Knowledge-Based Systems 95 (2016) 12–34

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

The fuzzy ontology reasoner *fuzzyDL*

Fernando Bobillo[a,*], Umberto Straccia[b]

[a] *Department of Computer Science and Systems Engineering, University of Zaragoza, Zaragoza, Spain*
[b] *Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche (ISTI-CNR), Pisa, Italy*

ARTICLE INFO

Article history:
Received 13 May 2015
Revised 24 November 2015
Accepted 25 November 2015
Available online 4 December 2015

Keywords:
Fuzzy ontologies
Fuzzy description logics
Fuzzy ontology reasoner

ABSTRACT

Classical, two-valued, ontologies have been successfully applied to represent the knowledge in many domains. However, it has been pointed out that they are not suitable in domains where vague or imprecise pieces of information play an important role. To overcome this limitation, several extensions to classical ontologies based on fuzzy logic have been proposed. We believe, however, that the success of fuzzy ontologies strongly depends on the availability of effective reasoners able to deal with fuzzy ontologies.

In this paper we describe *fuzzyDL*, an expressive fuzzy ontology reasoner with some unique features. We discuss its possibilities for fuzzy ontology representation, the supported reasoning services, the different interfaces to interact with it, some implementation details, a comparison with other fuzzy ontology reasoners, and an overview of the main applications that have used it so far.

1. Introduction

In the last decade, *OWL 2 ontologies* [61], or simply *ontologies*, have become standard for knowledge representation. An ontology is an explicit and formal specification of the concepts, individuals and relationships that exist in some area of interest, created by defining axioms that describe the properties of these entities [74].

The theoretical underpinnings of ontologies are strongly based on *Description Logics* (DLs) [5]. DLs are a family of logics for representing structured knowledge that play a key role in the design of ontologies. Notably, DLs are essential in the design of OWL 2 (Web Ontology Language) [31,61], the current standard language to represent ontologies. As a matter of fact, OWL 2 is almost equivalent to the DL $\mathcal{SROIQ}(\mathbf{D})$.

Ontologies have been successfully used as part of expert and multiagent systems, as well as a core element in the Semantic Web, which proposes to extend the current web to give information a well-defined meaning [7]. Despite of their many advantages, ontologies also have some limitations. One of them is that their classical two-valued semantics cannot directly manage *imprecise* or vague pieces of knowledge, which are inherent to several real-world problems [71].

Fuzzy set theory and fuzzy logic [99] have proved to be suitable formalisms to handle these types of knowledge. Therefore, *fuzzy ontologies* emerge as useful in several applications, such as information retrieval [3,23,55,93], recommendation systems [25,51,64,97], image interpretation [32,33], the Semantic Web and the Internet [30,66,71], ambient intelligence [35,54], ontology merging [27,88], matchmaking [68], decision making [79], summarisation [50], robotics [36], and many others [6,37,45,49,52,58,59,65,69,73]. The interested reader is referred to Section 5 for details about some of these applications.

The main formalism behind fuzzy ontologies are fuzzy DLs, proposed as an extension to classical DLs with the aim of dealing with *fuzzy/vague/imprecise information*. Since the first work of Yen in 1991 [98], an important number of works can be found in the literature. For a good survey on *fuzzy DLs* we refer the reader to [56,81].

Fuzzy ontologies require the development of new languages and reasoning algorithms, together with the implementation of tools. Although there is not a standard for fuzzy ontology representation yet, some languages based on fuzzy DLs have been proposed [14], such as *Fuzzy OWL 2*. Due to the capability of ontologies to enable the automatic deduction of implicit knowledge, the success of fuzzy ontologies will strongly depend on the availability of effective reasoners.

fuzzyDL is an ontology reasoner supporting fuzzy logic reasoning. The objective of this paper is to present the latest version of the reasoner *fuzzyDL*, which offers various novel features with respect to previous versions. From a historical point of view, *fuzzyDL* can be considered as the first fuzzy DL reasoner. It is a popular and well-known tool that has been applied in several applications. In this paper we will present its possibilities for fuzzy ontology representation, the supported reasoning services, the different interfaces to interact with the system, some useful implementation details for ontology developers, a comparison of the main differences with other fuzzy ontology reasoners, and an overview of some applications that have successfully used *fuzzyDL*.

The rest of this paper is organised as follows. Section 2 describes an overview of the system, including its architecture and the

* Corresponding author. Tel.: +34 876555546.
E-mail addresses: fbobillo@unizar.es (F. Bobillo), straccia@isti.cnr.it (U. Straccia).

http://dx.doi.org/10.1016/j.knosys.2015.11.017
0950-7051/

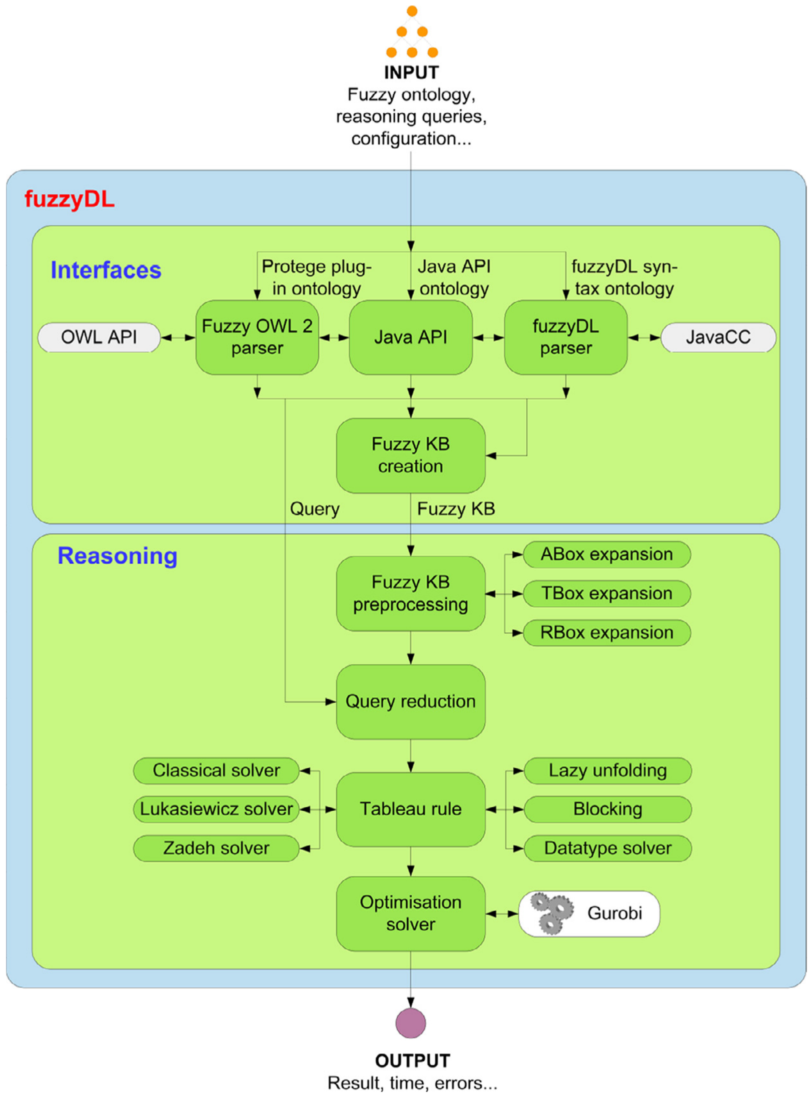

Fig. 1. Architecture of *fuzzyDL* reasoner.

different interfaces to interact with the tool. Section 3 enumerates the features and capabilities of the reasoner for fuzzy ontology representation and reasoning. Section 4 discusses some implementation details and summarises some optimisations. Next, Section 5 illustrates the usefulness of the system by revisiting some applications using it. Then, Section 6 compares *fuzzyDL* with other existing fuzzy DL reasoners, while Section 7 concludes and addresses some ideas for future work. Finally, we include two appendixes with some technical background (Appendix A) and a user manual with the syntax, semantics, and deeper details about the interaction with the tool (Appendix B).

2. System overview

fuzzyDL is a fuzzy ontology reasoner. It supports a very expressive language: a fuzzy extension of a fragment of the language OWL 2 extended with some unique features and capabilities of fuzzy logic. It supports fuzzy operators from several families of fuzzy logics and implements a unique algorithm based on a combination of tableau rules and an optimisation problem.

fuzzyDL is publicly available in its webpage.[1] The installation of *fuzzyDL* is easy, since the user can download the file, uncompress it and run the main jar file without any further installation. However, the user must also install Gurobi (a *Mixed Integer Linear Programming* solver, MILP)[2] and get a valid license. Currently, Gurobi offers free licenses for academic purposes and limited trials otherwise.

This section provides a quick overview of the system. We will firstly discuss the architecture of the system in Section 2.1, specifying the inputs and outputs. Then, Section 2.2 details the different interfaces to interact with *fuzzyDL*. A detailed description of what types of fuzzy ontologies can be represented and what reasoning types are supported can be found in Section 3.

2.1. Architecture

The architecture of the reasoner is depicted in Fig. 1 and describes its main components. There are two main parts: the upper part of the figure is dedicated to manage the user inputs, while the lower part orchestrates the reasoning procedures. These parts will be analysed in more detail in Sections 2.2 and 4, respectively.

The input of the reasoner is a fuzzy ontology and a set of user queries. As we will discuss in the next section, there are three different interfaces to provide this information: a Protégé plug-in to build fuzzy OWL 2 ontologies, its own syntax, and a Java API. These interfaces are handled by a Fuzzy OWL 2 parser, a *fuzzyDL* parser, and a Java

[1] http://straccia.info/software/fuzzyDL/fuzzyDL.html

[2] http://www.gurobi.com