essentially constructed as a result of deriving the syntactic structure of the input. Different grammars have been employed to implement this approach. Facebook's Graph Search of 2013[19] used a Weighted Context-Free Grammar (WCFG). Pythia [177] and TBSL [178] rely on a Lexicalized Tree-Adjoining Grammar (LTAG). Combinatory Categorial Grammar (CCG) [163] is popular for semantic parsing [200, 201] because of its transparent interface between syntax and semantics and its ability to capture a wide range of linguistic phenomena. Both LTAG and CCG are classified as mildly context-sensitive and thus more expressive than CFG's. An example of a CCG-based semantic parser that has been evaluated on the QA task is given by Reddy et al. [136], where an ungrounded graph representing the input is created using the popular C&C [37] CCG parser, which is then grounded against a knowledge base (KB). UWSPF (University of Washington Semantic Parsing framework) [8] also focuses on CCG-based semantic parsing that can be used for QA. In the context of Linked Data QA, CCG-based parsing for QA has been explored by Hakimov et al. [75]. Even though such syntax-driven approaches should be able to capture complex constructions, they are less

[19] `https://www.facebook.com/notes/facebook-engineering/under-the-hood-the-natural-language-interface-of-graph-search/10151432733048920`

robust w.r.t. malformed input (telegraphic language like in tweets) as it does not obey normal syntactic rules. In fact, Zettlemoyer and Collins focused on relaxing CCG [201] by adding non-standard combination rules.

Some other approaches to semantic parsing do not rely on phrase structure grammars (PSG). One of the most interesting recent works in this direction is Sempre [17, 16]. Sempre does not rely on a grammar to parse the input, instead taking a more flexible approach that generates a large number of possible interpretations according to a few generation rules and subsequently searches and ranks the most promising interpretations. ParaSempre [16] further builds upon this approach and exploits KB-independent paraphrasing models to improve the ranking. An example of another work on QA in the same spirit is Sina [156], which takes a keyword-based approach to federated query construction. The advantage of such approaches is their higher robustness to malformed or noisy input, however, their ability to correctly process complex expressions is questionable. Other recent notable work that does not rely on a PSG includes Xser [192], which implements a custom phrase-based dependency-style parser that exploits lexical features and semantic labels of phrases recognized in the preceding step.

One of the core problems of any semantic parser is handling the mismatch between natural language and the knowledge available in a KB. Entity Linking (EL) (see Section 3.3.1) is concerned with mapping a phrase to an entity, which entails a (relatively straightforward) matching problem from strings to nodes in a semantic network. A less straightforward problem is to determine how the entities referred to by different phrases in the NL input should relate in the semantic representation built using the target KB. Relations can be explicitly referred to in the input (e.g. `:spouse` in *"Who is the **wife** of Obama?"*) or inferred (e.g. `:writtenBy` in *"Give me all Tolstoy plays"*). Determining which relation is referred to in the first case can benefit from a lexicon (like in EL) but may require inference based on context and knowledge in the second case. Sometimes the NL expression of a property for some entity cannot be mapped in a straightforward way, requiring the translation from a canonical representation to a KB-specific representation that can grow complex. Some NL expressions may map to complex structures (e.g. "polyglot" $\rightarrow \lambda x.person(x) \wedge \texttt{COUNT}(\lambda l.speaks(x,l) \wedge language(l)) > 1$) and sometimes a relation is expressed through more complex phrases (e.g. idioms like "kick the bucket"). In addition to recognizing relations, also operators (aggregation, comparison) must be recognized (e.g. *"number of mountains higher than Elbrus"*). Useful resources of lexical information are relation extraction patterns like in BOA [64] (used in TBSL [178]) or ReVerb [54] (used in Sempre [17, 16]). (A part of) the lexicon can also be specified manually [177, 178, 179].

Finding the argument structure is another core task in semantic parsing. We need to determine how the elements (relations, entities, types, operators) referred to in the input should be put together to represent the meaning correctly. Non-trivial linguistic constructions (e.g. ellipses) may pose a challenge, especially when the input is not well-formed. CCG-based approaches are particularly advantageous in this respect, but, as already mentioned, they can suffer from a low robustness w.r.t. malformed or noisy input. On the other hand, approaches like Sempre [17, 16] and Sina [156] are more robust but do not exploit syntactic features and thus are less able to handle complex language. The ideal approach to determine argument structure would leverage both syntactic and semantic features to process complex constructions and enable robustness to malformed input by exploiting knowledge. Some approaches separate argument structure finding and relation finding (e.g. [192, 136]) for the sake of KB independence of the argument structure decision. Other work uses KB-specific lexica for parsing (e.g. [17, 16, 200, 201]). In general, though, relation finding can be considered inherent to semantic parsing because of the unique character of the problem and its higher co-dependence with the argument structure decision. Moreover, argument structure finding could substantially benefit from the knowledge contained in a KB when done together with relation finding.

A statistical parser needs to be trained and the availability of suitable training data is another problem. A dataset consisting of sentences annotated with their formal representation is the most straightforward training source (e.g. as required by [200] for CCG parsing). However, such datasets are costly to obtain and recent research efforts have focused on reducing/simplifying the supervision required for training. For example, Sempre [17] is trained on question-answer pairs, which are easier to obtain, and ParaSempre [16] tries to improve the ranking of candidate interpretations by exploiting KB-independent paraphrasing models learnt from purely textual corpora. Another method to improve training is to use better features. Especially promising is the application of the deep learning paradigm by modelling language and knowledge using embedding techniques. This recently invigorated field has produced interesting results for relation extraction and KB completion (e.g. [63, 34, 139, 189]) as well as language modelling (e.g. [120, 121, 88]).

Example of a CCG parse: As an illustration, here we present an example of a CCG-based parse of the sentence *"Barack married Michelle."* that produces a logical representation. CCG operates on a small number of combination rules that can be divided in (1) application, (2) type-raising and (3) functional composition. During CCG parsing, first the tokens in the input are assigned categories, in our example those are:

1. *Barack*: NP
2. *married*: (S\NP)/NP
3. *Michelle*: NP

The NP categories for Barack and Michelle bear the same meaning as in POS tagging. The (S\NP)/NP is shorthand notation for something that will become something that will become a sentence after being left-applied with an NP, after being right-applied with an NP. So that when *"Michelle"* is applied with *"married"* from the right, *"married Michelle"* becomes S\NP, which after left-application with *"Barack"* becomes an S (sentence). This gives the following derivation for the sentence:

$$\dfrac{\dfrac{Barack}{NP : \texttt{Barack\_Obama}} \quad \dfrac{\dfrac{married}{(S\backslash NP)/NP : \lambda\texttt{xy.spouse(y,x)}} \; \dfrac{Michelle}{NP : \texttt{Michelle\_Obama}}}{S\backslash NP : \lambda\texttt{y.spouse(y,Michelle\_Obama)}} >}{S : \texttt{spouse(Barack\_Obama,Michelle\_Obama)}} < \tag{1}$$

For semantic parsing, the challenge is in learning the lexicon, including category assignment as well as the parameters that control parsing decisions.

3.4 Query construction

Given an interpretation of some NL input, formally represented, a query can easily be constructed using a set of rules, assuming the query language is expressive enough to handle the formalism of the representation. For example, $\lambda x.spouse(Barack\_Obama, x)$ is mapped to the following SPARQL query: `SELECT ?x WHERE {:Barack_Obama :spouse ?x}`. Code for SPARQL query generation is provided as part of the work on Sempre [17, 16].

A non-trivial aspect in query construction is federated querying, i.e. constructing queries that contact multiple KBs to retrieve the answer. Most QA systems query only one KB (with Sina [156] and PowerAqua [105] among the exceptions). while the idea behind Linked Data assumes a decentralized cloud of interlinked KBs with different focus domains. In order to query the Linked Data Cloud, care must be taken of determining which information can be retrieved from which KB. Also the querying strategy is important to avoid unnecessary load and ensure efficiency. One approach, as taken by Sina [156], sends a query to all target KBs. However, better approaches exploiting information content features of KBs should be feasible.

3.5 Benchmarks

There are three popular benchmarks for QA: Free917, WebQuestions and QALD. All of them offer a set of training questions, a set of test questions and some fixed evaluation metrics. Free917[27] consists of 917 questions that can be answered using Freebase. The benchmark contains pairs of questions and semantically equivalent logical forms. The idea behind the benchmark is to use machine learning techniques to learn how to translate a question to an equivalent logical form. WebQuestions[17] contains 5810 questions that can also be answered using Freebase. The benchmark contains questions and the corresponding answers. Here the idea is to learn how to construct a query using question-answer pairs. QALD is a series of yearly benchmark started in 2011. Each year a benchmark is proposed with a specific focus like: addressing multilinguality, questions over interlinked ontologies and hybrid search.

3.6 Interweaving decisions

The order of different interpretation decisions is one of the characterizing aspects of a QA system. For example, one can first detect phrases, then determine the argument structure and then ground the phrases in the structure to a KB (as is done in Xser [192]). Another way could be to first disambiguate phrases in the input and then arrange the pieces into a structure.

Classic QA takes a pipelined approach, where first a certain task is performed and its results are used as features in a following task (e.g. NER in Xser [192] that feeds to phrase tagging). However, such a pipelined approach is prone to error propagation. Errors made in the early steps will most likely cause errors in following steps that depend on the results of the earlier steps. Whichever way the pipeline is constructed, one-way dependence of one task upon another will cause error propagation. Of course, one could arrange the steps in such a way that less error-prone tasks are performed earlier. Another way is re-ranking the different possible interpretations, however, if the right interpretation is not among the candidates, the errors will still not be recovered.

Another way to address the problem of error propagation and to improve the results of the whole task is joint modelling. A prime example of joint models is the work of Singh et al. [160]. In this work, a joint model for (1) entity tagging, (2) relation extraction and (3) coreference resolution is investigated. The authors first present separate models for each of the three tasks and then model the dependencies between the three models by connecting the three tasks into one factor graph.

The advantage of joint modelling for complex tasks such as QA is evident: different subtasks of the complex task can have dependencies among each other (e.g. linking a phrase to some entity provides some indications for coreference resolution and vice-versa). Joint inference enables us to solve such a complex task with better results by deciding for the different subtasks *simultaneously* (as opposed to solving them in a pipeline), allowing each subtask's decision to take into account the decisions of the other subtasks. This idea has recently gained more traction, with the work of Singh et al. [160] and the QA system CASIA [82] being some of the interesting examples.

Probabilistic Graphical Models (PGM) are straightforward ways for joint modelling but other techniques (such as Integer Linear Programming (ILP)) can be used as well. One of the challenges of joint modelling is the bigger solution space resulting from joining multiple models into one. Recent research has focused on efficient inference and training of large PGMs (e.g. the work of Sutton and McCallum [167, 168]) but it still remains an interesting area of investigation. Another challenge is the availability of suitable training data. Determining the parameters of the parts of a joint model that model inter-dependencies between different parts requires training data that provide supervision for all those different parts we wish to join.

Joint inference might prove an interesting direction of research and development. It could also be of particular interest for our architecture to enable easy development and joining of models by providing supporting facilities in the framework.

3.7 Architecture for query processing

The Web of Data has attracted the attention of the QA community and recently, a number of QA systems that take advantage of web data and are aware of the schemas of such data have been introduced. In this subsection, we describe different state-of-the-art QA systems and their limitations, motivating the need for a modular, message driven architecture.

QA is a multidisciplinary field; it bridges artificial intelligence, information retrieval, and knowledge management. Recently, the QA community paid considerable attention to adapting and improving QA systems by taking the Web of Data into account. These attempts led to a new generation of data-aware QA systems. These systems can be distinguished by their scope of applicability and their approaches. *Closed domain* QA systems consider a specific domain to answer a question. These QA systems are often limited to a specific Knowledge Base (KB), for example one containing medical facts [1]. The advantage of limiting a system's scope to an explicit domain or ontology is that there is less chance of ambiguity and a high accuracy of answers. On the other hand, it is difficult and costly to extend closed domain systems to a new domain or reusing them in implementing a new system. To overcome the limitations of closed domain QA systems, researchers have shifted their focus to *open domain* QA systems. The latter use publicly available semantic information resources such as FreeBase [22] or DBpedia [21] to answer questions.

Other types of QA systems described in [165] extract answers from an unstructured corpus (e.g. news articles), or other various form of documents available on the Web. They are known as *corpus based* QA systems. QuASE [165] is such a corpus based QA system that mines answers directly from the Web. In 2011, the yearly benchmark series QALD (Question Answering over Linked Data) was introduced.[20] In the latest advancements, QALD now focuses on hybrid approaches using information from both structured and unstructured data. Many developers of open domain QA systems now use QALD as a benchmark for their evaluation. PowerAqua [105] is an ontology based QA system which answers the question using information that can be distributed across heterogeneous semantic resources. FREyA [40] is another QA system that increases precision by learning users query formulation preferences. It also focuses on resolving ambiguity while using natural language interfaces.

[20] `http://www.sc.cit-ec.uni-bielefeld.de/qald/`

QAKiS [26] is an QA system that matches fragments of the question with binary relations of the triple store to address the problem of question interpretation by relation-based matching. SINA [156] is a semantic search engine that can process both keyword-based and natural language queries. It uses a Hidden Markov Model for disambiguating mapped resources and then applies forward chaining to generate formal queries. It formulates graph pattern templates using the knowledge base. TBSL [178] is a template based QA system over linked data that matches a question against a specific SPARQL query. It combines natural language processing capabilities (NLP) with linked data to produce good results w.r.t. the QALD benchmark.

Besides domain specific question answering, QA systems can be further classified according to the type of question (input), its data sources (structured or unstructured data), and according to traditional intrinsic challenges posed by the search environment (scalability, heterogeneity, openness, etc.) [108].

Many systems have been designed for a particular input type. For example, DeepQA of IBM Watson [125], Swoogle [47], and Sindice [127] focus on keyword-based search, whereas the system described in [153] integrates QA and automated speech recognition (ASR). Similarly, there are several examples of QA based on different sources used to generate an answer, such as Natural Language Interfaces to Data Bases (NLIDB) [6] and QA over free text.

Earlier in this section, we have observed that the field of QA is growing and new advancements have been made for each existing approach over a short period of time. However, there is a need of an open framework for generating QA systems that integrate different state-of-the-art approaches. We now discuss some approaches for establishing an abstraction of QA systems and semantic search. Research presented in [176] describes a search ontology that provides an abstraction over a user's question. Users can create complex queries using this ontology without knowing the syntax of the query. This approach provides a way to specify and reuse search queries, but the approach is limited to defining properties represented within the ontology. Using the search ontology, the user can neither define the dataset that should be used nor other specific properties. The open source QALL-ME framework [57] is an attempt to provide a reusable architecture for a multilingual, context aware QA framework. QALL-ME uses an ontology to model structured data of a targeted domain. This framework is restricted to closed domain QA, and exposes limitations w.r.t. extensions towards heterogeneous data sources and open domain QA systems. The openQA [113] framework is an architecture dedicated to implementing a QA pipeline. The implementation of the QA pipeline is limited to the Java programming language.

For implementing an open QA framework, it is important to define a generalized vocabulary of QA terms. This vocabulary will serve as an abstraction level on top of all the existing QA approaches and will provide interoperability and exchangeability between them. The generalized vocabulary can be further used to integrate different components and web services within a QA system and to address the whole process of QA from the provision of data to communicating the answer to the end user.

For example, DBpedia Spotlight [114] is an open source web service for annotating text documents with DBpedia resources. AIDA [196] is a similar project which uses the YAGO [164] ontology. While the last two examples address only specific ontologies, AGDISTIS [182] is an approach for named entity disambiguation that can use any ontology. To leverage the capabilities of different available web services and different tools for QA systems, a generalized vocabulary is needed.

Considering the mentioned state-of-the-art QA systems and frameworks, it is clear that three groups of problems exist:

1. *Lack of a generic conceptual view on QA systems:* While there are many different architectures for QA systems (e.g. [165, 153, 113, 107]) , most of them are tailored to specific and limited use cases as well as applications. Reusability and interoperability has not (sufficiently) been considered in designing these approaches.

2. *No standardized message format for QA systems:* While existing QA systems employ plenty of available tools and services, general interoperability is still not ensured due to a missing message format. However, there might be great synergy effect in creating QA systems that combine different tools. For example, in a given architecture, Named Entity Recognition (NER) and Named Entity Disambiguation (NED) might be integrated in a single tool. NER solutions might be evolved and thus implemented in either a novel way (e.g. in [114]) or employ existing tools (e.g. the Stanford NER [59]). Thus, integrating a (new) NER approach without a standardized message format is also cumbersome and time consuming. However, integrating different components is difficult and causes a lot of effort for each new implementation.

3. *Scalability and Coverage problem:* Existing schema-driven approaches mainly focus on the input query

(usually even limited to textual representations of input queries), and are not sufficiently flexible to fulfill emerging demands which have not yet been discovered (e.g. [176]).

It is important to note that most of the available QA systems have limited reusability and extensibility in other QA approaches. Hence, considering the challenges of QA systems there is the need of a generalized approach for architecture or ontology of QA systems and semantic search to be able to integrate all state-of-the-art advancements of QA under a single umbrella. While many of these systems have achieved significant performance in special use cases, a shortage was observed in all of them. We figured out that existing QA systems suffer from the following drawbacks: (1) the potential of reusing their components is very low, (2) extension of the components is problematic, and (3) interoperability between the employed components is not systematically defined. Therefore, there is a need for a descriptive approach that defines a conceptual view of QA systems. The approach must cover all needs of current QA systems and must be abstracted from implementation details. For this, there is the need for a message-driven vocabulary built on an abstract level. The vocabulary should implement lessons learned from conceptual views on different QA systems and will act as a first step towards our aim of designing a message driven QA architecture.

4 Preparation and presentation of answers

After the query has been processed and the data for the answer is gathered, the next step is to process this data for generating an answer in natural language for the user. This section showcases relevant State of the Art that can be useful for that task. First, we overview Q&A communities (subsection 4.1), where user interaction can provide useful information for modelling an automatic QA system. Then, we tackle text generation techniques (subsection 4.2), which can serve as a basis for answer generation in a QA environment.

4.1 User-to-user interaction for QA

Understanding the context of Q&A communities Nowadays, Internet is the mean where people express their ideas, publish their information, disagree with other people and most importantly access the public information and a wide range of resources. It is remarkable that Internet has become a ubiquitous platform and users can access it with ease. Q&A websites play a significant role on the people's lives, thus users tend to interact with them on a regular basis.

Q&A sites can be considered as communities of users and participants, providing services similar to the traditional institutions. In the same way that existing organizations provide useful information and answers, Q&A sites can produce faster, more accurate and non-expensive services to the crowd. This reason makes Q&A sites popular and users derive a great amount of question-answering knowledge, which is increasing constantly the latest years. These repositories of data started initially as platforms for exclusive question-answering which evolved gradually in a driven social community of knowledge which process a great deal of information. Nowadays, modern Q&As sites embed an integrated social machine platform and thus participants can interact, connect together and follow the context generated by the crowd.

Content Analysis on Q&As as a provision to an effective User Interaction Interface This evolution of Q&As sites has spotted the attention of the researchers and the analysis they conduct plays a powerful role in the implementation of future QAs applications. Thus, there comes a great need of understanding and examining the already thriving popular Q&A sites. Doing this will evaluate a lot of aspects of QA regarding the user behaviour, user interaction and users' needs. Given these studies, there is clearly the opportunity to develop the ideal QA system and broaden the current ones. On top of that, designers will benefit from the knowledge related to the users activity [56] and entropy and would ameliorate the overall experience. Combining all of these ingredients make Q&A sites an interesting case of study for both the researchers and the participants.

There are lots of principles and methods that can produce useful results and insights by studying those sites. Overall, the massive demand is focused on the content created by these communities. Firstly, the analysis will reveal the interaction patterns between the participants [41] and the empathy towards certain subjects of the repositories. Second, it will outcome notable differences in the accuracy of the replies and the plenitude of them. Furthermore, the content analysis will find correlations between the activity of the active and passive users and will yield ideas on how people choose their topics and answers recursively among all categories. Likewise, analysing the content and underlying social network can offer different ways to filter and rank answers. For example, those ways can vary from identifying similar questions answered in the past to understanding the vocabulary that is used to formulate the questions and reuse it in an improved way. Besides, content analysis could also determine the patterns that experts follow to validate answers and identify their criteria of doing that. Another notable case would be to analyse why platforms such Google Answers failed to grow and the shut down was unexpected. Results from current researches have shown that these studies can lead to an effective prediction of the future implementations of Q&A sites. Finally, for all of the reasons mentioned above, we suggest that studying the Q&A sites will lead to a new domain where experts could highlight the needs of the modern society and develop the optimal platform for QA.

To a better understanding of Q&A sites there is a variety of online repositories such as Yahoo Answers, Quora[185], Stack Overflow, Ask, Knowledge Search etc. that have been studied with different approaches in respect to the context of the knowledge. Firstly, in the case of Yahoo Answers[4] researchers seek to understand the knowledge sharing activity by clustering the content to characteristics and interactions between the users. Results of this study were able to characterize the users' interests and predict the popularity of the new questions when posed. Quora is more focused on questions with a wide variety of topics, it's study is trying to understand what drives Quora's growth until now and and the way it continues attract users. Quora follows a different community style and its structure differs from common community sites integrating a social machine where connections and networks can be observed through the participants' interactions. Other popular studies and efforts related to Q&A sites mostly try to understand them [109] and try to improve their performance and yield new strategies for the benefit of the community. Many researchers are fond of analysing the behaviour of the participants in anonymous communities [18] while others are examining the data integrity and how users react to the lack of this. In the last years, Q&A sites have evolved and grew dramatically due to the attempts of the researchers to understand how people express themselves, what drives their motivation and why many of them prefer to observe than participate.

The outcomes of the research questions deriving from these studies will contribute to the design of modern QA applications and actually we propose and consider this content analysis as a compulsory precondition before proceeding to the implementation of a realistic QA system.

4.2 Text Generation

Natural Language Generation is a sub-field of computational linguistics and artificial intelligence and is the way of visualising information in the form of textual, humanly-perceivable information. For many years template-based approaches were regarded as dominant for the implementation of NLG systems [137, 135, 36]. As Reiter describes in [137], these systems carry out their basic functionality in three different phases: (i) document planning, (ii) microplanning, and (iii) realisation. Each one of the above mentioned stages is associated almost explicitly not only with the domain of the end-application but, in most cases, with the application itself. The difficulty of transferring an NLG system across different domains or languages along with their native limitations for generating non-generic content are considered the fundamental reasons that prevented these systems for becoming widely accepted [161].

Language Modeling with Neural Networks Among the statistical-based approaches for language modelling, neural network implementations have exhibited promising results over the recent years. Language models based on neural networks outperform most of the state-of-the-art statistical approaches, such as n-gram models [15]. Neural network language models can facilitate text generation purposes and due to their statistical nature they are absolved from most of the domain or language limitations. Their inherent capability of learning the syntactical and grammatical, in the case of word-level implementations, features of a language that allows them to generate text has been discussed in the recent literature [166, 70, 162].

In 2003 Bengio [15] introduced the first neural network approach in the field of language modelling. The proposed feed-forward neural net achieved better perplexity than the state-of-the-art n-gram models. Subsequent literature [119, 166, 70] employs architectures based on Recurrent Neural Networks (RNN) variants and demonstrates that due to their intristic ability to cycle information inside the nodes of their high-dimensional hidden state, RNNs are extremely powerful sequence models. In contrast with feed-forward-based models that are based on the Markov assumption, RNNs are capable of storing information in the form of high-dimensional distributed representations [112].

Context-Sensitive Response Generation The general concept of a software capable of participating in human-computer conversations has been initially proposed by Weizenbaum [190]. Weizenbaum implemented ELIZA, a keyword-based program that was achieving its functionality in "the MAC time-sharing system at MIT"

[190] and set the basis for all the descendant *chatterbots* that were subsequently developed. Many template-based approaches [89, 184] have been suggested in the scientific literature, as a way of transforming the computer into a competitive conversation participant. However, they usually adopt variants of the nearest-neighbour method to facilitate their response generation process from a number of limited sentence paradigms and, as a result, they are limited to specific topics or scenarios of conversation. Ritter [141], in 2011, has introduced the idea of text generation in the form of creating meaningful responses to a conversational incentive. By experimenting with conversations that have been extracted from Twitter, he demonstrated that models for Statistical Machine Translation (SMT) are able to exhibit promising results in the field of short-length response generation. It is also worth noting that by escaping template-based paradigms, Ritter's novel approach is the first that actually illustrates the prospect for conversation-aware dialogue systems, capable of multi-domain applicability.

In 2015, Sordoni [162] extended the premises of Ritter's work on the context-sensitive conversation systems by introducing a novel RNN-based approach for response generation. The context vectors are computed every single time that a response is generated and are formed from processing the sequence of comments that have been published prior to the message, to which a response is expected, along with the message itself. It should be noted that, in contrast with Ritter's approach [141], the response generation is not based explicitly on the last message but it takes into consideration the whole sequence of utterances in a conversation. By evaluating the generated responses with the BLEU[1] and METEOR[2] [101] metrics, they exhibited significant improvements over the state-of-the-art SMT-based methods [141] and demonstrated that properly-modified neural network language models can play a significant role in the field of conversational response and question-answering systems.

5 Conclusion

The Semantic Web technologies have fostered the research on a new paradigm of QA systems, based on the huge quantities of available data. The WDAqua project is an innovative approach to globally tackle all the different dimensions of such systems to provide a general QA solution. The applied results of this project will make the Semantic Web data available to the population, while the research results will advance the State of the Art in related research fields. This document is a first step in that direction, where we review current Web and Data Science techniques for QA and discussed their relevance for data-driven QA systems. We have focused on current challenges and, when they exist, on different approaches to tackle them. To the best of our knowledge, this is first time such a global approach on the state-of-the-art for QA systems has been performed.

References

[1] Asma Ben Abacha and Pierre Zweigenbaum. Medical question answering: translating medical questions into sparql queries. In *ACM International Health Informatics Symposium, IHI '12, Miami, FL, USA, January 28-30, 2012*, pages 41–50, 2012. doi: 10.1145/2110363.2110372. URL `http://doi.acm.org/10.1145/2110363.2110372`.

[2] Ziawasch Abedjan and Felix Naumann. Improving rdf data through association rule mining. *Datenbank-Spektrum*, 13(2):111–120, 2013.

[3] Ziawasch Abedjan and Felix Naumann. Amending RDF entities with new facts. In *Proceedings of the 3rd Workshop on Knowledge Discovery and Data Mining Meets Linked Open Data co-located with 11th Extended Semantic Web Conference (ESWC 2014), Crete, Greece, May 25, 2014.*, 2014. URL `http://ceur-ws.org/Vol-1243/paper3.pdf`.

[4] Zhang Adamic and Ackerman Bakshy. Knowledge sharing and yahoo answers: Everyone knows something. In *Proceedings of the 17th International World Wide Web Conference*, pages 665–674. WWW, 2008.

[5] Rehab Alnemr, Adrian Paschke, and Christoph Meinel. Enabling reputation interoperability through semantic technologies. In *Proceedings of the 6th International Conference on Semantic Systems*, page 13. ACM, 2010.

[1] BLEU (Bilingual Evaluation Understudy) [128] is an automatic evaluation metric for measuring the quality of generated text.
[2] METEOR (Metric for Evaluation of Translation with Explicit ORdering) [101] is an evaluation metric for measuring the quality of machine-translated text.

[6] Ion Androutsopoulos, Graeme D Ritchie, and Peter Thanisch. Natural language interfaces to databases–an introduction. *Natural language engineering*, 1(01):29–81, 1995.

[7] Donovan Artz and Yolanda Gil. A survey of trust in computer science and the semantic web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):58–71, 2007. doi: 10.1016/j.websem.2007.03.002.

[8] Yoav Artzi and Luke Zettlemoyer. Uw spf: The university of washington semantic parsing framework. *arXiv preprint arXiv:1311.3011*, 2013.

[9] Sören Auer, Jens Lehmann, and Axel-Cyrille Ngonga Ngomo. Introduction to linked data and its lifecycle on the web. In *Reasoning Web*, pages 1–75, 2011.

[10] Collin F Baker, Charles J Fillmore, and John B Lowe. The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics, 1998.

[11] Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. Open information extraction for the web. In *IJCAI*, volume 7, pages 2670–2676, 2007.

[12] Michele Banko, Oren Etzioni, and Turing Center. The tradeoffs between open and traditional relation extraction. In *ACL*, volume 8, pages 28–36, 2008.

[13] Holger Bast, Alexandru Chitea, Fabian M Suchanek, and Ingmar Weber. ESTER : Efficient Search on Text , Entities , and Relations. *Search*, (2):671–678, 2007. doi: 10.1145/1277741.1277856. URL `http://dblp.uni-trier.de/db/conf/sigir/sigir2007.html#BastCSW07`.

[14] Asma Ben Abacha and Pierre Zweigenbaum. MEANS: A medical question-answering system combining NLP techniques and semantic Web technologies. *Information Processing & Management*, 51(5):570–594, 2015. ISSN 03064573. doi: 10.1016/j.ipm.2015.04.006. URL `http://linkinghub.elsevier.com/retrieve/pii/S0306457315000515`.

[15] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155, March 2003. ISSN 1532-4435. URL `http://dl.acm.org/citation.cfm?id=944919.944966`.

[16] Jonathan Berant and Percy Liang. Semantic parsing via paraphrasing. In *Proceedings of ACL*, volume 7, page 92, 2014.

[17] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on freebase from question-answer pairs. In *EMNLP*, pages 1533–1544, 2013.

[18] Hernandez Bernstein, Andre Harry, and Vargas Panovich. 4chan and /b/:an analysis of anonymity and ephemerality in a large online community. In *Proceedings of the 5th International AIII Conference on Weblogs and Social Media*, page 8. AIII, 2011.

[19] Stefan Bischof, Christoph Martin, Axel Polleres, and Patrik Schneider. Collecting, integrating, enriching and republishing open city data as linked data. In Marcelo Arenas, Óscar Corcho, Elena Simperl, Markus Strohmaier, Mathieu d'Aquin, Kavitha Srinivas, Paul T. Groth, Michel Dumontier, Jeff Heflin, Krishnaprasad Thirunarayan, and Steffen Staab, editors, *International Semantic Web Conference (2)*, volume 9367 of *Lecture Notes in Computer Science*, pages 57–75. Springer, 2015. ISBN 978-3-319-25009-0. URL `http://dblp.uni-trier.de/db/conf/semweb/iswc2015-2.html#0002MPS15`.

[20] Christian Bizer and Andreas Schultz. The r2r framework: Publishing and discovering mappings on the web. In Olaf Hartig, Andreas Harth, and Juan Sequeda, editors, *COLD*, volume 665 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2010. URL `http://dblp.uni-trier.de/db/conf/semweb/cold2010.html#BizerS10`.

[21] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. Dbpedia-a crystallization point for the web of data. *Web Semantics: science, services and agents on the world wide web*, 7(3):154–165, 2009.

[22] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM, 2008.

[23] Kurt D. Bollacker, Robert P. Cook, and Patrick Tufts. Freebase: A shared database of structured general human knowledge. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*, pages 1962–1963, 2007. URL `http://www.aaai.org/Library/AAAI/2007/aaai07-355.php`.

[24] Dan Brickley and Ramanathan V Guha. Rdf schema 1.1. *W3C Recommendation*, 25:2004–2014, 2014.

[25] Lorenz Bühmann and Jens Lehmann. Pattern based knowledge base enrichment. In Harith Alani, Lalana Kagal, Achille Fokoue, Paul T. Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha F. Noy, Chris Welty, and Krzysztof Janowicz, editors, *International Semantic Web Conference (1)*, volume 8218 of *Lecture Notes in Computer Science*, pages 33–48. Springer, 2013. ISBN 978-3-642-41334-6. URL `http://dblp.uni-trier.de/db/conf/semweb/iswc2013-1.html#BuhmannL13`.

[26] Elena Cabrio, Julien Cojan, Alessio Palmero Aprosio, Bernardo Magnini, Alberto Lavelli, and Fabien Gandon. Qakis: an open domain qa system based on relational patterns. In *11th International Semantic Web Conference ISWC 2012*, page 9. Citeseer, 2012.

[27] Qingqing Cai and Alexander Yates. Large-scale semantic parsing via schema matching and lexicon extension. In *ACL (1)*, pages 423–433. Citeseer, 2013.

[28] G Carothers. Rdf 1.1 n-quads: A line-based syntax for rdf datasets. *W3C Recommendation*, 2014.

[29] Jeremy J Carroll. Signing rdf graphs. In *The Semantic Web-ISWC 2003*, pages 369–384. Springer, 2003. doi: 10.1007/978-3-540-39718-2_24.

[30] Jeremy J Carroll and Patrick Stickler. Rdf triples in xml. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pages 412–413. ACM, 2004. doi: 10.1145/1013367.1013501.

[31] Jeremy J Carroll, Christian Bizer, Pat Hayes, and Patrick Stickler. Named graphs, provenance and trust. In *Proceedings of the 14th international conference on World Wide Web*, pages 613–622. ACM, 2005. doi: 10.1145/1060745.1060835.

[32] Davide Ceolin, Archana Nottamkandath, Wan Fokkink, and Valentina Maccatrozzo. Towards the definition of an ontology for trust in (web) data. In *10th International Workshop on Uncertainty Reasoning for the Semantic Web (URSW 2014)*, page 73, 2014.

[33] Nathanael Chambers and Dan Jurafsky. Template-based information extraction without the templates. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 976–986. Association for Computational Linguistics, 2011.

[34] Kai-Wei Chang, Wen-tau Yih, Bishan Yang, and Christopher Meek. Typed tensor decomposition of knowledge bases for relation extraction. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1568–1579, 2014.

[35] Didier Cherix, Ricardo Usbeck, Andreas Both, and Jens Lehmann. CROCUS: Cluster-based ontology data cleansing. In *Proceedings of the 2nd International Workshop on Semantic Web Enterprise Adoption and Best Practice*, 2014. URL `http://jens-lehmann.org/files/2014/wasabi_crocus.pdf`.

[36] Alexander Clark, Chris Fox, and Shalom Lappin. *The Handbook of Computational Linguistics and Natural Language Processing*. Wiley-Blackwell, 2010.

[37] Stephen Clark and James R Curran. Wide-coverage efficient statistical parsing with ccg and log-linear models. *Computational Linguistics*, 33(4):493–552, 2007.

[38] Michael Collins and Nigel Duffy. Convolution kernels for natural language. In *Advances in neural information processing systems*, pages 625–632, 2001.

[39] Aron Culotta and Jeffrey Sorensen. Dependency tree kernels for relation extraction. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 423. Association for Computational Linguistics, 2004.

[40] Danica Damljanovic, Milan Agatonovic, and Hamish Cunningham. Freya: An interactive way of querying linked data using natural language. In *The Semantic Web: ESWC 2011 Workshops*, pages 125–138. Springer, 2012.

[41] Maarten de Laat. Network and content analysis in an online community discourse. In *Proceedings of the Conference on Computer Support for Collaborative Learning: Foundations for a CSCL Community*, page 2. CSCL, 2002.

[42] Jeremy Debattista, Santiago Londoño, Christoph Lange, and Sören Auer. Luzzu-a framework for linked data quality assessment. *arXiv preprint arXiv:1412.3750*, 2014.

[43] Jeremy Debattista, Christoph Lange, and Sören Auer. Luzzu quality metric language–a dsl for linked data quality assessment. *arXiv preprint arXiv:1504.07758*, 2015.

[44] Corina Dima. Intui2: A prototype system for question answering over linked data. *Proceedings of the Question Answering over Linked Data lab (QALD-3) at CLEF*, 2013.

[45] Corina Dima. Answering natural language questions with intui3. In *Conference and Labs of the Evaluation Forum (CLEF)*, 2014.

[46] Li Ding, Lina Zhou, and Timothy W Finin. Trust based knowledge outsourcing for semantic web agents. In *IEEE / WIC International Conference on Web Intelligence*, pages 379–387, 2003. doi: 10.1109/WI.2003.1241219.

[47] Li Ding, Timothy W. Finin, Anupam Joshi, Rong Pan, R. Scott Cost, Yun Peng, Pavan Reddivari, Vishal Doshi, and Joel Sachs. Swoogle: a search and metadata engine for the semantic web. In *Proceedings of the 2004 ACM CIKM International Conference on Information and Knowledge Management, Washington, DC, USA*, pages 652–659, 2004. doi: 10.1145/1031171.1031289. URL `http://doi.acm.org/10.1145/1031171.1031289`.

[48] Li Ding, Pranam Kolari, Tim Finin, Anupam Joshi, Yun Peng, and Yelena Yesha. On homeland security and the semantic web: A provenance and trust aware inference framework. In *AAAI Spring Symposium: AI Technologies for Homeland Security*, pages 157–160, 2005.

[49] Renata Queiroz Dividino, Simon Schenk, Sergej Sizov, and Steffen Staab. Provenance, trust, explanations-and all that other meta knowledge. *KI*, 23(2):24–30, 2009.

[50] Cıcero Nogueira dos Santos, Bing Xiang, and Bowen Zhou. Classifying relations by ranking with convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pages 626–634, 2015.

[51] Edd Dumbill. Tracking provenance of rdf data. Technical report, Technical report, ISO/IEC, 2003.

[52] Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[53] Fredo Erxleben, Michael Günther, Markus Krötzsch, Julian Mendez, and Denny Vrandečić. Introducing wikidata to the linked data web. In *The Semantic Web–ISWC 2014*, pages 50–65. Springer, 2014. doi: 10.1007/978-3-319-11964-9_4.

[54] Anthony Fader, Stephen Soderland, and Oren Etzioni. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545. Association for Computational Linguistics, 2011.

[55] Christiane Fellbaum. *WordNet*. Wiley Online Library, 1998.

[56] Miriam Fernandez, Arno Scharl, Kalina Bontcheva, and Harith Alani. User profile modelling in online communities. In *Third International Workshop on Semantic Web Collaborative Spaces at the 13th International Semantic Web Conference*, page 15. SWCS, 2014.

[57] Óscar Ferrández, Christian Spurk, Milen Kouylekov, Iustin Dornescu, Sergio Ferrández, Matteo Negri, Rubén Izquierdo, David Tomás, Constantin Orasan, Guenter Neumann, Bernardo Magnini, and José Luis Vicedo González. The QALL-ME framework: A specifiable-domain multilingual question answering architecture. *J. Web Sem.*, 9(2):137–145, 2011. doi: 10.1016/j.websem.2011.01.002. URL `http://dx.doi.org/10.1016/j.websem.2011.01.002`.

[58] David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A Kalyanpur, Adam Lally, J William Murdock, Eric Nyberg, John Prager, et al. Building watson: An overview of the deepqa project. *AI magazine*, 31(3):59–79, 2010.

[59] Jenny Rose Finkel, Trond Grenager, and Christopher D. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA*, 2005. URL `http://acl.ldc.upenn.edu/P/P05/P05-1045.pdf`.

[60] Annika Flemming. Quality Characteristics of Linked Data Publishing Datasources. `http://sourceforge.net/apps/mediawiki/trdf/index.php?title=Quality_Criteria_for_Linked_Data_sources`, 2010. [Online; accessed 13-February-2014].

[61] Giorgos Flouris, Irini Fundulaki, Panagiotis Pediaditis, Yannis Theoharis, and Vassilis Christophides. *Coloring RDF triples to capture provenance*. Springer, 2009. doi: 10.1007/978-3-642-04930-9_13.

[62] Michel Gagnon, Caroline Barrière, and Eric Charton. Full syntactic parsing for enrichment of rdf dataset. In Anna Lisa Gentile, Ziqi Zhang, Claudia d'Amato, and Heiko Paulheim, editors, *LD4IE@ISWC*, volume 1057 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2013. URL `http://dblp.uni-trier.de/db/conf/semweb/ld4ie2013.html#GagnonBC13`.

[63] Alberto García-Durán, Antoine Bordes, and Nicolas Usunier. Effective blending of two and three-way interactions for modeling multi-relational data. In *Machine Learning and Knowledge Discovery in Databases*, pages 434–449. Springer, 2014.

[64] Daniel Gerber and A-C Ngonga Ngomo. Bootstrapping the linked data web. In *1st Workshop on Web Scale Knowledge Extraction@ ISWC*, volume 2011, 2011.

[65] Lise Getoor and Christopher P. Diehl. Link mining: a survey. *SIGKDD Explor. Newsl.*, 7(2):3–12, 2005. ISSN 1931-0145. doi: http://doi.acm.org/10.1145/1117454.1117456.

[66] Roxana Girju, Preslav Nakov, Vivi Nastase, Stan Szpakowicz, Peter Turney, and Deniz Yuret. Semeval-2007 task 04: Classification of semantic relations between nominals. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 13–18. Association for Computational Linguistics, 2007.

[67] Jennifer Golbeck. Combining provenance with trust in social networks for semantic web content filtering. In *Provenance and Annotation of Data*, pages 101–108. Springer, 2006. doi: 10.1007/11890850_12.

[68] Jennifer Golbeck. Trust on the world wide web: a survey. *Foundations and Trends in Web Science*, 1(2): 131–197, 2006. doi: 10.1561/1800000006.

[69] Jennifer Golbeck, Bijan Parsia, and James Hendler. *Trust networks on the semantic web*. Springer, 2003. doi: 10.1007/978-3-540-45217-1_18.

[70] Alex Graves. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850, 2013. URL `http://arxiv.org/abs/1308.0850`.

[71] Todd J Green, Grigoris Karvounarakis, and Val Tannen. Provenance semirings. In *Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 31–40. ACM, 2007. doi: 10.1145/1265530.1265535.

[72] W3C Owl Working Group et al. OWL 2 web ontology language document overview. 2009.

[73] Christophe Guéret, Paul Groth, Claus Stadler, and Jens Lehmann. Assessing linked data mappings using network measures. In *The Semantic Web: Research and Applications*, pages 87–102. Springer, 2012.

[74] Peter Haase, Michael Schmidt, and Andreas Schwarte. The information workbench as a self-service platform for linked data applications. In *COLD*. Citeseer, 2011.

[75] Sherzod Hakimov, Christina Unger, Sebastian Walter, and Philipp Cimiano. Applying semantic parsing to question answering over linked data: Addressing the lexical gap. In *Natural Language Processing and Information Systems*, pages 103–109. Springer, 2015.

[76] Dilek Hakkani-Tür, Gökhan Tür, Larry Heck, Asli Celikyilmaz, Ashley Fidler, Dustin Hillard, Rukmini Iyer, and Sarangarajan Parthasarathy. Employing web search query click logs for multi-domain spoken language understanding. In *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*, pages 419–424. IEEE, 2011.

[77] Harry Halpin and James Cheney. Dynamic provenance for sparql updates. In *The Semantic Web–ISWC 2014*, pages 425–440. Springer, 2014. doi: 10.1007/978-3-319-11964-9_27.

[78] Harry Halpin and James Cheney. Dynamic provenance for sparql updates using named graphs. In *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, pages 287–288. International World Wide Web Conferences Steering Committee, 2014. doi: 10.1145/2567948.2577357.

[79] Sanda Harabagiu, Dan Moldovan, and Joe Picone. Open-domain voice-activated question answering. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics, 2002.

[80] Steve Harris, Andy Seaborne, and Eric Prud'hommeaux. Sparql 1.1 query language. *W3C Recommendation*, 21, 2013.

[81] Andreas Harth, Jürgen Umbrich, Aidan Hogan, and Stefan Decker. YARS2: A federated repository for querying graph structured data from the Web. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4825 LNCS:211–224, 2007. ISSN 03029743. doi: 10.1007/978-3-540-76298-0_16.

[82] Shizhu He, Yuanzhe Zhang, Kang Liu, and Jun Zhao. Casia@ v2: A mln-based question answering system over linked data. *Proc. of QALD-4*, 2014.

[83] Marti A Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*, pages 539–545. Association for Computational Linguistics, 1992.

[84] Tom Heath and Enrico Motta. The hoonoh ontology for describing trust relationships in information seeking. *Personal Identification and Collaborations: Knowledge Mediation and Extraction (PICKME2008)*, 2008.

[85] Larry Heck and Dilek Hakkani-Tür. Exploiting the semantic web for unsupervised spoken language understanding. In *Spoken Language Technology Workshop (SLT), 2012 IEEE*, pages 228–233. IEEE, 2012.

[86] Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions*, pages 94–99. Association for Computational Linguistics, 2009.

[87] Edzard Höfig and Ina Schieferdecker. Hashing of rdf graphs and a solution to the blank node problem. In *10th International Workshop on Uncertainty Reasoning for the Semantic Web (URSW 2014)*, page 55, 2014.

[88] Ignacio Iacobacci, Mohammad Taher Pilehvar, and Roberto Navigli. SENSEMBED: Learning Sense Embeddings for Word and Relational Similarity. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, Beijing, China*, 2015.

[89] Charles Lee Isbell, Jr., Michael J. Kearns, Dave Kormann, Satinder P. Singh, and Peter Stone. Cobot in lambdamoo: A social statistics agent. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 36–41. AAAI Press, 2000. ISBN 0-262-51112-6. URL `http://dl.acm.org/citation.cfm?id=647288.721582`.

[90] Robert Isele, Anja Jentzsch, Chris Bizer, and Julius Volz. Silk - A Link Discovery Framework for the Web of Data, January 2011. URL `http://www4.wiwiss.fu-berlin.de/bizer/silk/`.

[91] Ali Ismayilov, Dimitris Kontokostas, Sören Auer, Jens Lehmann, and Sebastian Hellmann. Wikidata through the eyes of dbpedia. 2015.

[92] Ian Jacobi, Lalana Kagal, and Ankesh Khandelwal. Rule-based trust assessment on the semantic web. In *Rule-Based Reasoning, Programming, and Applications*, pages 227–241. Springer, 2011. doi: 10.1007/978-3-642-22546-8_18.

[93] Wittawat Jitkrittum, Choochart Haruechaiyasak, and Thanaruk Theeramunkong. QAST : Question Answering System for Thai Wikipedia. *Knowledge Creation Diffusion Utilization*, (August):11–14, 2009.

[94] Michael I Jordan. Serial order: A parallel distributed processing approach. *Advances in psychology*, 121: 471–495, 1997.

[95] Daniel Jurafsky and James H Martin. *Speech and language processing, 2nd edition*. Prentice Hall, 2008.

[96] Aditya Kalyanpur, Branimir K Boguraev, Siddharth Patwardhan, J William Murdock, Adam Lally, Chris Welty, John M Prager, Bonaventura Coppola, Achille Fokoue-Nkoutche, Lei Zhang, et al. Structured data and inference in deepqa. *IBM Journal of Research and Development*, 56(3.4):10–1, 2012.

[97] H. Kim, Kyungsun Kim, G.G. Lee, and Jungyun Seo. MAYA: A fast question-answering system based on a predictive answer indexer. *Proceedings of the workshop on Open-domain question answering-Volume 12*, page 8, 2001. URL `http://portal.acm.org/citation.cfm?id=1117856.1117858`.

[98] Paul Kingsbury and Martha Palmer. From treebank to propbank. In *LREC*. Citeseer, 2002.

[99] Dimitris Kontokostas, Patrick Westphal, Sören Auer, Sebastian Hellmann, Jens Lehmann, Roland Cornelissen, and Amrapali Zaveri. Test-driven evaluation of linked data quality. In *Proceedings of the 23rd international conference on World Wide Web*, pages 747–758. ACM, 2014.

[100] Andrew Koster, Jordi Sabater-Mir, and Marco Schorlemmer. Inductively generated trust alignments based on shared interactions. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*, pages 1571–1572. International Foundation for Autonomous Agents and Multiagent Systems, 2010. doi: 10.1145/1838206.1838486.

[101] Alon Lavie and Abhaya Agarwal. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 228–231, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL `http://dl.acm.org/citation.cfm?id=1626355.1626389`.

[102] Timothy Lebo, Satya Sahoo, Deborah McGuinness, Khalid Belhajjame, James Cheney, David Corsar, Daniel Garijo, Stian Soiland-Reyes, Stephan Zednik, and Jun Zhao. Prov-o: The prov ontology. *W3C Recommendation*, 30, 2013.

[103] Jens Lehmann, Daniel Gerber, Mohamed Morsey, and Axel-Cyrille Ngonga Ngomo. Defacto - deep fact validation. In *Proc. of the International Semantic Web Conference*, 2012. URL `http://jens-lehmann.org/files/2012/iswc_defacto.pdf`.

[104] Vanessa Lopez, Michele Pasin, and Enrico Motta. Aqualog: An ontology-portable question answering system for the semantic web. In *The Semantic Web: Research and Applications*, pages 546–562. Springer, 2005.

[105] Vanessa Lopez, Enrico Motta, Marta Sabou, and Miriam Fernandez. Poweraqua: A multi-ontology based question answering system–v1. *OpenKnowledge Deliverable D8. 4 Pp1*, 14, 2007.

[106] Vanessa Lopez, Victoria Uren, Marta Reka Sabou, and Enrico Motta. Cross ontology query answering on the semantic web: an initial evaluation. In *Proceedings of the fifth international conference on Knowledge capture*, pages 17–24. ACM, 2009.

[107] Vanessa Lopez, Miriam Fernández, Enrico Motta, and Nico Stieler. Poweraqua: Supporting users in querying and exploring the semantic web. *Semantic Web*, 3(3):249–265, 2011.

[108] Vanessa Lopez, Victoria Uren, Marta Sabou, and Enrico Motta. Is question answering fit for the semantic web? a survey. *Semantic Web*, 2(2):125–155, 2011.

[109] Markus Luczak-Roesch, Ramine Tinati, Elena Simperl, Max Van Kleek, Nigel Shadbolt, and Robert Simpson. Why won't aliens talk to us? content and community dynamics in online citizen science. In *Proceedings of the 8th International AAAI Conference on Weblogs and Social Media*, page 10. AIII, 2014.

[110] Yi Ma, Paul A Crook, Ruhi Sarikaya, and Eric Fosler-Lussier. Knowledge graph inference for spoken dialog systems.

[111] Robert M MacGregor and In-Young Ko. Representing contextualized data using semantic web tools. In *PSSS1 - Practical and Scalable Semantic Systems, Proceedings of the First International Workshop on Practical and Scalable Semantic Systems*, CEUR Workshop Proceedings, 2003.

[112] James Martens and Ilya Sutskever. Learning recurrent neural networks with hessian-free optimization. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pages 1033–1040, 2011.

[113] Edgard Marx, Ricardo Usbeck, Axel-Cyrille Ngonga Ngomo, Konrad Höffner, Jens Lehmann, and Sören Auer. Towards an open question answering architecture. In *Proceedings of the 10th Int. Conf. on Semantic Systems, SEMANTiCS 2014, Leipzig, Germany*, pages 57–60, 2014. doi: 10.1145/2660517.2660519. URL `http://doi.acm.org/10.1145/2660517.2660519`.

[114] Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings the 7th Int. Conf. on Semantic Systems, I-SEMANTICS 2011, Graz, Austria*, pages 1–8, 2011. doi: 10.1145/2063518.2063519. URL `http://doi.acm.org/10.1145/2063518.2063519`.

[115] Pablo N Mendes, Hannes Mühleisen, and Christian Bizer. Sieve: linked data quality assessment and fusion. In *Proceedings of the 2012 Joint EDBT/ICDT Workshops*, pages 116–123. ACM, 2012.

[116] Grégoire Mesnil, Xiaodong He, Li Deng, and Yoshua Bengio. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In *INTERSPEECH*, pages 3771–3775, 2013.

[117] Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tür, Xiaodong He, Larry Heck, Gökhan Tür, Dong Yu, et al. Using recurrent neural networks for slot filling in spoken language understanding. *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, 23(3): 530–539, 2015.

[118] Jan Michelfeit and Tomas Knap. Linked data fusion in odcleanstore. In Birte Glimm and David Huynh, editors, *International Semantic Web Conference (Posters & Demos)*, volume 914 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2012.

[119] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Cernockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Japan, September 26-30, 2010*, pages 1045–1048, 2010.

[120] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[121] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[122] Taniya Mishra and Srinivas Bangalore. Qme!: A speech-based question-answering system on mobile devices. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 55–63. Association for Computational Linguistics, 2010.

[123] Christian Morbidoni, Axel Polleres, Giovanni Tummarello, and Danh Le Phuoc. Semantic web pipes. *Rapport technique, DERI*, 71:108–112, 2007.

[124] Andrea Moro, Alessandro Raganato, and Roberto Navigli. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244, 2014.

[125] Alessandro Moschitti, Jennifer Chu-Carroll, Siddharth Patwardhan, James Fan, and Giuseppe Riccardi. Using syntactic and semantic structural kernels for classifying definition questions in jeopardy! In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 712–724, 2011. URL `http://www.aclweb.org/anthology/D11-1066`.

[126] Vinh Nguyen, Olivier Bodenreider, and Amit Sheth. Don't like rdf reification?: making statements about statements using singleton property. In *Proceedings of the 23rd international conference on World wide web*, pages 759–770. ACM, 2014. doi: 10.1145/2566486.2567973.

[127] Eyal Oren, Renaud Delbru, Michele Catasta, and Richard Cyganiak. Sindice . com : A Document-oriented Lookup Index for Open Linked Data.

[128] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL `http://dx.doi.org/10.3115/1073083.1073135`.

[129] Heiko Paulheim and Christian Bizer. Type inference on noisy rdf data. In Harith Alani, Lalana Kagal, Achille Fokoue, Paul T. Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha F. Noy, Chris Welty, and Krzysztof Janowicz, editors, *International Semantic Web Conference (1)*, volume 8218 of *Lecture Notes in Computer Science*, pages 510–525. Springer, 2013. ISBN 978-3-642-41334-6. URL `http://dblp.uni-trier.de/db/conf/semweb/iswc2013-1.html#PaulheimB13`.

[130] Heiko Paulheim and Christian Bizer. Improving the quality of linked data using statistical distributions. *Int. J. Semantic Web Inf. Syst.*, 10(2):63–86, 2014. URL `http://dblp.uni-trier.de/db/journals/ijswis/ijswis10.html#PaulheimB14`.

[131] Isaac Pinyol and Jordi Sabater-Mir. Computational trust and reputation models for open multi-agent systems: a review. *Artificial Intelligence Review*, 40(1):1–25, 2013. doi: 10.1007/s10462-011-9277-z.

[132] Patti Price. Evaluation of spoken language systems: The atis domain. In *Proceedings of the Third DARPA Speech and Natural Language Workshop*, pages 91–95. Morgan Kaufmann, 1990.

[133] E. Rahm and H.H. Do. Data cleaning: Problems and current approaches. *Data Engineering Bulletin*, 23: 3–13, 2000.

[134] TB Rajashekar and Bruce W Croft. Combining automatic and manual index representations in probabilistic retrieval. *Journal of the American society for information science*, 46(4):272–283, 1995.

[135] Adwait Ratnaparkhi. Trainable methods for surface natural language generation. In *Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference*, NAACL 2000, pages 194–201, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics. URL `http://dl.acm.org/citation.cfm?id=974305.974331`.

[136] Siva Reddy, Mirella Lapata, and Mark Steedman. Large-scale semantic parsing without question-answer pairs. *Transactions of the Association for Computational Linguistics*, 2:377–392, 2014.

[137] Ehud Reiter and Robert Dale. *Building Natural Language Generation Systems.* Cambridge University Press, New York, NY, USA, 2000. ISBN 0-521-62036-8.

[138] Matthew Richardson, Rakesh Agrawal, and Pedro Domingos. Trust management for the semantic web. In *The Semantic Web-ISWC 2003*, pages 351–368. Springer, 2003.

[139] Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M Marlin. Relation extraction with matrix factorization and universal schemas. 2013.

[140] Bryan Rink and Sanda Harabagiu. Utd: Classifying semantic relations by combining lexical and semantic resources. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 256–259. Association for Computational Linguistics, 2010.

[141] Alan Ritter, Colin Cherry, and William B. Dolan. Data-driven response generation in social media. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 583–593, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-937284-11-4. URL http://dl.acm.org/citation.cfm?id=2145432.2145500.

[142] Paolo Rosso, Lluis-F. Hurtado, Encarna Segarra, and Emilio Sanchis. On the voice-activated question answering. *IEEE Trans. on Systems, Man and Cybernetics-Part C: Applications and Reviews*, 42(1): 75–85, Jan 2012.

[143] Edna Ruckhaus, Oriana Baldizán, and María-Esther Vidal. Analyzing linked data quality with liquate. In *On the Move to Meaningful Internet Systems: OTM 2013 Workshops*, pages 629–638. Springer, 2013.

[144] Jordi Sabater and Carles Sierra. Review on computational trust and reputation models. *Artificial intelligence review*, 24(1):33–60, 2005. doi: 10.1007/s10462-004-0041-5.

[145] Satya S Sahoo, Olivier Bodenreider, Pascal Hitzler, Amit Sheth, and Krishnaprasad Thirunarayan. Provenance context entity (pace): Scalable provenance tracking for scientific rdf data. In *Scientific and Statistical Database Management*, pages 461–470. Springer, 2010. doi: 10.1007/978-3-642-13818-8_32.

[146] Emilio Sanchis, Davide Buscaldi, Sergio Grau, Lluis Hurtado, and David Griol. Spoken qa based on a passage retrieval engine. In *Spoken Language Technology Workshop, 2006. IEEE*, pages 62–65. IEEE, 2006.

[147] Craig Sayers and Alan H Karp. Computing the digest of an rdf graph. *Mobile and Media Systems Laboratory, HP Laboratories, Palo Alto, USA, Tech. Rep. HPL-2003-235*, 1, 2004.

[148] Craig Sayers and Alan H Karp. Rdf graph digest techniques and potential applications. *Mobile and Media Systems Laboratory, HP Laboratories Palo Alto, HPL-2004-95*, 2004.

[149] Simon Schenk, Renata Queiroz Dividino, and Steffen Staab. Reasoning with provenance, trust and all that other meta knowlege in owl. In *SWPM*, 2009.

[150] Nico Schlaefer, Jeongwoo Ko, Justin Betteridge, Manas A Pathak, Eric Nyberg, and Guido Sautter. Semantic extensions of the ephyra qa system for trec 2007. In *TREC*, 2007.

[151] Max Schmachtenberg, Christian Bizer, and Heiko Paulheim. State of the lod cloud. URL http://linkeddatacatalog.dws.informatik.uni-mannheim.de/state/.

[152] Michael Schmitz, Robert Bart, Stephen Soderland, Oren Etzioni, et al. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 523–534. Association for Computational Linguistics, 2012.

[153] Edward Schofield and Zhiping Zheng. A speech interface for open-domain question-answering. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 2*, pages 177–180. Association for Computational Linguistics, 2003.

[154] Guus Schreiber and Yves Raimond. Rdf 1.1 primer. *W3C Working Group Note*, 2014.

[155] Andreas Schultz, Andrea Matteini, Robert Isele, Pablo Mendes, Christian Bizer, and Christian Becker. Ldif-a framework for large-scale linked data integration. In *21st International World Wide Web Conference (WWW2012)*, 2012.

[156] Saeedeh Shekarpour, Edgard Marx, Axel-Cyrille Ngonga Ngomo, and Sören Auer. Sina: Semantic interpretation of user queries for question answering on interlinked data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 30:39–51, 2015.

[157] Wanita Sherchan, Surya Nepal, Jonathon Hunklinger, and Athman Bouguettaya. A trust ontology for semantic services. In *2010 IEEE International Conference on Services Computing*, pages 313–320. IEEE, 2010. doi: 10.1109/SCC.2010.42.

[158] Mohamed Ahmed Sherif, Axel-Cyrille Ngonga Ngomo, and Jens Lehmann. Automating RDF dataset transformation and enrichment. In *12th Extended Semantic Web Conference, Portoroz, Slovenia, 31st May - 4th June 2015*. Springer, 2015. URL `http://svn.aksw.org/papers/2015/ESWC_DEER/public.pdf`.

[159] J Sherwani, Dong Yu, Tim Paek, Mary Czerwinski, Yun-Cheng Ju, and Alex Acero. Voicepedia: towards speech-based access to unstructured information. In *INTERSPEECH*, volume 7, pages 146–149, 2007.

[160] Sameer Singh, Sebastian Riedel, Brian Martin, Jiaping Zheng, and Andrew McCallum. Joint inference of entities, relations, and coreference. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 1–6. ACM, 2013.

[161] Richard Socher and Christopher Manning. Deep learning for natural language processing. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Atlanta, Georgia, June 2013. Association for Computational Linguistics. URL `http://nlp.stanford.edu/courses/NAACL2013/`.

[162] Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 196–205, Denver, Colorado, May–June 2015. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/N15-1020`.

[163] Mark Steedman and Jason Baldridge. Combinatory categorial grammar. *Non-Transformational Syntax: Formal and Explicit Models of Grammar. Wiley-Blackwell*, 2011.

[164] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago: a core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, pages 697–706, 2007. doi: 10.1145/1242572.1242667. URL `http://doi.acm.org/10.1145/1242572.1242667`.

[165] Huan Sun, Hao Ma, Wen-tau Yih, Chen-Tse Tsai, Jingjing Liu, and Ming-Wei Chang. Open domain question answering via semantic enrichment. In *Proceedings of the 24th International Conference on World Wide Web, WWW 2015, Florence, Italy*, pages 1045–1055, 2015. doi: 10.1145/2736277.2741651. URL `http://doi.acm.org/10.1145/2736277.2741651`.

[166] Ilya Sutskever, James Martens, and Geoffrey Hinton. Generating text with recurrent neural networks. In Lise Getoor and Tobias Scheffer, editors, *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, ICML '11, pages 1017–1024, New York, NY, USA, June 2011. ACM. ISBN 978-1-4503-0619-5.

[167] Charles Sutton and Andrew McCallum. Piecewise training for structured prediction. *Machine learning*, 77(2-3):165–194, 2009.

[168] Charles Sutton and Andrew McCallum. Piecewise training for undirected models. *arXiv preprint arXiv:1207.1409*, 2012.

[169] Alberto Tonon, Michele Catasta, Gianluca Demartini, and Philippe Cudré-Mauroux. Fixing the domain and range of properties in linked data by context disambiguation. In Christian Bizer, Sören Auer, Tim Berners-Lee, and Tom Heath, editors, *LDOW@WWW*, volume 1409 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2015.

[170] Giovanni Tummarello, Richard Cyganiak, Michele Catasta, Szymon Danielczyk, Renaud Delbru, and Stefan Decker. Sig. ma: Live views on the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(4):355–364, 2010.

[171] Gökhan Tür, Dilek Hakkani-Tür, and Larry Heck. What is left to be understood in atis? In *Spoken Language Technology Workshop (SLT), 2010 IEEE*, pages 19–24. IEEE, 2010.

[172] Gökhan Tür, Dilek Z Hakkani-Tür, Dustin Hillard, and Asli Celikyilmaz. Towards unsupervised spoken language understanding: Exploiting query click logs for slot filling. In *INTERSPEECH*, pages 1293–1296, 2011.

[173] Gökhan Tür, Minwoo Jeong, Ye-Yi Wang, Dilek Hakkani-Tür, and Larry P Heck. Exploiting the semantic web for unsupervised natural language semantic parsing. In *INTERSPEECH*, 2012.

[174] Gökhan Tür, Ye-Yi Wang, and Dilek Hakkani-Tür. Techware: Spoken language understanding resources [best of the web]. *Signal Processing Magazine, IEEE*, 30(3):187–189, 2013.

[175] Peter D Turney. Similarity of semantic relations. *Computational Linguistics*, 32(3):379–416, 2006.

[176] Alexandr Uciteli, Christoph Goller, Patryk Burek, Sebastian Siemoleit, Breno Faria, Halyna Galanzina, Timo Weiland, Doreen Drechsler-Hake, Wolfram Bartussek, and Heinrich Herre. Search ontology, a new approach towards semantic search. In *Workshop on Future Search Engines at 44. Jahrestagung der Gesellschaft für Informatik, Informatik 2014, Stuttgart, Deutschland*, pages 667–672, 2014. URL `http://subs.emis.de/LNI/Proceedings/Proceedings232/article194.html`.

[177] Christina Unger and Philipp Cimiano. Pythia: Compositional meaning construction for ontology-based question answering on the semantic web. In *Natural Language Processing and Information Systems*, pages 153–160. Springer, 2011.

[178] Christina Unger, Lorenz Bühmann, Jens Lehmann, Axel-Cyrille Ngonga Ngomo, Daniel Gerber, and Philipp Cimiano. Template-based question answering over rdf data. *Proceedings of the 21st international conference on World Wide Web*, (639–648), 2012.

[179] Christina Unger, John McCrae, Sebastian Walter, Sara Winter, and Philipp Cimiano. A lemon lexicon for dbpedia. In *Proceedings of 1st International Workshop on NLP and DBpedia, co-located with the 12th International Semantic Web Conference (ISWC 2013), October 21-25, Sydney, Australia*, 2013.

[180] Christina Unger, André Freitas, and Philipp Cimiano. An introduction to question answering over linked data. In Manolis Koubarakis, Giorgos B. Stamou, Giorgos Stoilos, Ian Horrocks, Phokion G. Kolaitis, Georg Lausen, and Gerhard Weikum, editors, *Reasoning Web*, volume 8714 of *Lecture Notes in Computer Science*, pages 100–140. Springer, 2014. ISBN 978-3-319-10586-4.

[181] Ricardo Usbeck, Axel-Cyrille Ngonga Ngomo, Lorenz Bühmann, and Christina Unger. HAWK – Hybrid Question Answering using Linked Data.

[182] Ricardo Usbeck, Axel-Cyrille Ngonga Ngomo, Sören Auer, Daniel Gerber, and Andreas Both. Agdistis-agnostic disambiguation of named entities using linked open data. In *International Semantic Web Conference*, page 2, 2014.

[183] Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.

[184] Marilyn Walker, Rashmi Prasad, and A Stent. A trainable generator for recommendations in multimodal dialog. In *Proceedings of EUROSPEECH*, pages 1697–1701, 2003.

[185] Gill Wang, Zheng Mohanlal, and Zhao. Wisdom in the social crowd: an analysis of quora. In *Proceedings of the 22nd International Conference on World Wide Web*, page 11. WWW, 2013.

[186] Lu Wang, Larry Heck, and Dilek Hakkani-Tür. Leveraging semantic web search and browse sessions for multi-turn spoken dialog systems. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 4082–4086. IEEE, 2014.

[187] Ye-Yi Wang, Li Deng, and Alex Acero. Spoken language understanding. *Signal Processing Magazine, IEEE*, 22(5):16–31, 2005.

[188] Ye-Yi Wang, Li Deng, and Alex Acero. Semantic frame-based spoken language understanding. *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*, pages 41–91, 2011.

[189] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 1112–1119. Citeseer, 2014.

[190] Joseph Weizenbaum. Eliza—a computer program for the study of natural language communication between man and machine. *Commun. ACM*, 9(1):36–45, January 1966. ISSN 0001-0782. doi: 10.1145/365153.365168. URL `http://doi.acm.org/10.1145/365153.365168`.

[191] Edward WD Whittaker, Joanna Mrozinski, and Sadaoki Furui. Factoid question answering with web, mobile and speech interfaces. In *Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: companion volume: demonstrations*, pages 288–291. Association for Computational Linguistics, 2006.

[192] Kun Xu, Sheng Zhang, Yansong Feng, and Dongyan Zhao. Answering natural language questions via phrasal semantic parsing. In *Natural Language Processing and Chinese Computing*, pages 333–344. Springer, 2014.

[193] Ying Xu, Mi-Young Kim, Kevin Quinn, Randy Goebel, and Denilson Barbosa. Open information extraction with tree kernels. In *HLT-NAACL*, pages 868–877, 2013.

[194] Mohamed Yahya, Klaus Berberich, Shady Elbassuoni, and Gerhard Weikum. Robust question answering over the web of linked data. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 1107–1116. ACM, 2013.

[195] Zi Yang, Ying Li, James Cai, and Eric Nyberg. QUADS : Question Answering for Decision Support. *Proc. SIGIR 2014*, pages 375–384, 2014. doi: 10.1145/2600428.2609606.

[196] Mohamed Amir Yosef, Johannes Hoffart, Ilaria Bordino, Marc Spaniol, and Gerhard Weikum. Aida: An online tool for accurate disambiguation of named entities in text and tables. *Proceedings of the VLDB Endowment*, 4(12):1450–1453, 2011.

[197] Amrapali Zaveri, Dimitris Kontokostas, Mohamed A Sherif, Lorenz Bühmann, Mohamed Morsey, Sören Auer, and Jens Lehmann. User-driven quality evaluation of dbpedia. In *Proceedings of the 9th International Conference on Semantic Systems*, pages 97–104. ACM, 2013.

[198] Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. Quality assessment for linked data. *Semantic Web Journal*, preprint(preprint), 2015. URL `http://www.semantic-web-journal.net/content/quality-assessment-linked-data-survey`.

[199] Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. Kernel methods for relation extraction. *The Journal of Machine Learning Research*, 3:1083–1106, 2003.

[200] Luke S. Zettlemoyer and Michael Collins. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *UAI '05, Proceedings of the 21st Conference in Uncertainty in Artificial Intelligence, Edinburgh, Scotland, July 26-29, 2005*, pages 658–666, 2005. URL `https://dslpitt.org/uai/displayArticleDetails.jsp?mmnu=1&smnu=2&article_id=1209&proceeding_id=21`.

[201] Luke S Zettlemoyer and Michael Collins. Online learning of relaxed ccg grammars for parsing to logical form. In *EMNLP-CoNLL*, pages 678–687, 2007.

[202] Antoine Zimmermann, Nuno Lopes, Axel Polleres, and Umberto Straccia. A general framework for representing, reasoning and querying with annotated semantic web data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 11:72–95, 2012. doi: 10.1016/j.websem.2011.08.006.

[203] Lei Zou, Ruizhe Huang, Haixun Wang, Jeffer Xu Yu, Wenqiang He, and Dongyan Zhao. Natural language question answering over rdf: a graph data driven approach. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 313–324. ACM, 2014.