

# SeDeLo: Using Semantics and Description Logics to Support Aided Clinical Diagnosis



Alejandro Rodríguez-González · Jose Emilio Labra-Gayo · Ricardo Colomo-Palacios · Miguel A. Mayer · Juan Miguel Gómez-Berbís · Angel García-Crespo

**Abstract** Automated medical diagnosis systems based on knowledge-oriented descriptions have gained momentum with the emergence of semantic descriptions. The objective of this paper is to propose a normalized design that solves some of the problems which have been detected by authors in previous tools. The authors bring together two different technologies to develop a new clinical decision support system: description logics aimed at developing inference systems to improve decision support for the prevention, treatment and management of illness and semantic technologies. Because of its new design, the system is capable of obtaining improved diagnostics compared with previous efforts. However, this evaluation

is more focused in the computational performance, giving as result that description logics is a good solution with small data sets. In this paper, we provide a well-structured ontology for automated diagnosis in the medical field and a three-fold formalization based on Description Logics with the use of Semantic Web technologies.

**Keywords** Description logics · Semantic technologies · Clinical diagnosis · Ontologies

## Introduction

Health care is one of the most complex and dynamic sectors in our society. In recent decades, medical institutions have increasingly adopted information technologies and computer-aided decision support systems. All organizations, including medical organizations, need to constantly improve their competitive advantage and respond faster to changing markets by reducing costs, improving quality and increasing productivity [1]. At the same time, the use of web technologies has also popularized the consultancy of health information by the general public. Thus, there is a need for computer-aided systems which could help experts and non-experts alike, in prescribing or understanding the given prescriptions or in diagnostic assistance.

Clinical diagnosis can be defined as the estimated identification of the disease by analyzing the signs and symptoms of a patient. The use of computational methods to enhance clinical diagnosis has a long tradition with a proliferation of decision-support expert systems. These systems can also be employed by non-experts who can learn from the accumulated expertise that is saved in those systems, and by medical students to help them prepare for their examinations.

---

A. Rodríguez González (✉) · R. Colomo Palacios ·  
J. M. Gómez Berbís · A. García Crespo  
Computer Science Department, Universidad Carlos III de Madrid,  
Av. Universidad 30, Leganés,  
28911 Madrid, Spain  
e mail: alejandro.rodriguez@uc3m.es

R. Colomo Palacios  
e mail: ricardo.colomo@uc3m.es

J. M. Gómez Berbís  
e mail: juanmiguel.gomez@uc3m.es

A. García Crespo  
e mail: angel.garcia@uc3m.es

J. E. Labra Gayo  
Computer Science Department, Universidad de Oviedo,  
Valdes Salas Street S/N,  
Oviedo 33007, Spain  
e mail: labra@uniovi.es

M. A. Mayer  
Research Programme on Biomedical Informatics (GRIB),  
IMIM Universitat Pompeu Fabra,  
Dr. Aiguader, 88,  
Barcelona 08003, Spain  
e mail: mmayer@comb.cat

Nevertheless, the implementation of effective clinical decision support is a challenging task and there is a lack of systematic studies such as those by [2, 3] which assess the efficiency and precision of those systems.

Semantic web can be seen as a vision for the future of the Web, where the unit of information is the data, instead of the content of a web page, as in the traditional Web. Around that vision of a web of data, the World Wide Web Consortium (W3C) has promoted the development of several technologies to describe resources by means of ontologies and rules. The main difference with more traditional approaches in knowledge engineering is the focus on a decentralized open world assumption and on the vast quantity of information, which are two features of the World Wide Web.

In this paper, we propose a new approach for developing a clinical diagnosis system by using semantic web technologies to infer diseases from symptoms, signs and laboratory tests formalized as logical descriptions. SeDeLo is relevant because in previous works (E.g. ADONIS [4]) it was necessary to use more than one single solution using description logics, which makes the system not efficient enough. We propose a solution based on single description logics that are accurate enough to solve the problems incurred by previous approaches.

The remainder of the paper is organized as follows. Section “**Background**” outlines relevant literature in the area. In Section “**System core**”, the architecture for the SeDeLo approach is presented along with the description of the ontology used and the description logics deployed. Conclusions and future work are discussed in Section “**Evaluation**”.

## **Background**

Decision support systems have been used for a variety of purposes, ranging from quality and safety to efficiency, and across a variety of clinical domains such as alert and reminders, screening, diagnosis and therapy [5]. Computerized Clinical Decision Support Systems (CDSS) are information systems designed to improve clinical decision making [3]. More specifically, CDSSs are “software that is designed to be a direct aid to clinical decision-making in which the characteristics of an individual patient are matched to a computerized clinical knowledge base, and patient-specific assessments or recommendations are then presented to the clinician and/or the patient for a decision” [6]. The work of many authors supports the evidence that decision support systems can be extremely effective in improving the quality and safety of healthcare (e.g. [3, 7–10]). The most famous expert systems were built as diagnosis assistants and therapy advisors in different medical areas (MYCIN [11], ONCOCIN [12], etc.) [13]. There are several taxonomies for

CDSSs in the literature (E.g. [5, 14, 15]) but in each of them Diagnosis Decision Support Systems are present.

Medical Diagnostic Decision Support Systems (MDSS), [16], a particular type of CDSS, have become a valuable aid in improving the accuracy of medical diagnosis (e.g. [17, 18]). Traditionally, MDSS have been based on expert systems [19], usually based on knowledge or rules, requiring significant efforts by professionals to model the domain in order to obtain accurate results. The main concept of the MDSS is an inductive engine that learns the decision characteristics of the diseases and can then be used to diagnose future patients with uncertain disease states [20].

On the other hand, Semantic Technologies have been pointed out as the future of the Web [21] and a new way to support knowledge [22, 23] in a wide range of domains [24], including medicine and biomedical domains [25–27]. Semantic Technologies, based on ontologies [28], provide a common framework that enables data integration, sharing and reusing from multiple sources. According to [29], semantic technologies can be exploited to reveal machine-readable latent relationships within specific diagnostic-related information in the medical field, where the homogeneity of terminology is particularly problematic. In this way, modern formal ontology facilitates the creation of knowledge-based systems such as those required for managing medical information [30]. A recent review of the use of ontologies in the medical domain can be found in [31]. In addition, the adaptation of clinical ontologies to real world scenarios is depicted in [32].

SeDeLo, following the path initiated by the authors in ADONIS approach, solves some of the problems detected, and achieves better results by using the capabilities of Ontology Web Language 2 (OWL 2) in clinical diagnosis decision support system.

## **System core**

Medical systems are extremely difficult to develop and evaluate. Notwithstanding, these systems should be considered as critical depending on the area where they will be used [33]. This is the reason why these systems need to be developed and evaluated with maximum efficiency.

Another important factor is the increasing use of Semantic Web technologies, which will help to create more accurate medical systems in this field which follow very simple but powerful logical rules integrating different sources of knowledge by means of shared ontologies like Open-Galen [34].

The solution proposed in this paper is an extension of the work realized in ADONIS. In this case, we are focusing only on the development of a new solution based also on Description Logics [35] where the algorithmic solution

developed in ADONIS can be discarded and where it is not necessary to use the two kinds of solutions for this problem. With this new solution, one single solution based on description logics is enough. These improvements are made through the definition of new description logics assertions to define the relations between findings and diseases. This new definition allows to, on a set of assertions, solve the diagnosis through multilevel inference (this concept is explained in the following sections).

In the semantic reasoner area, SEDELO implements Pellet [36]. A reasoner is a software tool that allows one to infer logical consequences from a set of asserted facts or axioms [37]. Some reasoners use different types of logics for reasoning including first order logic and description logic. Pellet started as a proof of concept system to help meet the W3C's implementation experience requirements for OWL. It has since become a practical and popular tool for working with OWL. Pellet has been the first reasoner to support all of OWL-DL, i.e. the Description Logic (DL). There are many other existing semantic reasoners like OWLIM [38], [39], FaCT++ [40], KAON [41], Racer [42] or more recently Hermit [43] to cite just some of them. In the literature there is a huge scientific production devoted to semantic reasoners benchmarking, being the work of Mishra & Kumar [44] one of the newest and more complete reviews.

In the next sections, we will present the problem with a brief description of the items that make up the development of our approach and the final solution obtained.

## Problem

**Note:** From now on we are going to make a simplification between signs and symptoms. Given that a symptom is considered as the subjective perception of a manifestation and a sign is a physical manifestation (and for hence, objective perception), we are going to treat both as "findings". With this simplification is easier to design the system given that you can ensure that signs and symptoms are always kind of findings, and if you want to code a sign or a symptom into the knowledge base is easier if you use the nomenclature of finding always.

The first approach we developed allowed us to make the correct inference in the diagnosis processes. However, the solution was not complete. The problem was that depending on how we formalized a given disease, the reasoner could not find a solution even though a solution could exist.

To understand better the formalization of the diseases, we show an example with two diseases and the findings associated with both of them (Fig. 1).

As is shown, Disease X has the findings "SymC, SymD and SymE" and Disease Y has the findings "Sym A, Sym B and DISX". DiseaseY is called by us "complex disease" because it is formed by at least one finding which is

represented by a disease. DISX in fact is "Disease X", so, Disease Y could be described as (Fig. 2):

The ability of the system to diagnosis DiseaseY introducing the findings of DiseaseX is which we have named as "*diagnosis through multilevel inference*". The multilevel situation is depicted in Fig. 3.

In Fig. 3 we can see the representation of the multilevel problem. The idea of our system is that depending of how can be described the assertions of the different diseases using description logics the system will be able to, diagnosis the disease of level 0 (DisY) introducing their own findings (S1, S2, S3, S4) (level 1) and the findings which belong to DisX (S5, S6, S7) which are at level 2.

The first approach that was also implemented in the ODDIN system [45] was able to make the next inferences (following scheme depicted in Fig. 2):

[SYM\_C, SYM\_D, SYM\_E] → Diseases X

[SYM\_A, SYM\_B, DIS\_X] → Diseases Y

This kind of inference was identified as "*normal diagnosis*". However, it was not able to make the next inference:

[SYM\_A, SYM\_B, SYM\_C, SYM\_D, SYM\_E] → Diseases Y

In this second scenario, note that the terms marked as bold are the findings of Disease X. The idea of this scenario is represent a case where you introduce the findings of a concrete disease instead the disease itself.

The reason why the original system, ODDIN, was not able to make this inference is based in the description logics and rule design. The original design of the description

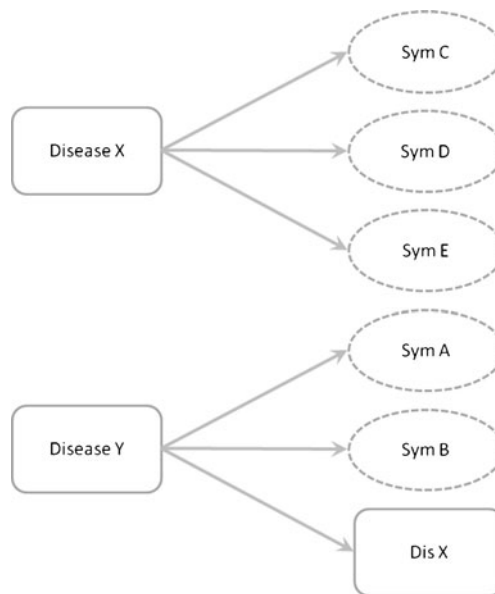


Fig. 1 Example of problem

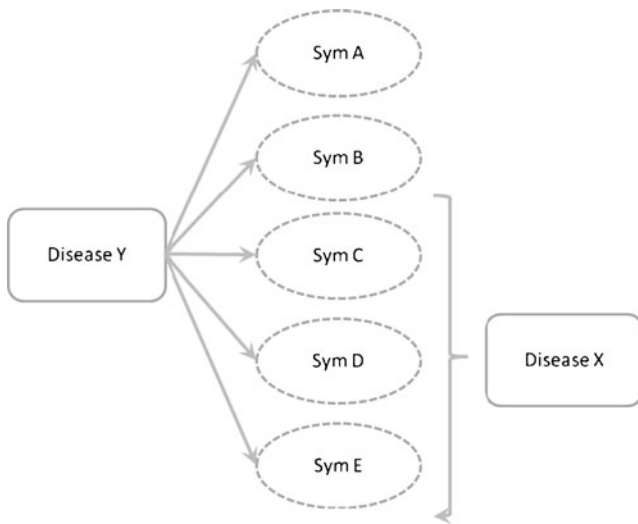


Fig. 2 Disease Y

logics was made to allow only inference of diseases which were designed in such a way that it was only possible to infer them if you introduce the exact terms that they were composed. In other words, in the presented case, the disease Y it only can be inferred if you introduce “SYM A”, “SYM B” and “DIS X”. If you try to interchange DIS X with their associated findings, the rules or description logics associated to DIS Y will not be able to infer DIS Y.

For this reason, we need to create some kind of approach that allows us two kind of inference:

1. The first one should be able of inference diseases when you give the original disease instead of their findings (first case presented).
2. The second one should be able of inference diseases when you give the findings associated to a concrete disease as findings input (second case presented).

Our current solution will be able to infer both cases using features of the new OWL 2 language [46].

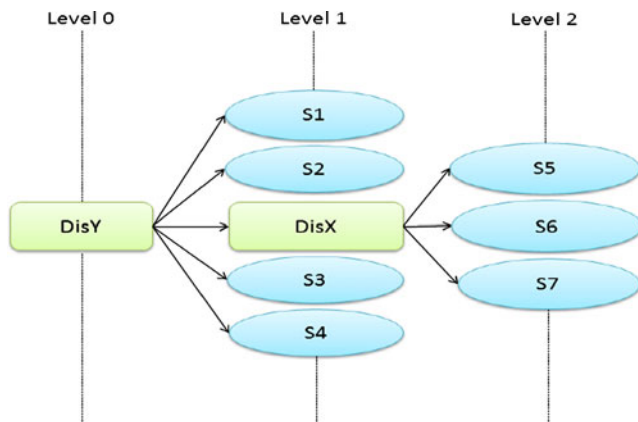


Fig. 3 Multilevel representation

## Ontology definition

In the next sections we will present a brief introduction of the elements of the previous and current solution.

The ontology that was used for the application of this case study is similar to the one that was developed for other works related to ontologies in medical applications for diagnostic data storage like MedFinder [47]. The main difference with the previous ontologies is the use of a more standardized taxonomy to identify the classes and instances. In this case we use SNOMED-CT codes to represents classes and instances instead of ICD-10 which was used in previous works. It should be mentioned that the fact of developing a new ontology is based in performance decisions. There are other several ontologies or taxonomy approaches which were designed to solve the representation problem like Disease Ontology or Symptoms Ontology. However, the main problem of these ontologies is the high amount of information that they represents. For our scenario we only need the codes (which represents the finding or the disease), and for this reason we discard the use of them, for this research.

## Ontology classes

There is a basic hierarchy that contains the elements of the diagnosis process:

- **Diseases:** The diseases class is structured following part of the multi-hierarchy structure provided by SNOMED-CT taxonomy. Given that the SNOMED-CT is multi-hierarchy and the difficulty of represents this multi-hierarchy structure we choose to create a single-hierarchy structure. The method used to transform this multi-hierarchy structure into a single-hierarchy structure was based on take as father of a concrete class, the class which seems to be the main father class. For this process we use a SNOMED browser to see the father of the different classes. In this class all the elements which are represented, in the SNOMED taxonomy are listed as “disorders”.
- **Findings:** The findings class contains all the findings available in the system. The structure is similar to the Disease class. There are some findings (e.g. Nausea and Vomiting) which are represented as “disorder” type in the SNOMED taxonomy. For this reason, these findings are also stored on Diseases class because in SNOMED-CT they are represented as disorders.
- **Diagnostic Tests (DiagTests):** The diagnostic tests class contains all the diagnostic tests available in the system. The structure is similar to Disease class.
- **Drugs:** This class contains all the drugs available in the system. The structure is similar to the Disease class.
- **Query:** Query class is used for the internal working of the system. Given that when we create a “clinic case” to

be asked to the system, we need to create an instance which contains the data associated to the clinic case (signs, symptoms, diagnostic tests, etc...); we use Query class as store class.

### Ontology properties

The ontology contains some object type properties in order to establish the relations between the instances of the classes in the ontology. In Table 1 we show the main relations. Underlined properties mean that they are functional so they cannot take more than one value. To get more information about the properties, see ODDIN or ADONIS papers.

The meaning of these properties is as follows:

- **isUsingDrug**: Specifies if the patient is taking some drug. It connects Query class with Drugs class.
- **isDiagTestOf**: Property to connect diagnostic tests with diseases. It is used to specify if a diagnostic test belongs to a concrete disease.
- **isFindingOf**: Property to connect findings with diseases to know if a concrete finding is characteristic of a concrete disease.
- **hasDiagTest**: Property to connect a disease with a diagnostic test.
- **hasFinding**: Property to connect a disease with a finding.

The ontology also contains some data type properties to establish values like name, codes, etc., but these properties are not important for the purposes of this study so no more mention shall be made. As an example, suppose there is a disease P with findings (symptoms for example in this case) A and E and diagnostic test LT1. The representation of this disease in the ontology will be as in the following:

- hasFinding
  - SYM A
  - SYM E
- hasDiagTest
  - LT 1

**Table 1** Object type properties of ontology

Name	Domain	Range
isUsingDrug	Query	Drugs
isDiagTestOf	DiagTests	Diseases
isFindingOf	Diseases Findings	Query Diseases/Drugs
hasDiagTest	Diseases/Query	DiagTests
hasFinding	Query/Diseases/Drugs	Diseases/Findings

### First approach

The original developed system, called ODDIN [45] had some faults in certain cases. The ADONIS approach [4] was developed in order to solve the problems generated in the ODDIN System. This first approach was divided in two parts:

#### Normal diagnosis

Definition for the DiseaseX:

$$\begin{aligned} &\exists \text{ hasSymptom } SYM\_C\_Class \cup \\ &\exists \text{ hasSymptom } SYM\_D\_Class \cup \\ &\exists \text{ hasSymptom } SYM\_E\_Class \end{aligned}$$

We specify that *DiseaseX* will be inferred when property *hasSymptom* (in ADONIS and ODDIN approach *hasSymptom* is the equivalent of *hasFinding*. We change the ontology structure in the actual approach to be more reliable with medical terms), which connects with the instances of SYM C, SYM D and SYM E classes matching with at least one of the given symptoms and signs to define a Disease to make the diagnosis. It is also necessary to specify the unique possible values of *DiseaseX* so that any other value will not be considered in the reasoning and, as a consequence, the disease should not be inferred. To that end, we must declare a closure axiom:

$$\forall \text{ hasSymptom}(SYM\_C\_Class \cup SYM\_D\_Class \cup SYM\_E\_Class)$$

The above line specifies that the only possible symptoms of DiseaseX are those that come from classes SYM C Class, SYM D Class or SYM E Class, and, as a consequence, we are restricting the inference to all the instances that do not belong to these classes.

Finally, we define the laboratory tests (again, in the current ontology structure we use hasDiagTest to relate diagnostic tests with diseases instead of make use only of laboratory tests). In this case, as DiseaseX does not have any laboratory test associated with it, we can define it as:

$$\forall \text{ hasLabTest owl : Nothing}$$

For DiseaseY, the definition will be the following:

$$\begin{aligned} &(\exists \text{ hasSymptom } SYM\_A\_Class \cup \\ &\exists \text{ hasSymptom } DIS\_B\_Class \cup \\ &\exists \text{ hasSymptom } DIS\_X\_Class \cap) \end{aligned}$$

$$\begin{aligned} &\forall \text{ hasSymptom} \\ &(SYM\_A\_Class \cup \\ &SYM\_B\_Class \cup \\ &DIS\_X\_Class) \cap \end{aligned}$$

$$\forall \text{ hasLabTest owl : Nothing}$$

As a final comment, it is important to mention that when a consult is formed, it is necessary to specify the number of symptoms and laboratory tests in the DL of the consult. For example, if we want to infer Disease X, we need to create a consult like:

$$\begin{aligned} &hasSymptom\ SYM\_C \wedge \\ &hasSymptom\ SYM\_D \wedge \\ &hasSymptom\ SYM\_E \wedge \\ &hasSymptom = 3 \wedge \\ &hasLabTest = 0 \end{aligned}$$

The reason for the need of this construct is the Open World Assumption (OWA) which is employed by the reasoner. In our definitions we are specifying that we have some concrete symptoms, but, as OWL makes use of OWA, it does not imply that there cannot be other symptoms or lab tests not mentioned in the description. To solve that problem, we employ the Closed World Assumption (CWA) to establish cardinalities for the instances that will make up the consult, as we have shown.

#### Diagnosis by levels

In order to formalize the diagnosis by levels, it was necessary to define new classes which collect the different symptoms (findings) associated with each disease. In our example case, we could define:

$$SymptomX = oneOf\{sym\_C, sym\_D, Sym\_E\}$$

Then, we could state

$$\begin{aligned} HasDiagnosisX &= \exists hasSymptom\ SymptomsX \cap \\ &\quad \forall hasSymptom\ SymptomsX \cap \\ &\quad hasSymptom = 3 \end{aligned}$$

This clause indicates that someone has diagnosis DiseaseX when it has all the symptoms from SyptomsX and only has the symptoms from *SymptomsX*. We also apply the closure declaring the cardinality of the set. In this way, the layered case can be covered by describing a new class

$$SymptomsY = oneOf\{sym\_A, sym\_B\} \cup SymptomsX$$

And the corresponding class

$$\begin{aligned} HasDiagnosisY &= \exists hasSymptom\ SymptomsY \cap \\ &\quad \forall hasSymptom\ SymptomsY \cap \\ &\quad hasSymptom = 5 \end{aligned}$$

Now, it is possible to infer from the knowledge that some patient has symptoms

$$\{sym\_A, sym\_B, sym\_C, sym\_D\ and\ sym\_E\}$$

That can be diagnosed with *DiseaseY*.

#### New approach by levels

The problem of the approach presented in the previous section is that it cannot solve a composed case like the following. Suppose we know that some patient has *DiseaseX* and the findings *sym A* and *sym B*, then the system should discard *DiseaseX* and infer *DiseaseY*.

In monotonic logic, like the description logics employed by OWL, it is not possible to discard entailments once they have been inferred. In this way, it has been necessary to separate the entailment of which findings belong to a given disease from the entailment of which disease can be diagnosed for a given patient. In general, for a given disease D, we define the class *HasSymptomsD* which collects all the findings that can be associated with that disease.

However, once we know that a given patient has all the findings of a given disease D, we could not yet diagnose D, because the patient could still have other findings related to another disease E, which should be inferred instead of D. It is necessary to help the reasoner by declaring that the patient has all the findings of D and only those findings. A simple solution is just to count the number of findings that a patient has and to check that number with the number of findings of a given disease. This process can be automated and it is the solution the current system employs. In the case of our running example, we would declare a class *HasFindingsX* as:

$$HasFindingsX \equiv hasFinding\ FindingsX = 3$$

The previous definition employs qualified cardinality restrictions, which were introduced in OWL 2 [46] and enable us to state that the set *HasFindingsX* is equivalent to those elements which have three findings from the set *FindingsX*. Given that the cardinality of that set is 3, a member of *HasFindingsX* will have all the findings of *DiseaseX*. Now, in the case of a patient with findings *A* and *B*, and the findings of *DiseaseX* the query would be as follows:

$$\begin{aligned} &hasFinding\ symA \wedge \\ &hasFinding\ symB \wedge \\ rdf &: type\ HasFindingsX \wedge \\ &hasFinding = 5 \wedge \\ &hasDiagTest = 0 \end{aligned}$$

Notice that it is also necessary to declare the total number of findings of the patient (in this case, 5) in order to restrict the

number of possible findings of that patient and to enable the reasoner to infer the proper disease. This calculation can be automatically made by the system. There is a more efficient approach based on concept products [48], which would not require counting the number of findings of each disease. Using description logics syntax, we could state:

$$\text{HasFindingsX} \times \text{FindingsX} \subseteq \text{hasFinding}$$

This means that for any patient  $p$  that belongs to  $\text{HasFindingsX}$  and for any findings that belongs to  $\text{FindingsX}$ ,  $p$  hasFinding. Concept products offer a more efficient solution but they are not yet supported by OWL 2 and have to be simulated using property chain axioms and top object properties.

## Evaluation

### Research design

In previous works which is based SeDeLo (ODDIN, and ADONIS) an evaluation about the diagnostic performance of the system was done. Given that ADONIS proposed a solution to the problem called diagnosis through multilevel inference, the diagnostic accuracy respect to ADONIS does not change, because it was able to solve at the end the same problem, but, with different approaches.

For this reason we consider that it is not necessary to make again another evaluation about diagnosis performance, given that the evaluation done in ADONIS and ODDIN covers all the possible diagnosis cases.

However, the development of SeDeLo evidenced the need to focus on other important evaluation area: computational performance. This kind of systems should be applied in some environments that can be considered “critical”. For this reason, we need to know if the system will be able not only to give a correct result (diagnosis accuracy, which was already evaluated). We need to know that the results will be given in a concrete time and with concrete resources.

Given that situation we have developed an evaluation process based on computational performance. The idea of this evaluation is to measure the time which the system needs to make a concrete diagnosis, and the memory consumption associated to this process.

The evaluation has been made measuring execution time of inference process.

### Sample

The sample was composed of two ontologies. First ontology (ontology 1) defines the assertions of the relations

of two hypothetical diseases and five findings (symptoms or signs). In this ontology, one of the diseases has as symptom the other one (one complex disease).

The second test is done using an ontology (ontology 2) which defines the assertions of the relations of five hypothetical diseases and 11 findings. In this ontology we have the definitions of the two diseases created in the first ontology and three diseases more. From these three additional diseases we have established that one of them also contains as finding another disease (two complex diseases).

Apart from the designed ontologies, we have created nine cases with the aim of obtaining the associated diagnosis to each case. From these nine ontologies, the ninth is designed to get as inference result, at least, one of the complex diseases. The cases which have been used in SeDeLo are different like the once used in ADONIS. The reason is that the evaluation aim of ADONIS was to measure the diagnosis performance, and for this reason, the cases were created specifically for this aim. However, in SeDeLo we are going to measure computational performance, and for this reason the cases which have been designed are different.

The main idea of this sample is to get the inference times when the elements (assertions and instances) of the ontologies are increased.

### Experiment

The experiment process consists in making an execution of the nine cases with the ontologies which have been presented in “Sample” section, and the measurement of two times: inference time and total time.

Inference time represents the inference process itself, and is excluded the process where the ontology is loaded. Total time is the sum of the inference time and ontology load time.

The experiment process has been executed ten times in order to try to avoid atypical values. The aim of the experiment is to know if there are significant differences between the executions of both ontologies in the nine cases and if the variables are related or independent between them.

### Results and discussion

#### Statistical analysis

In first place we are going to show the values which correspond to the execution of the nine cases over ontology 1. The values of the ten executions can be downloaded from Internet [49]. Table 2 shows the mean of the ten executions with standard deviation of both measured times.

**Table 2** Execution results of ontology 1. Mean values and T student results

Case	Total (ms)	Inference (ms)	T student result total (ms)	Significant differences
1	1,270.1±26.3	1,269.8±25.96	(t(18)=0.026, $p<.05$ )	No
2	1,72.4±2.12	172.2±2.25	(t(18)=0.205, $p<.05$ )	No
3	141.2±1.69	140.9±1.73	(t(18)=0.393, $p<.05$ )	No
4	156.1±2.88	155.7±2.98	(t(18)=0.305, $p<.05$ )	No
5	124.1±2.28	123.6±2.01	(t(18)=0.520, $p<.05$ )	No
6	110.9±3.38	110.3±3.5	(t(18)=0.390, $p<.05$ )	No
7	63.3±1.83	62.7±2.11	(t(18)=0.679, $p<.05$ )	No
8	78.2±0.92	71.4±19.49	(t(18)=1.102, $p<.05$ )	No
9	1,017.3±5.21	1,016±4.45	(t(18)=0.600, $p<.05$ )	No

The first test performed in order to find out whether there are significant differences between Total and the Inference time was a Student's *t*-test (comparison of two means) in ontology 1. The results of this test, which can be seen also in Table 2, showed that there are not significant differences between the Total time and Inference time for each case.

This circumstance, which can be easily inferred from the results, reveals that the total time is in the most of the cases equally to the inference time. This can be explained in an easy way because the ontology which contains the assertions of the diseases only contains a few description logics and this make that the API is able to load the ontology and prepare it (the prepare process consists in analyze the ontology to know if is consistent) in a few milliseconds, and in some occasions this process is even null.

The second step consists in showing and analyzing the values which correspond to the execution of the nine cases over ontology 2. The values of the ten executions can be downloaded from Internet [40]. Table 3 shows the mean of the ten executions with standard deviation of both measured times. Results of T-student and if there are significant differences between total time and inference time are also showed in Table 3.

The second test consists in the same that was performed in test 1. The only difference is that now, we are going to

compare the times of the second ontology. The results of this test showed again that there are not significant differences between the Total and Inference time in most of cases (only case 4 seems to has significant differences). This circumstance, again, can be easily inferred from the results, which reveals that the total time is in the most of the cases equally to the inference time. The explanation for this phenomenon is the same that occurs in the first test. In the most of the cases the process which consumes more time is the inference time. The unique difference that can be seen in this test compared with the other one is the result of case 4, which given the results obtained produce as result that there are differences between total time and inference time.

The third test consist in analyze both time variables (total and inference) using statistical methods which allows to see if exists a dependence or correlation between the variables. The method chosen in this case is based in measure the correlations between both variables, to know how related are. In this case, we have performed two subtest which consists in check if exists correlation between the variables analyzing cases of ontology 1 and analyzing cases of ontology 2.

The Spearman correlation showed an *R* value of 1.00,  $p<0.0001$ ,  $n=9$  in ontology 1 and an *R* value of 0.99  $p<0.0001$ ,  $n=9$  in ontology 2. These results indicate the high

**Table 3** Execution results of ontology 2. Mean values and T student results

Case C	Total (ms)	Inference (ms)	T student result total (ms)	Significant differences
1	9,850,300±4,295.99	9,847,800±3,457.68	(t(18)=1,434, $p<.05$ )	No
2	34,387,300±2,057.51	34,386,500±1,779.51	(t(18)=0,930, $p<.05$ )	No
3	3,138,800±1,316.56	3,137,800±1,229.27	(t(18)=1,756, $p<.05$ )	No
4	34,106,200±1,316.56	34,104,600±1,074.97	(t(18)=2,977, $p<.05$ )	Yes
5	12,483,300±1,494.43	12,482,800±1,229.27	(t(18)=0,817, $p<.05$ )	No
6	4,673,700±2,002.78	4,673,000±1,763.83	(t(18)=0,829, $p<.05$ )	No
7	7,301,900±1,286.68	7,301,200±1,398.41	(t(18)=1,165, $p<.05$ )	No
8	18,308,000±1,414.21	18,307,100±1,286.68	(t(18)=1,489, $p<.05$ )	No
9	178,542,100±2,469.82	156,505,400±3,405.88	(t(18)=1,489, $p<.05$ )	No



degree of correlation which exists between both variables in both ontologies. Something which is not surprising seeing the values.

As conclusion we can confirm that the main problem of reasoners which lead with ontologies is the inference process which consumes in most of cases the entirely time process. In the next section a qualitative analysis of the performance will be done based on the values obtained previously.

### *Performance analysis*

Once we have studied and analyzed the variables which measures the time of the execution process we are going to analyze from a more qualitative point of view the measurements and their impact.

As can be seen, the results obtained in ontology 1 (see Table 2) and ontology 2 (see Table 3) are quite different. Ontology 1 was described as a “light” ontology where the number of assertions is small (only contains two diseases and five findings, and for hence the number of assertions to describes the relations between the diseases and the findings is quite lower than in ontology 2). However, ontology 2 contains more assertions for the 11 diseases which contains.

The number of assertions is the main problem with the inference speed. While in ontology 1 the reasoner only has to deal with the assertions of two diseases, in ontology 2 the number of diseases, which is 11, makes that the assertions grows in comparison with the first ontology, and this is the reason of the inference and total times between both ontologies.

From this result a clear conclusion can be obtained: actual reasoners are not prepared to deal with the complexity (and number) of the assertions which need to be modeled to create a knowledge base which contains the descriptions and the relations of a big number of diseases.

### **Conclusions and future work**

This paper presents SeDeLo, which is a diagnosis system based on a well-structured ontology for automated diagnosis and a three-fold formalization based on Description Logics. The authors believe that SeDeLo could be used to improve the quality of healthcare services and clinical decision support by using semantic technologies for the prevention, treatment, and management of certain illnesses. Results show that, compared with previous works, SeDeLo is capable of solving certain problems like the ability of develop diagnosis decision support systems which will be able to diagnosis illnesses.

One important conclusion which this paper presents is that actual reasoners need to be improved in their internal algorithms to be able to provide a higher speed in the

inference process when they are dealing with complex description logics. Description logics can be used to formulate very complex domains with accuracy, making use of restrictions and assertions to improve the definition of the domain. However, work with description logics is a hard task from a computational point of view and this makes that their use in some domains like the once presented in this paper (medical diagnosis) could not be suitable.

Finally, SeDeLo is heading towards a four-pronged approach in terms of future work. The first one is the use of other reasoners to know if the scalability problems presented depends only on the reasoner or if the problem is in the underlying technology. The second one consists in try to improve the capacity of reasoning over ontologies with description logics using parallel and distributed computing. The third one is the development of the description rules presented in SeDeLo system as rule format (SWRL, Jena Rules, etc.) to improve the temporal efficiency of the system following the path described in [50]. The idea is to attempt to increase the performance of the system using a rule-based system instead of description logics where actual reasoners have efficiency problems as is shown in this paper. In the fourth approach, SeDeLo can be applied to a wider range of illnesses by expanding the set of illnesses included in the ontology.

**Acknowledgements** This work is supported by the Spanish Ministry of Industry, Tourism, and Commerce under the project TRAZAMED (IPT 090000 2010 007).

### **References**

1. O’Sullivan, D., and Dooley, L., Collaborative innovation for the management of information technology resources. *Int. J. Hum. Cap. Inf. Technol. Professionals*. 1(1):16 30, 2010.
2. Ammenwerth, E., Schnell Inderst, P., Machan, C., and Siebert, U., The effect of electronic prescribing on medication errors and adverse drug events: A systematic review. *J. Am. Med. Inform. Assoc.* 15(5):585 600, 2008.
3. Amit, X. G., Neill, K. J. A., Heather, M. M., Rosas Arellano, P., Devereaux, P. J., Beyene, J., Sam, J., and Haynes, R. B., Effects of Computerized Clinical Decision Support Systems on Practitioner Performance and Patient Outcomes: A Systematic Review. *J. Am. Med. Assoc.* 293(10):1223 1238, 2005.
4. García Crespo, A., Rodríguez González, A., Colomo Palacios, R., Labra Gayo, J. E., Gómez Berbis, J. M., and Alor Hernández, G., Automated Diagnosis through Ontologies and Logical Descriptions: the ADONIS approach. *International Journal of Decision Support System Technology (IJDSST)*. 3(1):21 39, 2011.
5. Wright, A., and Sittig, D. F., A four phase model of the evolution of clinical decision support architectures. *Int. J. Med. Inform.* 77 (10):641 649, 2008.
6. Sim, I., Gorman, P., Greenes, R., Haynes, R., Kaplan, B., Lehmann, H., and Tang, P. C., Clinical decision support systems

- for the practice of evidence based medicine. *J. Am. Med. Assoc.* 8 (6):527 534, 2001.
7. Kawamoto, K., Houlihan, C. A., Balas, E. A., and Lobach, D. F., Improving clinical practice using clinical decision support systems: A systematic review of trials to identify features critical to success. *Br. Med. J.* 330:765 768, 2005.
  8. Osheroﬀ, J. A., Teich, J. M., Middleton, B., Steen, E. B., Wright, A., and Detmer, D. E., A roadmap for National action on clinical decision support. *J. Am. Med. Inform. Assoc.* 14(2):141 145, 2007.
  9. Sittig, D. F., Wright, A., Osheroﬀ, J. A., Middleton, B., Teich, J. M., Ash, J. S., Campbell, E., and Bates, D. W., Grand challenges in clinical decision support. *J. Biomed. Inform.* 41(2):387 392, 2008.
  10. Kuperman, G. J., Bobb, A., Payne, T. H., Avery, A. J., Gandhi, T. K., Burns, G., Classen, D. C., and Bates, D. W., Medication related clinical decision support in computerized provider order entry systems: A review. *J. Am. Med. Inform. Assoc.* 14(1):29 40, 2007.
  11. Shortliffe, E. H., *Computer based medical consultations: MYCIN*: Elsevier Publishing, New York, NY, 1976
  12. Suwa, M., Scott, A. C., and Shortliffe, E. H., An approach to verifying completeness and consistency in a rule based expert system. *AI Mag.* 3(4):16 21, 1982.
  13. Gutiérrez Estrada, J. C., De Pedro, S. E., López Luque, R., Pulido Calvo, I., SEDPA, an expert system for disease diagnosis in eel rearing systems. *Aquac. Eng.* 33(2):110 125, 2005.
  14. Broverman, C. A., Standards for clinical decision support systems. *J. Healthc. Inf. Manag.* 13(2):23 31, 1999.
  15. Berlin, A., Sorani, M., and Sim, I., A taxonomic description of computer based clinical decision support systems. *J. Biomed. Inform.* 39(6):656 667, 2006.
  16. Mangiameli, P., West, D., and Rampal, R., Model selection for medical diagnosis decision support systems. *Decis. Support. Syst.* 36(3):247 259, 2004.
  17. Miller, R. A., Medical diagnostic decision support systems past, present, and future: a threaded bibliography and brief commentary. *J. Am. Med. Inform. Assoc.* 1(1):8 27, 1994.
  18. Sheng, O. R. L., Editorial: Decision support for healthcare in a new information age. *Decis. Support. Syst.* 30(2):101 103, 2004.
  19. Liao, S. H., Expert System methodologies and applications a decade review from 1995 to 2004. *Expert syst. Appl.* 28(1):93 103, 2005.
  20. Übeyli, E. D., Implementing automated diagnostic systems for breast cancer detection. *Expert Syst. Appl.* 33(4):1054 1062, 2007.
  21. Benjamins, V. R., Davies, J., Baeza Yates, R., Mika, P., Zaragoza, H., Gómez Pérez, J. M., Contreras, J., Domingue, J., and Fensel, D., Near term prospects for semantic technologies. *IEEE Intell. Syst.* 23(1):76 88, 2008.
  22. Vossen, G., Lytras, M., and Koudas, N., Editorial: Revisiting the (Machine) Semantic Web: The missing layers for the human Semantic Web. *IEEE Trans. Knowl. Data Eng.* 19(2):145 148, 2007.
  23. Fensel, D., and Munsen, M. A., The Semantic Web: A Brain for Humankind. *IEEE Intell. Syst.* 16(2):24 25, 2001.
  24. Lytras, M. D., and García, R., Semantic Web applications: a framework for industry and business exploitation What is needed for the adoption of the Semantic Web from the market and industry. *Int. J. Knowl. Learn.* 4(1):93 108, 2008.
  25. Gurupur, B. P., and Tanik, M. M., A system for building clinical research applications using Semantic Web based approach. *J. Med. Syst.* 2010. doi:10.1007/s10916 010 9445 8
  26. García Sánchez, F., Fernández Breis, J. T., Valencia García, R., Gómez, J. M., and Martínez Béjar, R., Combining Semantic Web Technologies with multi agent systems for integrated access to biological resources. *J. Biomed. Inform.* 41(5):848 859, 2008.
  27. Jalali, V., and Borujerdi, R. M. M., A unified architecture for biomedical search engines based on semantic Web Technologies. *J. Med. Syst.* 35(2):237 249, 2011.
  28. Fensel, D., *Ontologies: A silver bullet for knowledge management and electronic commerce*. Springer, Berlin, 2002.
  29. Fuentes Lorenzo, D., Morato, J., and Gómez Berbís, J. M., Knowledge management in biomedical libraries: A semantic web approach. *Inf. Syst. Front.* 11(4):471 480, 2009.
  30. Sicilia, J. J., Sicilia, M. A., Sánchez Alonso, S., García Barriocanal, E., and Pontikaki, M., Knowledge representation issues in ontology based clinical knowledge management systems. *Int. J. Technol. Manage.* 47(1 3):191 206, 2009.
  31. Schulz, S., Stenzhorn, H., Boeker, M., and Smith, B., Strengths and limitations of formal ontologies in the biomedical domain. *Electron. J. Commun. Inf. Innov. Health.* 3(1):31 45, 2009.
  32. Stenzhorn, H., Schulz, S., Boeker, M., and Smith, B., Adapting clinical ontologies in real world environments. *J. Univ. Comput. Sci.* 14(22):3767 3780, 2008.
  33. Graber, M., Diagnostic errors in medicine: A case of neglect. *Joint Comm. J. Qual. Patient Saf.* 31(2):106 113, 2005.
  34. Rector, A. L., Rogers, J. E., Zanstra, P. E., and Van der Haring, E., OpenGALEN: Open source medical terminology and tools. Proceedings of AMIA Symposium. 2003.
  35. Baader, F., Horrocks, I., and Sattler, U., Description Logics as Ontology Languages for the Semantic Web. In *Mechanizing Mathematical Reasoning*. SpringerLink. 2003.
  36. Sirin, E., Parsia, B., Cuenca Grau, B., Kalyanpur, A., Katz, Y., Pellet: A practical OWL DL reasoner. *Web. Semant. Sci. Serv. Agents World Wide. Web.* 5(2):51 53, 2007.
  37. Gómez Pérez, J. M., and Ruiz, C., Ontological Engineering and the Semantic Web. *Stud. Comput. Intell.* 311(2010):191 224, 2010.
  38. Kiryakov, A., Ognyanov, D., and Manov, D., OWLIM a Pragmatic Semantic Repository for OWL. Int. Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2005). 182 192, 2005.
  39. Bishop, B., Kiryakov, A., Ognyanoff, D., Peikov, I., Tashev, Z., and Velkov, R., OWLIM: A family of scalable semantic repositories. *Semantic Web.* 0:1 10, 2011.
  40. Tsarkov, D., and Horrocks, I., FaCT++ Description Logic Reasoner: System Description. In *Automated Reasoning Lect. Notes Comput. Sci.* 4130(2006):292 297, 2006.
  41. Motik, B., and Sattler, U., A comparison of reasoning techniques for querying large description logic ABoxes. Proceedings of the 13th international conference on logic for programming artificial intelligence and reasoning (LPAR 2006). 2006.
  42. Haarslev, V., and Moller, R., Consistence testing: the RACE experience. In: Dyckhoff R (ed) Proceedings, automated reasoning with analytic tableaux and related methods. Lecture Notes on Artificial Intelligence. 1847:57 61, 2000.
  43. Glimm, B., Horrocks, I., Motik, B., and Stoilos, G., HermiT: reasoning with large ontologies, <http://www.comlab.ox.ac.uk/projects/HermiT/index.html>, Computing Laboratory, Oxford University. 2009.
  44. Mishra, R. B., and Kumar, S., Semantic web reasoners and languages. *Artif. Intell. Rev.* 35(4):339 368, 2010.
  45. García Crespo, A., Rodríguez, A., Mencke, M., Gómez Berbís, J. M., and Colomo Palacios, R., ODDIN: Ontology driven differential diagnosis based on logical inference and probabilistic refinements. *Expert Syst. Appl.* 37(3):2621 2628, 2010.
  46. Motik, B., Patel schneider, P., and Parsia, B., OWL 2 Web Ontology Language, Structural specification and functional syntax. W3c Candidate recommendation. June 2009.

47. Rodriguez Gonzalez, A., Fernandez, J., Jiménez, E., Mencke, M., Radzinski, M., Gómez, J. M., Alor Hernandez, G., and Posada Gómez, R., MedFinder: Using Semantic Web, Web 2.0 and Geolocation methods to develop a Decision Support System to locate doctors. 5th International Conference on Web Information Systems and Technologies (WEBIST). 2009.
48. Rudolph, S., Krötzsch, M., and Hitzler, P., All Elephants are Bigger than All Mice. In Proceedings of the 21st International Workshop on Description Logics (DL 08). CEUR Workshop Proceedings 2008.
49. Rodriguez, A., SeDeLo Raw Data. Available online at: [http://nadir.uc3m.es/alejandro/pubs/sedelo/SeDeLo\\_Data.pdf](http://nadir.uc3m.es/alejandro/pubs/sedelo/SeDeLo_Data.pdf). Last accessed: March 29, 2011
50. Park, S., and Lee, J. K., Rule identification using ontology while acquiring rules from Web pages. *Int. J. Hum. Comput. Stud.* 65 (7):659-673, 2007.