Multi-Strategy Semantic Web Reasoning for Medical Knowledge Bases

William Van Woensel, Hossein Mohammadhassanzadeh, Samina Abidi, Raza Abidi

NICHE Research Group, Faculty of Computer Science,
Dalhousie University, Halifax, Canada
{william.van.woensel, hassanzadeh, samina.abidi, raza.abidi}@dal.ca

Abstract. Semantic Web technology offers excellent advantages for real-world medical knowledge bases, both on and off the Web. Based on ontologies from (bio)medical domains, OWL inferencing enhances knowledge bases with new facts, while deductive rules, written in semantic rule languages, supply additional inferences based on deterministic knowledge from e.g., Clinical Practice Guidelines (CPG). We argue that other mechanisms, representing weaker forms of inferencing, are also useful in dealing with incomplete healthcare knowledge. This includes inductive generalization, which leverages data similarities to induce new rules, and analogical reasoning, which relies on plausible domain knowledge. To cope with their shortcomings, we propose integrating such weak inferencing with a single, explanation-based generalization, allowing us to leverage their complementary strengths as well as apply a tutor-based paradigm for verification. In this integrated approach, justifications are generated explaining the potential correctness of queries, where missing medical knowledge is compensated by injecting plausible inferences. Based on their expertise, healthcare experts may then confirm particular justifications, materializing them in the knowledge base. Inversely, we argue that by leveraging OWL ontological knowledge, weak inferencing methods benefit from Semantic Web technology as well.

Keywords: medical knowledge bases, Semantic Web reasoning, explanation-based generalization, inductive generalization, analogical reasoning

1 Introduction

Clinical Decision Support Systems (CDSS) are knowledge-based systems, designed to assist health professionals in making 'evidence-informed' clinical decisions, based on the best evidence and approved clinical protocols [1] in concert with a patient's medical records [2]. Given a clinical case, a CDSS reasons over the available knowledge to generate evidence-based recommendations for diagnosis/therapy, alerts, risk assessments and order sets. For the knowledge base of a CDSS, the source of knowledge are published, evidence-based clinical guidelines and the expert's clinical experiences [3]. Developing a complete and consistent clinical knowledge base therefore is a major undertaking, as it requires (a) the computerization of the clinical guidelines in terms of decision rules, (b) the validation of the abstracted rules to ensure relevance, consistency and conciseness, and (c) implementing the validated rules in terms of an executable knowledge representation formalism. To realize medical knowledge bases, the Semantic Web framework offers a formal, logic-based framework for modeling the knowledge from clinical guidelines—i.e., via expressive biomedical OWL ontologies, allowing clinical facts to be inferred based on utilized ontological constructs; and using deductive rule languages, such as RuleML, SPIN or SWRL, which allow applying deterministic knowledge to infer clinical conclusions. From a knowledge engineering perspective, we posit that semantic ontologies are particularly useful for representing and reasoning over medical knowledge bases. This observation is reflected by work in CDSS likewise applying Semantic Web technology [4–6].

Additionally, we argue that useful mechanisms for enriching semantic medical knowledge bases are not limited to a priori, deterministic knowledge. To further extend the knowledge closure, other mechanisms can be considered as well, for instance based on data similarities and plausible knowledge (i.e., knowledge that will likely not hold in all possible situations). Inductive generalization hypothesizes that commonalities between data entities likely account for a particular shared feature; this knowledge can then be formalized in terms of rules [7]. Analogical reasoning is guided by plausible domain knowledge, implying if two data entities are similar in one specific aspect, they are likely similar in another as well [8]. These two approaches are complementary: induction is a data-intensive approach, requiring a large number of positive examples; while analogy relies on less data, but requires the presence of (plausible) knowledge. In any case, these kinds of reasoning (also including e.g., abductive reasoning [9]) are considered weaker forms of inferencing, as they do not guarantee the truth of inferred knowledge [7]. This is especially problematic in medical settings, where drawing correct, evidence-based conclusions from the available data is paramount. An effective way to cope with this tentative nature is by integrating multiple weak methods with deterministic, explanation-based generalization, allowing their complementary strengths to be leveraged [7, 10, 11]. Based on facts and deductive rules, this generalization constructs justifications that explain why a given statement may be correct [11]. In an integrated approach, weak inferencing is applied when full deterministic knowledge is lacking, yielding *plausible* justifications that are supplemented with weak inferences [10]. Importantly, such integrated justifications allow medical domain experts to carefully weigh the tentative data and mechanisms used in the process. The expert may then choose to confirm and fine-tune a particular justification, to be materialized in the knowledge base for future re-use.

We argue that a multi-strategy reasoning approach is useful for enhancing access to medical semantic knowledge bases. In particular, we present a Semantic Web medical expert system that generates & presents plausible justifications to medical experts, using deterministic and plausible inferencing methods. This expert can further confirm and fine-tune particular justifications to add them to the knowledge base. In contrast to fully automated mechanisms, our approach combines machine inferencing capabilities with the tacit, medical experience of human experts, resulting in fully human-verified knowledge bases. Inversely, we argue that the accuracy of weak inferencing methods can be improved by leveraging Semantic Web technology, thus enhancing their utility in the knowledge engineering cycle. In particular, rich ontological knowledge can be leveraged to improve data similarity checking, as well as increase the expressivity of plausible knowledge. Our semantic expert system accepts RDF, OWL ontologies and deductive semantic rules (SPIN rules) as input, and generates plausible justifications for presentation, verification and fine-tuning by the domain expert. Confirmed justifications are transformed into RDF triples and added to the knowledge base.

In Section 2, we elaborate on the reasoning layer of our expert system, discussing explanation-based generalization, inductive generalization and analogical reasoning. Section 3 details the UI providing query access to the domain expert. In Section 4, we summarize our expert system implementation. Section 5 presents related work, and Section 6 finally ends with conclusions and future work.

2 Reasoning Layer

A logical first step to supplementing incomplete Semantic Web knowledge bases is to apply ontology inferencing; i.e., leveraging domain-specific, ontological knowledge to infer new information. As a second step, our integrated approach taps an additional source of deterministic knowledge, namely deductive Semantic Web rules. At its core, our approach applies explanation-based generalization, which constructs justifications explaining why a given query instance is correct. This generalization is implemented via backward chaining, which recursively finds rules proving the query statement. In this process, premises of found rules may be proven by knowledge base facts or again by other deductive rules; either ending when all involved premises have been proven, or when some premises were found to be unsatisfiable.

This is illustrated in Figure 2, which visualizes the initial query justification as a tree structure. We note that for clarity, figures and rules use predicate logic notation (namespaces are omitted), and type restrictions on arguments are abbreviated as nested "(type X)" statements. In the tree, the root represents the query instance; leafs stand for facts in the knowledge base (solid line) or missing premises (dashed line); and intermediate nodes represent conjunctive rule premises, connected via directed edges to the rule conclusion. Grey-shaded leafs are facts inferred via ontology reasoning.

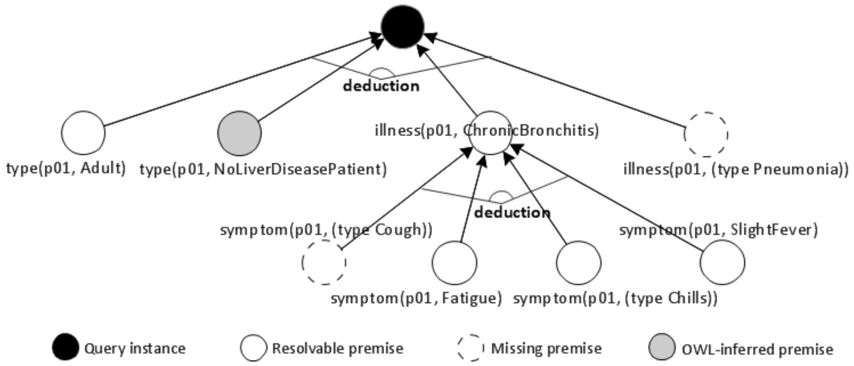

Fig. 1. Initial justification tree.

To explain our multi-strategy reasoning approach, we present a medical use case, where the system is asked to explain whether patient *p01* should be recommended *Azithromycin*, an antibiotic typically used to treat bacterial infections. Below, we show the deductive rules used to build the initial justification tree:

recommend(P, Azithromycin) : − type(P, Adult), type(P, NoLiverDiseasePatient),
illness(P, ChronicBronchitis), illness(P, (type Pneumonia)) .

illness(P, ChronicBronchitis) : − symptom(P, (type Cough)), symptom(P, Fatigue),
symptom(P, (type Chills)), symptom(P, SlightFever) .

Fig. 2. Deductive rules used in explanation-based generalization.

We note that the lack of liver disease history was inferred via an OWL property cardinality restriction:

```
cds:NoLiverDiseasePatient rdfs:subClassOf [
    a owl:Restriction ;
    owl:maxQualifiedCardinality "0" ;
    owl:onProperty cds:illness ;
    owl:onClass cds:LiverDisease  ] .
```

Fig. 3. OWL constraint cds:NoLiverDiseasePatient.

In the knowledge base, each current and past patient illness is recorded using the *illness* property. In case patients are not linked via the *illness* property to any instance of *LiverDisease*, this constraint implies that they are member of the *cds:noLiverDiseasePatient* class extension.

Currently, the cough symptom and pneumonia diagnosis are missing for *p01*, leading to an incomplete justification. To prove the correctness of the missing premises, inductive generalization and analogy-based reasoning are applied, resulting in a plausible justification tree that incorporates multi-strategy inferencing results. We elaborate on these two methods in the subsections below.

2.1 Analogy-based reasoning

Analogical inferencing supplements an incomplete knowledge base by implying that two entities who are similar in a particular aspect, are likely similar in another specific aspect as well. As such, knowledge can be transferred from a well-known entity S to a similar, lesser known entity T [8, 12, 13]. This kind of reasoning is guided by plausible rules of the form [7]:

$$Q(X,Y) :\sim P(X,Z)$$

Fig. 4. General structure of a plausible rule.

This rule states that "Q is plausibly determined by P", meaning if two entities S and T are both characterized by the same feature P, then they will likely also share feature Q. Below, we show a relevant plausible rule from our medical use case:

illness(P, (type Pneumonia)) :~ type(P, ReducedImmunityPatient), symptom(P, (type ShortnessOfBreath)), symptom(P, (type Fever)), symptom(P, ShakingChills)

Fig. 5. Example plausible rule.

This rule indicates that two data entities sharing particular characteristics and symptoms, namely a reduced immune system, a type of shortness of breath, a type of fever and shaking chills, will likely share the same kind of pneumonia. This example shows how ontological knowledge may be leveraged to increase the expressivity of plausible rules. In particular, by allowing to specify rules at arbitrary levels of granularity: any kind of fever, shortness of breath or reduced immune system suffices as knowledge-transfer condition, while the patient should exhibit "shaking chills" in particular. We note that the rule only specifies which kinds of features will collectively account for a type of pneumonia; the particulars are left up to the well-known entity. Below, we show RDF statements for patient *p07* and *p01*:

cds:p07 *rdf:type cds:OrganTransplantPatient ; cds:symptom cds:ShortBreathOnExertion cds:symptom cds:MediumFever ; cds:symptom cds:ShakingChills ;*
cds:illness cds:ViralPneumonia ; …

cds:p01 *rdf:type cds:HIVP\atient ; cds:symptom cds:ShortBreathOnExertion cds:symptom cds:MediumFever ; cds:symptom cds:ShakingChills ; …*

Fig. 6. RDF code describing entities *p01* and *p07*.

By sharing the same kind of shortness of breath (i.e., ShortBreathOnExertion), fever (i.e., MediumFever) and "shaking chills", *p01* shares almost all the required characteristics with *p07*. By further leveraging ontological knowledge, the system can cope with inexact matches as well; in particular, the system is able to infer that *HIVPatient* and *OrganTransplantPatient* are both subtypes of *ReducedImmunityPatient*. Based on the ontology hierarchy, our system can calculate and present accurate concept similarities to the medical expert. In Section 2.2, we elaborate on establishing such semantic similarities. Regarding our example, since patient p01 adheres to the knowledge-transfer condition (in italic), it can be plausibly inferred that p01 has viral pneumonia (in bold).

Table 1 shows the pseudocode for analogical reasoning in our expert system. The analogical reasoning process is driven by failed justification premises. In this case, the process starts by looking for plausible rules that may resolve a missing premise (line 1). The knowledge base is then searched for facts that *unify* the rule; i.e., match the rule condition and consequent (line 2). For each found fact, the algorithm checks whether its instantiated knowledge-transfer condition matches the original entity (line 3). If so, its consequent is added to the entity, at least in the system's working memory (line 4). If the expert confirms the overall justification, the working memory will be materialized in the knowledge base.

1. given *failed premise*, **query** KB for *plausible rules* that infer the missing property
   $\boldsymbol{example\ query}$: $illness(p01, (type\ Pneumonia))$
   $\rightarrow\ find\_plausible\_rules(illness(X, (type\ Pneumonia)), KB) = \{\ rule_i\ \}$
2. for each *plausible rule*, find KB facts that **unify** the rule
   $unifications = \{\ \forall\ rule_i : unif_i = unify(rule_i, KB)\ \}$
3. for each *unification*, check whether its **condition matches** the *input entity*
   $results = \{\ \forall\ unif_i : result_i = matches(condition(unif_i), p01)\ \}$

4. **add the consequent** of a positive match to the *input entity*
$p01 \leftarrow consequent(unif_i)$

Table 1. Analogical reasoning algorithm

Figure 8 illustrates part of our justification tree extended by analogical reasoning.

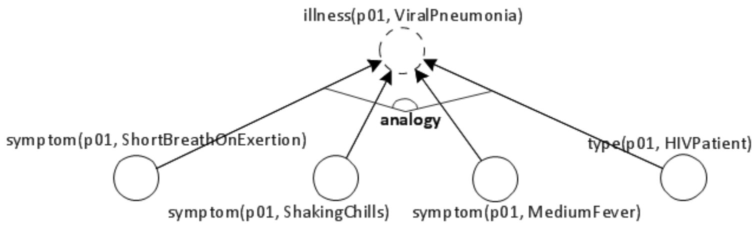

Fig. 7. Part of justification tree extended by analogical reasoning.

2.2 Inductive generalization

Inductive generalization generates new deterministic knowledge, based on similarities between facts in the knowledge base. Given a particular feature, the knowledge base is searched for a set of entities sharing the feature (called "positive coverage"). Other similarities between these entities are hereby hypothesized as accounting for the feature [7]. Applying inductive reasoning gives rise to the following type of rule:

$$feature(X, value) \colon\!- similarity_{S_i}(X)$$

Fig. 8. General structure of an induced rule

Where *feature(X, value)* represents the input feature, and *similarity*$_{Si}$ represents similarities between the found entities S_i. By applying this rule, the feature value is thus inferred for any entity X, which exhibits the characteristics assumed to account for the feature. We note that in this process, the number of matching entities (positive coverage) strengthens the validity of the hypothesis. Below, we discuss different ways of determining entity similarity.

A good basis for finding commonalities is to take the conjunction between entity properties, i.e., all shared properties [7]. As mentioned before (see Section 3.1), conceptual similarity between entity types can be considered as well. Calculating semantic concept similarity has been studied extensively in the literature (e.g., [14, 15]). Typically, these works determine concept similarity by calculating their conceptual distance, indicated by the distance to their closest common subsuming concept. It is further argued that two "specific" concepts, lower in the concept hierarchy, will be more similar than two "abstract" concepts [15]. We rely on the well-known measure suggested by Wu and Palmer [14], which considers both conceptual distance and concept specificity:

$$ConSim(C_1, C_2) = \frac{2 \times N_3}{N_1 + N_2 + 2 \times N_3}$$

Fig. 9. Conceptual similarity calculation

Where C3 is the closest common parent of C1 and C2; N1 is the number of nodes on the path from C1 to C3, and N2 is the number of nodes between C2 and C3 (indicating conceptual similarity); and N3 is the number of nodes on the path from C3 to the root (indicating conceptual specificity). In our approach, the medical expert will use this information to confirm, fine-tune or reject an inference (see Section 4).

The following RDF code shows two similar entities *p11* and *p15*, which share a number of characteristics as well as a particular feature currently missing from our justification tree (see Figure 2):

cds:p11 *cds:symptom cds:SlightFever ; cds:symptom cds:ChestDiscomfort cds:symptom cds:MucusProduction ;* **cds:symptom cds:CoughWithMucus** ; …

cds:p15 *cds:symptom cds:SlightFever ; cds:symptom cds:ChestDiscomfort cds:symptom cds:MucusProduction ;* **cds:symptom cds:CoughWithMucus** ; …

Fig. 10. RDF code describing similar entities *p11* and *p15*.

In this case, the following deductive rule could be inferred (positive coverage of 2):

symptom(P, CoughWithMucus) : − symptom(P, SlightFever),
symptom(P, ChestDiscomfort), symptom(P, MucusProduction)

Fig. 11. Rule generalized via inductive reasoning.

We note that, depending on the facts in the knowledge base, different sets of shared characteristics may yield the same missing feature, meaning multiple deductive rules could be inferred. Table 2 shows the pseudocode for inductive reasoning in our system. As before (see Table 1), the process is triggered by a failed premise in the justification tree. Firstly, the knowledge base is queried with (a generalized version of) the failed premise, to retrieve all entities with the missing property value (line 1). Subsequently, similarity is determined between matching entities and the query entity (line 2) using the *similarity* function (see Figure 9). The next step is to aggregate equivalent similarities (i.e., involving the same property values), thus collecting the necessary information (e.g., positive coverage) for each potential deductive rule (line 3). Afterwards, deductive rules are generated based on these aggregated similarities (line 4). These will be presented to the domain expert (see Section 4), who can possibly choose one of the rules to be added to the knowledge base.

1. given *failed premise*, **query** KB for *matching facts*
   $\boldsymbol{example\ query}: symptom\big(p01, (type\ Cough)\big)$
   $\rightarrow find\_matching\_facts\big(symptom\big(X, (type\ Cough)\big), KB\big) = \{\ result_i\ \}$
2. using *similarity* function, **determine similarity** between *results* and *initial entity*
   $similarities = \{\ \forall\ result_i: sim_i = similarity(result_i, p01)\ \}$
3. **aggregate** *equivalent similarities*
   $aggregated\ = \{\ \forall\ sim_i, sim_j\ (i \neq j): sim_i = sim_j \rightarrow\ aggr\_sim\big(\ sim_i, sim_j\big)\ \}$
4. **generate** *deductive rules* based on *aggregated similarities*
   $rules = \{\ \forall\ aggr\_sim_i: feature(X, value): -aggr\_sim_i(X)\ \}$

Table 2. Inductive generalization algorithm

In Figure 12, we show part of the justification tree extended by induction.

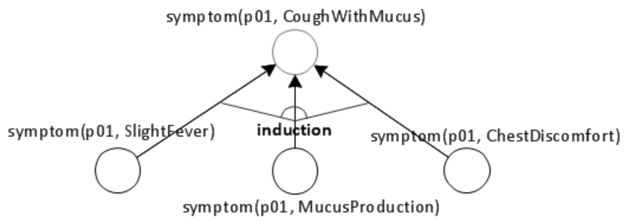

Fig. 12. Part of justification tree extended by inductive generalization.

3 User Interface Layer

In our Semantic Web expert system, healthcare experts query the knowledge base. In Section 2, we elaborated on how multiple reasoning mechanisms cooperate in creating plausible query justifications. In this section, we illustrate how our system presents these justifications, and their plausible extensions, to the medical expert.

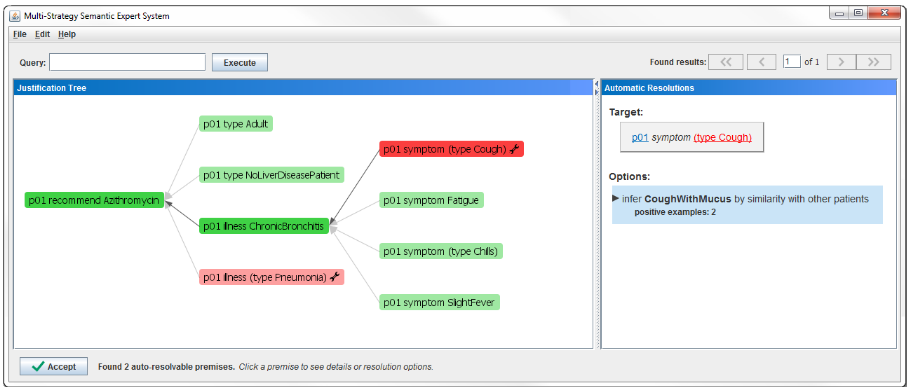

Fig. 13. User *Interface*: Main query interface

Figure 13 shows the main query interface, which presents the user with one or more justifications to explain a posed query, visualized as trees (see Figure 2). As shown, failed premises are highlighted in red. Upon selecting a failed premise, a list of automatic resolution options are shown (see Figure 14, right-panel), as well as a short description. After selecting the analogy resolution, the screen in Figure 15 is shown.

In this screen, the top part describes the plausible rule being applied, using color coding to indicate shared variables/instances between the rule conjuncts. The bottom shows the found unification of the rule (see Section 3.1), as well as the inference that could be made by analogy. As mentioned, ontological knowledge may be leveraged to cope with inexact matches. In this case, *HIVPatient* and *OrganTransplantPatient* do not match exactly, but were found to have a direct common subsumer, namely *ReducedImmunityPatient*. This semantic similarity is presented to the domain expert, together with the calculated score (see Figure 10).

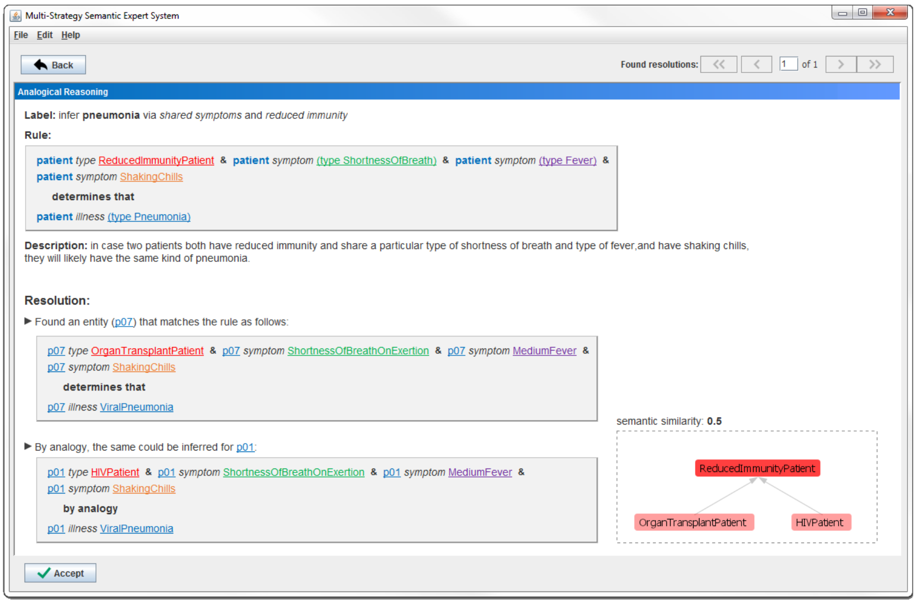

Fig. 14. User *Interface*: Analogical Reasoning screen

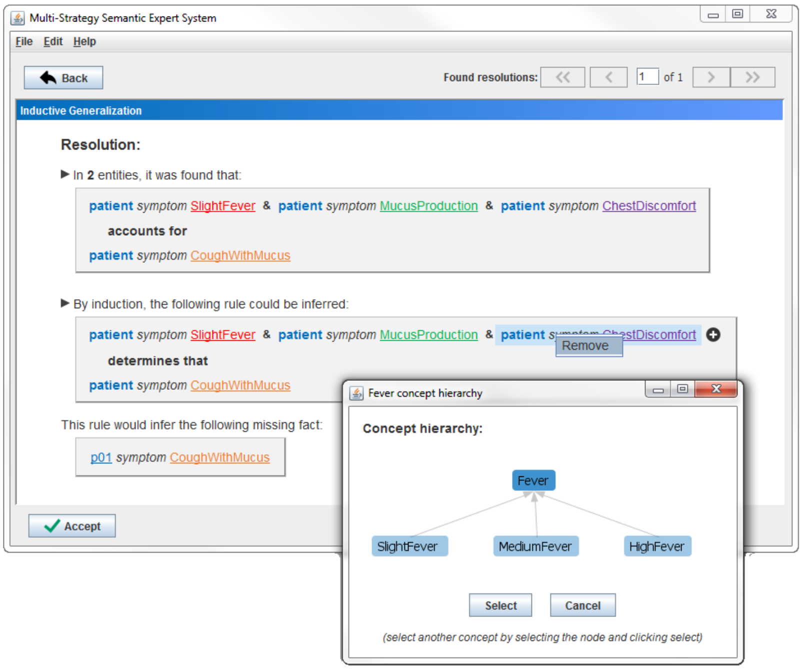

Fig. 15. User *Interface*: Inductive Generalization screen

Figure 16 shows the inductive generalization screen. At the top, the screen indicates properties found by the system that can account for the missing premise (see Section 3.2). Below, the deductive rule is shown that may be induced, accompanied by the missing premise that the deductive rule would infer. Before confirming the inference, the domain expert can fine-tune the rule to cope with inductive bias. Firstly, the expert can select any instance or type to select any of their (super or sub) types in the related concept hierarchy, making the instance or type more general or specific. In this case, the domain expert chooses to replace the *SlightFever* condition by *(type Fever)*, as any kind of fever may indicate a cough. After clicking *SlightFever*, a popup shows the *Fever* concept hierarchy, which allows the expert to choose any related concept. In addition, the expert may add (see "+" icon) or remove any premise from the rule condition. For instance, the domain expert could decide that having chest discomfort is unrelated to cough, and thus remove the premise from the condition.

After accepting an automatic resolution option, the visualized justification tree (see Figure 14) is extended with the plausible inference nodes (highlighted in orange; not shown). In addition, its inferences will be added to the system's working memory. Afterwards, once the domain expert accepts the extended plausible justification, the inferences will be materialized in the knowledge base.

4 Implementation Details

Our system currently relies on Prolog to realize deductive reasoning, and to perform the unification steps during analogical reasoning. Our system further employs Aleph[1] (A Learning Engine for Proposing Hypotheses), an Inductive Logic Programming (ILP) system, to realize analogical reasoning. Custom Java code implements the conceptual similarity check discussed in Section 3.2.

Regarding the expert system's Semantic Web layer, we utilize the well-known Apache Jena framework to apply OWL reasoning on incoming RDF data. Conversion from RDF triples to Prolog and back is performed by custom Java code. Similarly, we wrote Java code to convert SPIN rules to Prolog, utilizing the Topbraid SPIN API. The user interface was implemented using the javax.swing packages, and utilizes the Prefuse graph library for visualizing the justification trees and concept hierarchies.

5 Related work

To deal with incomplete (and partially incorrect) knowledge bases, Tecuci et al. [7, 10] employs justification trees to integrate inferences from multiple learning mechanisms. A three-step, tutor-based approach is suggested, ending with a sufficiently complete knowledge base ready for querying. Instances of justification trees, based on positive training examples and verified by a human tutor, are generalized and added to the knowledge base; allowing future similar inferences to be deductively entailed. Next, other instances are found and verified by the tutor, whereby positive and negative instances are used to fine-tune generalized justifications. Finally, inconsistencies (i.e., negative instances) are resolved by eliciting new knowledge (e.g., concepts). In contrast

[1] http://www.cs.ox.ac.uk/activities/machlearn/Aleph/aleph.html#SEC1

to Tecuci et al., our goal is not to exhaustively extend a knowledge base until it is ready for querying. Instead, an important aim is to supply plausible justifications for individual queries on a case-by-case basis. For instance, in our medical use case, domain experts (e.g., physicians) need to be acutely aware of the rationale behind every diagnosis or drug recommendation, including any related deterministic or plausible inference. In addition, we provide the domain expert with the choice to materialize and fine-tuned plausible justifications in the knowledge base; allowing these inferences to be deductively entailed in the future. In our work, we illustrated how semantic technology can improve this kind of approach. We note that, regarding OWL reasoning, Horridge et al. [16] discuss efficiently obtaining justifications for OWL entailments.

Fuzzy, probabilistic or possibilistic learning methods for dealing with uncertainty could be applied to further supplement justification trees (or initial knowledge bases). A number of works have studied specification and reasoning under uncertainty in OWL [17, 18]. However, as opposed to such learning methods, we note that inferences from inductive and analogical reasoning can be straightforwardly presented to domain experts. Studying the integration of such learning methods in our expert system, while still supplying useful explanations to domain experts, is considered future work.

6 Conclusions & Future work

In this paper, we illustrated a multi-strategy reasoning approach for enhancing query access to medical knowledge bases. As opposed to traditional Semantic Web reasoning, based on a priori deterministic knowledge, this approach additionally leverages data similarities as well as plausible knowledge. In particular, our Semantic Web expert system supplies plausible query explanations to medical experts, by complementing deterministic reasoning with plausible inferences from analogical and inductive reasoning. We illustrated the usefulness of our system by a real-world, medical use case, which, in light of incomplete knowledge, involved applying both deterministic and plausible knowledge. Based on their real-world expertise, domain experts can confirm plausible justifications, and fine-tune the resulting deterministic knowledge. Finally, we showed that weak inferencing can benefit from Semantic Web technology as well, by improving similarity checking and expressivity of plausible knowledge.

Future work consists of applying additional learning methods, and studying how their uncertainty results may be reflected in plausible justifications in a way that is straightforward for domain experts to understand. In the same vein, we aim to associate logical proofs with generated inferences; for instance, reflecting the inferencing steps responsible for inferred knowledge. Based on these “traces”, domain experts could choose to manually revert unsuitable materializations for certain entities. Additionally, we plan to run experiments with a large medical dataset, to study the impact of our integrated, multi-strategy reasoning approach on performance.

References

1. Sim, I., Gorman, P., Greenes, R.A., Haynes, R.B., Kaplan, B., Lehmann, H., Tang, P.C.: Clinical Decision Support Systems for the Practice of Evidence-based Medicine. J. Am. Med. Inform. Assoc. 8, 527–534 (2001).
2. Raymond, L., Paré, G., de Guinea, A.O., Poba-Nzaou, P., Trudel, M.-C., Marsan, J., Micheneau, T.: Improving performance in medical practices through the extended use of electronic medical record systems: a survey of Canadian family physicians. {BMC} Med. Inf. {&} Decis. Mak. 15, 27 (2015).
3. Bates, D.W., Kuperman, G.J., Wang, S., Gandhi, T., Kittler, A., Volk, L., Spurr, C., Khorasani, R., Tanasijevic, M., Middleton, B.: Ten Commandments for Effective Clinical Decision Support: Making the Practice of Evidence-based Medicine a Reality. J. Am. Med. Inform. Assoc. 10, 523–530 (2003).
4. Iqbal, A.M., Shepherd, M.A., Abidi, S.S.R.: An Ontology-Based Electronic Medical Record for Chronic Disease Management. 44th Hawaii International International Conference on Systems Science (HICSS-44 2011), Proceedings, 4-7 January 2011, Koloa, Kauai, HI, USA. pp. 1–10 (2011).
5. Bouamrane, M.-M., Rector, A., Hurrell, M.: Using OWL Ontologies for Adaptive Patient Information Modelling and Preoperative Clinical Decision Support. Knowl. Inf. Syst. 29, 405–418 (2011).
6. Rodriguez-Gonza¡lez, A., Labra-Gayo, Jose Emilio Colomo-Palacios, R., Mayer, M., Gomez-Berbas, J.M., Garcia-Crespo, A.: SeDeLo: Using Semantics and Description Logics to Support Aided Clinical Diagnosis. J. Med. Syst. 36, 2471–2481 (2012).
7. Tecuci, G.: Building Knowledge Bases through Multistrategy Learning and Knowledge Acquisition. Mach. Learn. Knowl. Acquis. Integr. Approaches. (1995).
8. Gentner, D.: The Mechanisms of Analogical Reasoning. Readings Mach. Learn. (1990).
9. Henson, C.A., Thirunarayan, K., Sheth, A.P.: An Efficient Bit Vector Approach to Semantics-Based Machine Perception in Resource-Constrained Devices. 11th International Semantic Web Conference. pp. 149–164. Springer (2012).
10. Tecuci, G.: Plausible Justification Trees: A Framework for Deep and Dynamic Integration of Learning Strategies. Mach. Learn. 11, 237–261 (1993).
11. Danyluk, A.P.: The use of explanations for similarity-based learning. IJCAI. 274–276 (1987).
12. Carbonell, J.G.: Derivational Analogy: a Theory of Reconstructive Problem Solving and Expertise Acquisition. Mach. Learn. An Artif. Intell. approach. 2, (1986).
13. Winston, P.H.: Learning and Reasoning by Analogy. Commun. ACM. 23, 689–703 (1980).
14. Wu, Z., Palmer, M.: Verbs semantics and lexical selection. Proceedings of the 32nd annual meeting on Association for Computational Linguistics. pp. 133–138. Association for Computational Linguistics (1994).
15. Chiabrando, E., Likavec, S., Lombardi, I., Picardi, C., Dupré, D.T.: Semantic similarity in heterogeneous ontologies. In: Bra, P. De and Grønbæk, K. (eds.) HT'11, Proceedings of the 22nd {ACM} Conference on Hypertext and Hypermedia, Eindhoven, The Netherlands, June 6-9, 2011. pp. 153–160. ACM (2011).
16. Horridge, M., Parsia, B., Sattler, U.: Extracting Justifications from Bioportal Ontologies. Proceedings of the 11th International Conference on The Semantic Web - Volume Part II. pp. 287–299. Springer-Verlag, Berlin, Heidelberg (2012).
17. Stoilos, G., Simou, N.: Uncertainty and the semantic web. Intell. Syst. …. (2006).
18. Ding, Z., Peng, Y.: A Probabilistic Extension to Ontology Language OWL. Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04). p. 40111.1–. IEEE Computer Society, Washington, DC, USA (2004).