Ontology driven semantic profiling and retrieval in medical information systems

Mehul Bhatt a,∗, Wenny Rahayu b, Sury Prakash Soni b, Carlo Wouters b

a Cognitive Systems, Universität Bremen, 28359 Bremen, Germany
b Data Engineering and Knowledge Management Group, Department of Computer Science and Computer Engineering, La Trobe University, Australia

Abstract

We propose the application of a novel sub-ontology extraction methodology for achieving interoperability and improving the semantic validity of information retrieval in the medical information systems (MIS) domain. The system offers advanced profiling of a user’s field of specialization by exploiting the concept of sub-ontology extraction, i.e., each sub-ontology may subsequently represent a particular user profile. Semantic profiling of a user’s field of specialization or interest is necessary functionality in any medical domain information retrieval system; this is because the (structural and semantic) extent of information sources is massive and individual users are only likely to be interested in specific parts of the overall knowledge documents on the basis of their area of specialization. The prototypical system, OntoMOVE, has been specifically designed for application in the medical information systems domain. OntoMOVE utilizes semantic web standards like RDF(S) and OWL in addition to medical domain standards and vocabularies encompassed by the UMLS knowledge sources.

1. Introduction

The new era of e-health information systems has introduced a number of research and development issues whereby the efficient integration of various health information domains is needed and timely retrieval and access to best practice information resources is vital. In general, underlying these issues are the closely related themes of interoperability of medical information sources and the efficient retrieval of semantically correct data from them [1,27]. In the last decade or so, the application of ontologies in information systems as a shared platform for information integration and for establishing a consensus on meaning has been promoted [19]. Grounded in the fundamental problems of integration and retrieval, applications of ontologies in the medical information systems (MIS) domain have been aplenty [10,14]. A comprehensive review of ontological applications in MIS is beyond the scope of this paper. As specific references, see [2] for a work on constructing medical domain terminological systems and [38] for a review of biomedical domain specific ontologies.

1.1. Ontologies on the MOVE

In this paper, we present our system OntoMOVE – ‘Ontologies on the MOVE’ – that combines the use of ontology driven annotations and the application of a novel sub-ontology extraction methodology (called MOVE) for achieving interoperability and improving the effectiveness of information retrieval for the specific MIS domain. The main difference between our work and existing work in ontology-based information retrieval is the fact that our approach begins with the process of extracting a sub-ontology that meets the user requirements, which is then followed by the Contextualization of the sub-ontology through annotating existing documents or resources to the specified sub-ontology. One of the most significant objectives of our approach is to reduce the search space of information retrieval by establishing a semantic scope through the sub-ontology Contextualization and profiling. Our ontology is based on the Unified Medical Language System (UMLS) knowledge sources, namely the UMLS Semantic Network (UMLS-SN) and UMLS Metathesaurus®. As such, the approach is compatible with controlled vocabularies and classifications used in patient records, administrative health data, bibliographic and full-text databases that are based on the UMLS initiative [11,23,48]. Although we restrict ourselves to a specific type of resource in this paper, namely information documents contained in the Medical Therapeutic Guidelines (TG) [45], it should be noted that the precise nature of the resources being retrieved is of no relevance to the

Article info

Article history:
Received 5 September 2008
Received in revised form 8 March 2009
Accepted 27 May 2009
Available online 12 June 2009

Keywords:
Semantic contextualization and profiling
Medical information systems
e-Health information systems
Ontology
Semantics
Knowledge representation and data models
Interoperability

Corresponding author.
Tel.: +49 421 218 64 237; fax: +49 421 218 98 64 237.
E-mail addresses: mbhatt@gmail.com, bhatt@informatik.uni-bremen.de (M. Bhatt), w.rahayu@latrobe.edu.au (W. Rahayu).

Available online 12 June 2009
Accepted 27 May 2009
Received 5 September 2008
Article history:

system (see Section 5.1). For instance, resources could even consist of patient records or other forms of health or operational data generated over a period of time. As long as annotational requirements (Section 5.3) of the framework are met, any type of resources repository can be integrated within the system presented in this paper.

1.2. Ontology, semantics and MOVE

The use of an ontology is intended to produce semantically correct results whilst retrieving information from knowledge sources consisting of large repositories of medical resources. The retrieval phase of the OntoMOVE system is based on the specialization of a general Materialised Ontology View Extraction (MOVE) framework for the medical domain exemplar presented in this paper. MOVE is our system for deriving semantically correct and independent sub-ontologies from potentially large base ontologies [6,8,9,50]. We exploit the general capability of the MOVE framework for deriving sub-ontologies toward operationalizing the concept of ‘semantic profiling’ (Sections 4 and 6) of user’s requirements based on the field of specialization and/or interest. The concept of semantic profiling is based on the premise that different users/customers/stakeholders of medical data have different informational requirements. This is especially important, keeping in mind the structural and semantic extent of most realistic medical domain information sources, such as the medical TG. Indeed, an individual user is not likely to be interested in every conceivable category of medical data that is present in the information sources or repositories. For instance, keeping in line with the medical domain application scenario discussed in this paper, consider the case of retrieving information from the medical therapeutic guidelines. Here, potential users include researchers, medical practitioners and even patients. In this scenario, a pharmacist may be more interested in drug and treatment related information whereas a biomedical scientist may be more interested in information about the latest findings related to certain cell malfunctions or new gene products. To serve user-specific (e.g., pharmacist, medical practitioner, biomedical scientist and so forth) information requirements, semantic profiling of their requirements is therefore a necessary functionality in any medical information retrieval system.

1.3. Organization

The rest of the paper is organised as follows: Section 2 consists of a review of work related to the use of ontologies in medical information systems. Section 3 presents a brief overview on the use of ontological formalism in the medical domain. The use of the OWL as a medium for information representation for UMLSKS content is introduced and the representational capability of OWL is highlighted. In Section 4, we illustrate the concept of sub-ontology extraction using the MOVE framework and also explain the intuition that underlies the use of sub-ontologies as semantic requirement profiles. In Section 5, all aspects of the OntoMOVE framework are presented in detailed and the framework is experimentally evaluated in Section 8. Finally, we conclude in Section 9 with a brief discussion of our approach and pointers to the future directions of this work.

2. Related work

A recent study conducted by Gartner, Inc. [17] indicates that an integrated ‘Semantic Web’ will be one of the highest impact emerging technologies in the next five to ten years and that many public domain industries have started to engage in this technology. Ontologies are utilized as a foundation to enable interoperability within the Semantic Web, and as a result the number of domain ontologies have grown significantly in the last few years. In the medical domain, researchers and health care standard bodies have also started to introduce semantic standard encoding for their clinical data specification [10]. It is crucial that medical information systems are built with an underpinning technology that supports global information interactions and management, and the various works in ontology have played an important role in this area.

The past few years have witnessed a range of applications and studies in the area of ontology management and processing. From the viewpoint of this paper, we categorise existing research and/or systems into the following groups: (a) foundational work encompassing topics such as ontology evolution, ontology editing and alignment, ontology merging, etc., (b) applications of ontologies in diverse domains, (c) ontological tools that provide general support within arbitrary domains of interest, and (d) specialized information retrieval systems in the medical/biomedical domain that utilize ontologies in some way. Here, we selectively describe some of the existing works from these categories and outline how our approach differs from that of existing systems.

2.1. Ontology evolution

Research in this area encompasses foundational work that focus on the ontology manipulation and tailoring techniques with an ultimate goal of ontology reuse and integration. The work by Maedche et al. [29] covers the area of ontology reuse and evolution in the context of ontology management within a distributed system environment. Their proposed method allows the creation of a new ontology by reusing an existing ontology, whilst taking into consideration ontology evolution and integration given the fact that the created ontologies are distributed on many different sites. Similar work by the same authors in [30] also describes the notion of ‘ontology registration’ in order to provide means to locate existing ontologies for reuse. Whilst this work has addressed many important issues in a distributed ontology environment, there are a few areas that have not been fully addressed. Firstly, the proposed method lacks a proper technique to optimize the created (reused) ontologies—although an algorithm to check the validity of the ontology is proposed, the method does not include a mechanism to derive the most optimum ontology. Note that optimality involves several criteria revolving around the size of the resulting ontology, e.g., semantic simplicity and/or the minimization of redundant content in the resulting ontology. Secondly, the proposed technique only focuses on the extraction of a new ontology from an existing ontology, without consideration for retrieving other artefacts or resources that might be linked or annotated against the existing ontology—any refinement of the structure or extent of the ontology bears a direct relation to its semantic scope within whatever resources have been described using that ontology.

2.2. Ontological applications

The second category of existing works in the ontology area include a variety of ontology-based domain specific applications. In particular, there have been numerous research and developments in the area of medical ontologies as mentioned in Section 1. Most of these works can be categorised into: utilizing medical ontologies to build a knowledge repository such as [39,46], consolidating and merging biomedical ontologies such as [4,26,28], integration of medical terminologies [16], and ontology based collaborative work in medical domain such as [39]. All these existing applications are focussing on utilizing a domain ontology (in this case a medical ontology) to support collaboration and communication between resources, domain experts, etc., in the area. As opposed to utilizing the whole large ontology or engineering a new medical ontology [12], our work focuses on extracting and optimizing a sub-ontology
from a given large domain ontology that fulfills a user requirement profile.

2.3. Ontology engineering tools

It is important to differentiate the system proposed here with existing tools for ontology editing and alignment such as Protégé [33,40], or OntoEDIT [34]. Whilst the above tools provide efficient techniques to create, view, visualize, edit and align ontologies, they do not particularly address the issues of (i) user-driven automatic extraction of valid sub-ontologies, and (ii) semantic and structural optimization of the resulting ontologies. In addition, these tools do not address application level annotation and semantic mapping.

2.4. Specialized information retrieval systems

There are many different information retrieval and browsing systems from a specialized medical/bio-medical viewpoint. Principal among them include Textpresso, GoPubMed and XplorMed. Textpresso is an information retrieval and extraction system that processes the full-text of biomedical papers [31]. The system tokenizes text into several categories with respect to an ontology. GoPubMed is a information retrieval engine that presents PubMed results in an ontology-based hierarchical form [15]. The focus here is on presenting a dynamic taxonomical view for user’s queries based on ontologies. Similarly, XplorMed is an exploratory tool that is aimed at overcoming the limitations of keyword based search [37]. The contributions of our research are foundational and are, in principle, aimed at leveraging upon the utility provided these or other specialized information retrieval algorithms and tools: we focus on the semantic profiling of a user’s requirements prior to performing a query so that an underlying information retrieval system has a better approximation on the scope of the user’s interest within a resource or document repository.

All of the above-mentioned works highlight the increase of interest to utilize ontology as a means to standardize processes or tasks. However, despite these recent efforts, there has been no real focus on tailoring the ontologies to meet user-specific needs as well as to integrate them with the extraction of ontology-annotated data sets or resources. Our proposed techniques will play an important role in applying the notion of reuse in ontology engineering.

3. Ontological representation for the medical domain

Ontologies play a pivotal role by providing a source of shared and precisely defined terms that can be used as meta-data, e.g., annotation of information-sources and other resources in order to make them accessible to automated agents. Although there are inherent distinctions between a taxonomy and an ontology, ontologies as typically used on the semantic web and software engineering applications consist of a hierarchical description of important concepts in a domain, along with descriptions of the properties of each concept. The degree of formality employed in capturing these descriptions can be quite variable [32], ranging from natural language to logical formalisms, but increased formality and regularity clearly facilitates machine understanding [21,44].

3.1. OWL—a formal knowledge representation structure

The Web Ontology Language (OWL) is a knowledge representation scheme designed specifically for use on the semantic web; it exploits existing web standards (XML and RDF), adding the familiar ontological primitives of object and frame based systems, and the formal rigor of a very expressive description logic (DL) that emerges from research in the field of Artificial Intelligence [22]. As exemplified in Tables 1 and 2, OWL consists a rich set of knowledge representation constructs that can be used to formally specify medical-domain knowledge, which in turn can be exploited by description logic reasoners for purposes of inferencing, i.e., deductively inferring new facts from knowledge that is explicitly available. The knowledge base (KB) of a typical DL based system comprises of two components, the TBox and the ABOX. The TBox introduces the terminology, i.e., the vocabulary of an application domain (e.g., ‘Neoplastic Process is a Biological Function’), whilst the ABox contains assertions about named individuals in terms of this vocabulary (‘Cancer is an instance of a Neoplastic Process’). The logical basis of the language means that reasoning services can be provided in order to make OWL described resources more accessible to automated processes thereby allowing one to infer implicitly represented knowledge from the knowledge that is explicitly contained in the knowledge base. From a formal point of view, OWL can be seen to be equivalent to a very expressive DL, with an OWL ontology corresponding to a DL terminology (TBox) whereas instance data pertaining to the ontology making up the assertions (ABox).

Our use of the OWL language to represent the medical ontology is driven by the fact that OWL is industry standard and is recommended by the W3C [49] for the representation of ontologies. Furthermore, numerous semantic web tools, for example, Protégé [18,40] and its associated OWL Plugin [41], OntoMat [35], etc., supporting OWL have been already developed in the open-source community. In addition, tool builders have developed powerful reasoning systems that support reasoning with ontologies represented in the OWL language, the best example here being RACER [42]. As a part of further extensions to the work presented in this research (Section 9), we envisage to apply formal description logic based reasoning functionality supported by such tools in the medical domain.

<table>
<thead>
<tr>
<th>Table 1</th>
<th>OWL axioms.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axiom</td>
<td>DL syntax</td>
</tr>
<tr>
<td>Sub class</td>
<td>C1 ⊑ C2</td>
</tr>
<tr>
<td>Equivalent class</td>
<td>C1 ≡ C2</td>
</tr>
<tr>
<td>Disjoint with</td>
<td>C1 ⊑ ¬C2</td>
</tr>
<tr>
<td>Same individual</td>
<td>x1 = x2</td>
</tr>
<tr>
<td>Different from</td>
<td>x1 = ¬x2</td>
</tr>
<tr>
<td>Sub property</td>
<td>P1 ⊑ P2</td>
</tr>
<tr>
<td>Equivalent property</td>
<td>P1 = P2</td>
</tr>
<tr>
<td>Inverse</td>
<td>P1 = P2</td>
</tr>
<tr>
<td>Transitive property</td>
<td>P* ⊑ P</td>
</tr>
<tr>
<td>Functional property</td>
<td>¬(∃x) P x</td>
</tr>
<tr>
<td>Inverse functional property</td>
<td>¬(∀x) P x</td>
</tr>
</tbody>
</table>

<p>| Table 2 | OWL class constructors. |</p>
<table>
<thead>
<tr>
<th>Constructor</th>
<th>DL syntax</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intersection</td>
<td>C1 n...n Cn</td>
<td>AnatomicalAbnormality n PathologicalFunction</td>
</tr>
<tr>
<td>Union</td>
<td>C1 u...u Cn</td>
<td>Body_Substance u Organic_Chemical</td>
</tr>
<tr>
<td>Complement</td>
<td>¬C</td>
<td>¬Invertebrate</td>
</tr>
<tr>
<td>One of</td>
<td>x1 u...u xn</td>
<td>Oestrogen u Progesterone</td>
</tr>
<tr>
<td>All values from</td>
<td>∀P,C</td>
<td>Vco_occurs_withPlant</td>
</tr>
<tr>
<td>Some values</td>
<td>∃P,C</td>
<td>∃coc_occurs_withAnimal</td>
</tr>
<tr>
<td>Max cardinality</td>
<td>≤nP</td>
<td>≤1has_ingredient</td>
</tr>
<tr>
<td>Min cardinality</td>
<td>≥nP</td>
<td>≥2has_ingredient</td>
</tr>
</tbody>
</table>

The Unified Medical Language System Knowledge Source Server (UMLSKS) has been used in this work to gain access to the vast amount of knowledge contained in the UMLS by way of its two main components, viz—the UMLS Metathesaurus® and the UMLS...
Semantic Network. Fig. 1 is a rather narrow view of the semantic network and is indicative of the sort of semantic types present in it. The information provided by the UMLSKS is accessed through a UMLS Plugin [47] that can be integrated with the Protégé ontology engineering environment [40]. This approach is useful and time-saving for our project since the plugin facilitates easy browsing of the UMLS Semantic Network and Metathesaurus in addition to supporting a mechanism whereby the knowledge contained therein may be easily integrated into our ontology. The UMLSKS allows a user/application to request information about particular Metathesaurus concepts, including attributes such as the concept’s definition, its semantic types, concepts that are related to it, hierarchical context details, etc., all of which can be restricted to source specific details.

4. Sub-ontology view extraction for semantic profiling

As information on the web increases significantly in size, Web Ontologies also tend to grow bigger to such an extent that they become too large to be used in their entirety by any single application (e.g., the UMLS base ontology [48]). Also, because of the size of the original ontology, the process of repeatedly iterating the enormous number of nodes and relationships to derive a sub-ontology becomes very computationally extensive, thereby necessitating the use of parallel and/or distributed techniques for the extraction of the same from a massively sized base ontology. These problems have stimulated our work in the area of sub-ontology extraction in a general context and its optimization using a distributed architecture.

4.1. Materialized ontology view extraction

The extraction process, referred to as Materialised Ontology View Extraction (MOVE) [7,8,50], is capable of deriving sub-ontologies, also referred to as materialised views, from a (typically large) base ontology. More important in the context of this paper, it should be noted that the resulting ontology is semantically complete [6] and a valid ontology independent of the base ontology. This is achieved in MOVE through the enforcement of relevant constraints that take the form of various ‘optimization schemes’ such as requirements consistency, semantic completeness, well-formedness, and total simplicity, which ensure the correctness and the optimality of the resulting sub-ontology. Broadly, the following four categories of optimization schemes (OS1–OS4) are applied for deriving a independent sub-ontology from a base ontology:

OS1. Requirements consistency: before a sub-ontology is derived, a user or application must provide an approximate indication of its interests in the form of a ‘labelling’ (Section 6) of conceptual categories in the base ontology. Requirements consistency checks for the consistency of these user specified labelling/requirements for a specific sub-ontology derivation process. Henceforth, we refer to this labelling as the r requirements specification.

OS2. Semantic completeness: semantic completeness considers the completeness of the concepts, e.g., if one concept is defined in terms of another concept, the latter cannot be omitted from the sub-ontology without loss of semantic meaning of the former concept.

OS3. Well formedness: it might be possible that the user requirements (labelling) is consistent, but there might be statements that inevitably lead to a solution that is not a valid ontology. Well-formedness contains the proper rules to prevent this from happening.

OS4. Total simplicity of solution: finally, applying total-simplicity to an existing solution (along with its requirements specification) will result in the smallest possible solution that is still a valid ontology. Total-simplicity achieves this by working on not only the solution, but also its requirements specification.

An elaboration of these optimization schemes in (OS1–OS4) and the overall extraction process of Fig. 2 is described in our earlier work [7,8,50]. However, what is relevant in the present context is the manner in which the specialization and application of MOVE for deriving sub-ontologies may be utilized as a foundational basis of user-specific semantic requirement profiling. An overview is presented in (P1–P2):

P1. The extraction process: Fig. 2 shows a schematic of the sequential extraction process. The process begins with the import of the ontology represented in the OWL language. The actual extraction process/execution of optimization schemes is initiated by way of requirements specification by a user or another application and the execution of the other optimization schemes. The externally provided requirements specification is used as the basis of the derivation/extraction process. The derived sub-ontologies are valid independent ontologies, known as Materialized Ontologies, that are specifically extracted to meet certain user/application needs. The result of the extraction process is not just simply an extracted sub-ontology, but rather an extracted materialized ontology view. In the extraction process, no new information is introduced (e.g. adding a new concept). However, it is possible that existing seman-
5. OntoMOVE: a semantic requirement profiling and retrieval framework

Our earlier work in [9] describes the general framework of OntoMOVE for the purpose of ontology reuse in software development processes in general. In this paper, we highlight the idea of optimizing the processes of ontology-based information retrieval through the utilization of sub-ontology annotation and extraction. We also demonstrate the effectiveness of our proposal through a prototypical implementation in a medical domain with large size document resources in the form of the Medical Therapeutic Guidelines (TG) [45].

5.1. Overview of the framework

OntoMOVE stands for ‘Ontologies on the MOVE’. The term ‘MOVE’ itself is an acronym for ‘Materialised Ontology View Extraction’ (see Section 4), which is foundational to the work reported herein. A brief overview of the main aspects of the OntoMOVE framework illustrated in Fig. 3 follows in (F1–F6):

F1. Component requirement specification: We use the general term ‘component’ to refer to an application or user that can supply a set of requirements/preferences that are expressed as a partial labelling of a base ontology. This labelling is indicative of the semantic types (concepts and relationships) that reflect the component’s field of specialization or interest. Indeed, for practically deployable scenarios, we do not expect users to manually provide such a labelling. However, its provision is achieved as such in our prototypical implementation. Henceforth, this will be referred to as a ‘component requirement specification’ or simply a requirement specification. Also, note the term ‘requirements specification’, which is utilized in several disciplines such as software and process modelling, where, in general, it refers to the description of a system or process being modelled [24]. The usage of this term in this work must be disambiguated—we categorically state that here it refers to the approximately incompletely specified knowledge by an user or an application program in the form of a partial labelling of a base ontology.

F2. Requirement profile: On the basis of the initial requirement specification, a profile of the requesting component is derived (see ‘requirement profile derivation’ in Fig. 3) based on the concept of a sub-ontology (Section 4). A component requirement profile is a complete requirement specification that is derived using the partial (initial) requirement specification that is provided by the component. Between the initial requirements specification for a component and the derivation of its complete semantic requirement profile lies the derivation of sub-ontologies (i.e., MOVE), as explained in Section 4. Finally, it is instructive to disambiguate a component’s ‘requirement specification’ from a component’s ‘semantic requirement profile’. Whereas the former is a partial specification of interest from a base ontology, the latter is a complete, independent sub-ontology derived from the base ontology on the basis of the former. The derivation itself is achieved by the application of the extraction process in (P2), i.e., the optimisation schemes in (OS1–OS4).

F3. Resource: We subscribe to a general notion of a resource since the actual type of a resource is irrelevant to the resource acquisition framework, i.e., OntoMOVE. However, in so far as this paper is concerned, we focus on resources as being unstructured or semi-structured data sets (i.e., medical therapeutic guidelines) that are of interest in medical information retrieval domain that is being exemplified in this paper. However, in general, the assumption that is applicable in this context is that irrespective of the precise type of a resource, the resource under consider-

---

1 We disambiguate, and use this term to conform with its usage as defined in previous works [7,8,50].

2 The concept of completeness is non-trivial and involves qualitative benchmarks along several dimensions. Details are presented in [50].
The UMLS semantic network (UMLS-SN) represents the meta-level of the medical ontology that we develop for purposes of annotation. This is because the UMLS Metathesaurus uses UMLS-SN as its meta level to define medical domain concepts from various medical vocabularies. We represent selective parts of UMLS-SN taxonomies in the form of a subsumption hierarchy in the OWL language, i.e., as an OWL ontology (referred to as the UMLS-SN ontology henceforth). Fig. 4 illustrates a limited part of the entire UMLS-SN ontology in the form of a subsumption hierarchy in the OWL language, although all semantic types have been mapped, we have not mapped every property or relationship that exists between the semantic types since the set of relationships between the semantic types is too massive to be used in its entirety and also because it is not our objective to develop a comprehensive mapping of the entire UMLS-SN in the OWL language. Instead, we intend to incrementally enrich and extend the SN relationships only in so much as what is required for us to encompass a given collection of medical TG documents. We use Protégé [40] and its associated OWL plugin [41] for the ontology development tasks. As mentioned previously, the Protégé UMLS tab (i.e., a plugin) [47] supports browsing of the UMLSKS and can be used to query the UMLS knowledge sources for medical terms and retrieve the results in a terminological format. The results can be exported into our own medical ontology. Obviously, the ontology that results is basically a very small view of the base UMLSKS—one that concerns our use of UMLSKS. The resulting ontology is imported in OntoMat [35] in synchrony with the medical information sources, namely the Medical Therapeutic Guidelines (TG) [45]. A subset of the guidelines can then be annotated using the ontology that we imported into OntoMat following which the modified (annotated) documents are serialized as a different version for purposes of testing.

### 5.3. Semantic categorization of information sources

We are using OntoMat [35], which is a publicly available annotation tool, for purposes of annotating the collection of Medical Therapeutic Guidelines (TG) used in this project. OntoMat annotator supports the task of creating and maintaining ontology-based OWL mark-ups, i.e., creation of OWL-instances, attributes and relationships. It includes an ontology browser for the exploration of the ontology and instances and an HTML browser that displays the annotated parts of the text. OntoMat serves our purposes since it allows easy import of externally develop OWL ontologies and HTML mark-up, which in our case are the UMLS-SN ontology and the Medical TG respectively.

#### 5.3.1. The annotation process

An annotation refers to the instantiation of an UMLS-SN ontology class in order to relate a chosen ‘term’ from a TG document with a semantic type from the UMLS-SN ontology. To preserve consistency with industry based medical vocabularies, we constrain our selection only to those terms that are identifiable with the concepts that are present in the UMLS Metathesaurus. The chosen term’s placement in the ontological hierarchy is selected on the basis of its registered UMLS Semantic Network type in the thesaurus. Presently, the selected terms are essentially keywords present in the static HTML pages of the Medical Therapeutic Guidelines (TG) with the annotations being embedded in the TG documents using a pre-defined RDF vocabulary. For example, in Listing 1, the RDF-code representing one annotation object has been represented—the annotation object consists of the following information:

1. The semantic type from the UMLS-SN ontology of the UMLS Metathesaurus term being annotated.
2. The exact location of the term within the Medical Therapeutic Guidelines using XMLPointers.
3. The precise term being annotated and a well-formed label associated with it meant for internal/system use.

Although the annotation objects are instances of semantic types in the UMLS-SN ontology (i.e., the TBox), the RDF based instantiations are different from the actual instances (i.e., the ABox) that corresponds to the UMLS-SN terminology. Note that the example in Listing is only one form of annotation; additionally, we also distinguish between other forms of annotations, namely those obtained.

---

1. The ontological or knowledge engineering perspective to be applied whilst mapping all parts of the UMLS semantic network to the OWL language is problematic [25].
by establishing relationships between the UMLS-SN semantic types by way of datatype and object properties. Whilst such properties are very important from an information retrieval viewpoint since they establish links between disparate but conceptually related information sources, we regard such links to be only orthogonal to the main annotation task (by a domain expert) that establishes the semantic types of medical terms contained within the therapeutic guidelines. Once the main annotation task is achieved, the secondary annotation task of providing fillers or values for the domain and range of properties can be handled even by a non-specialist on the basis of the UMLS-SN semantic types and their instances that are already obtained via term annotation by the domain expert.

**Listing 1**: Annotation object.

```xml
<urn:ULS-SN:ONTOLOGICAL-RELATIONSHIP-
   rdf:about="http://ontoserver/tgc/rsg/1078.htm#LARGE_VOLUME.NEBULIZER.DEVICE">
   <rdfs:label>Large volume nebulizer device</rdfs:label>
</urn:ULS-SN:ONTOLOGICAL-RELATIONSHIP-

5.3.2. Scope of annotations

For our present demonstrative purposes, 1156 UMLS Metathesaurus terms from a collection of 170 TG documents were selected for annotation on the basis of their UMLS-SN semantic type. Toward this task, 98 semantic types from a total of 135 have been utilized; with the difference representing those semantic types that did not have any instances in the selected document collection. It must be pointed out that these metrics (and others available within our system) are useful in determining the quantity and quality of the annotations being performed and can be dynamically obtained from within the system. For flexibility, the Annotation-Indexer (Section 5.3.3) and its associated statistics generation capabilities (for examples, see Sections 7.1 and 7.2 and the evaluation in Section 8) have been implemented to be usable either with or independent of the main OntoMOVE application.

5.3.3. Dynamic annotation and indexing

Dynamic annotation and indexing, in so far as this work is concerned, refers to the capability to dynamically integrate new annotations and build respective index entries thereby supporting integration of incremental updates and extensions to the underlying information sources (e.g., medical therapeutic guidelines). The new annotations could either belong to an existing index or they could be entirely new set of annotations for a different domain altogether. This functionality is important and essential for primarily two reasons, as stated in the following:

1. Incremental annotations and information update: Annotation is not a one time task and is indeed often performed in an incremental manner. This is primarily because the task is inherently qualitative (or human expertise driven) in nature and as such, requires considerable refinements before a complete and consensual semantic categorization of data is obtained. Furthermore, existing information sources may possibly be extended so as to include completely new information thereby necessitating new annotations.

2. Applicability in multiple-domains: We presently focus on semantic information retrieval the medical domain. However, the OntoMOVE framework is applicable in domains other than the one considered herein. As such, in order to preserve the generality and re-usability in other domains, it is necessary that the system remain functional when annotations for information sources from other domains are being utilized. This concept if re-usability is illustrated in Fig. 5: the annotation database consists of multiple indexes, with each possibly corresponding to a different set of information sources from which data needs to be retrieved.

A built-in Annotation-Indexer maintains and builds the relationships, namely index entries, between the existing semantic types and their instances in the medical TG. For purposes of efficiency,
field of specialization from the overall collection of TG is defined to be the mapping produced by a independent sub-ontology of the base ontology that reflects the semantic types related to the user's interest. Obviously, different sub-ontologies will map to different sets of documents/resources in the overall TG set, with the mapping information coming from a (pre-computed or dynamically generated) annotation index. This is denoted in Fig. 6 by the dashed-arrow connecting the two dashed-circles. As discussed in Section 4, the semantic requirement profile itself is derived by applying the sub-ontology extraction algorithm on an incompletely specified 'requirements specification' supplied by a end-user or application.

In Fig. 7, we apply the general idea of requirement driven resource retrieval (in Figs. 6 and 3) for the bio-medical domain. A brief description of the entire process, illustrated with an application-centric viewpoint in Fig. 7, follows:

**Stage 1.** Domain knowledge base: The domain ontology which represents the entire knowledge within the specified domain. In this example, we utilize and extract from UMLS (Unified Medical Language System) and GO (Gene Ontology) as the two base ontologies to contextualize the bio-medical area.

Requirements specification: a set of concepts and properties from both the UMLS-SN and GO base ontology are extracted on the basis of the area of interest or specialization of the end-user. Note that this specification is rather crude and does not constitute a valid sub-ontology. Basically, this specification takes the form of a 'labelling' of concepts, attributes and relationships that are of user's interest. For example, labelling for concepts related to 'Cancer'

**Stage 2.** Conceptualizing the requirements specification: Our system will extract a valid sub-ontology based on the requirement specification labelling as mentioned in Stage 1 above. The sub-ontology is indicative of the user's preferences as well as some implicit relevant concepts considered essential for inclusion by our system to make it a valid sub-ontology. For example, the valid sub-ontology for 'Cancer Treatment' is created from the UMLS, and a sub-ontology for 'Cancer Genome' is created from the Gene Ontology.

Semantic context based information retrieval: once a context has been established, all search queries that are performed within the context will narrow the range of the information documents to include only those documents (or their parts thereof) that are relevant to the context. Note that this functionality is encapsulated within the application of the sub-ontology extraction algorithm. In our example, we will retrieve all documents related to 'Cancer Palliative Treatment' from within the palliative care document repository (e.g., TG) and documents related to 'Cancer DNA Mutation' from the biosciences PubMed document repository.

**Stage 3.** Finally, collaborating end-users with disjoint and even overlapping specializations in the base ontology may obtain context specific sub-ontologies, and consequently, specialized search results whilst looking-up annotated resources in an annotated resource database. For example, the Centre for Cancer Genome would focus on the specific information from the Gene Ontology context, whereas a general Cancer Research Centre may need to create a context that combines concepts from the Medical ontology ('Cancer Treatment') as well as the Gene Ontology ('Cancer Genome' information).

In essence, there are 3 types of changes that have to be taken into consideration in the aforementioned OntoMOVE stages:

- Document collection updates: in this case, the Contextualization process in Stage 2 will need to be incrementally updated each time a new set of documents are required to be included or removed.
- User requirement changes: a new sub-ontology based on the new set of requirements will need to be created. The new sub-ontology can be created simply as a new 'version' of the previously built sub-ontology.
- Domain ontology evolution: whilst ontology is expected to be mostly stable, there are cases where the main ontology evolves. In this situation, the changes will have to be broadcasted to all relevant sub-ontologies. Whilst this is an important issue to address, the main focus of this paper is on the novelty of our proposed extraction process.

### 7. OntoMOVE: an implementation overview

We briefly present the accessibility information relevant to retrieving information from the medical sources via the OntoMove Search Interface (see Fig. 8(a)). The user performs a search query consisting of N keywords. In addition, the user may also specify the means of combining the keyword, viz - either conjunctively or disjunctively. By default, the ‘AND’ operator will be used to combine the search keywords in case where \(N > 1\). In addition, the user may optionally choose to apply a profile (Fig. 9(a)) in the context of which the search is to be performed. Note that it is also possible to apply more than one profile at the same time, for example, applying a patient and pharmacist profiles in conjunction. As discussed previously, the effect of applying a profile (or a combination thereof) is to establish context based on semantic information about the users area of interest that is contained in the profile. As can be seen in Fig. 8(a), the retrieved search results are sequentially listed in the lower part of the interface—the precise order of the results is based on the conceptual similarity/distance between the identifiable semantic type of the search keywords with the anno-

---

6 In Fig. 6, solid lines denote information flows whereas dashed lines refer to conceptual links between related components.
tations that are present in the retrieved documents. In addition to a brief summary of every document that is retrieved, the user also has access to a detailed semantic Report (Fig. 9(b)) and an utility-oriented Annotation Viewer (Fig. 8(b)). A brief description for the two is provided in the sections to follow.

7.1. Detailed semantic report

The report consists of semantic information about other Metathesaurus items, which might possibly be of interest to the user, that are present in a certain document retrieved as a result of the search process. For an example, see the report in Fig. 9(b) that is obtained for the very first result/document out of a total result set of 7 documents by searching for the key-words—'Palliative Care Cancer'. Note that in addition to a provision for easy navigation of search results, the interface (see middle portion of Fig. 8(a)) also allows for direct browsing of the hyperlinked information sources directly in the form of a tree-structure. This tree-structure is obtained solely on the basis of the structural organisation provided by the information source and is derived (only once) by parsing the complete set of HTML based information sources. As a pointer to future work, it would be interesting to obtain a dynamic taxonomy from the set of documents that are retrieved for a search query in question. However, this is beyond the scope of our project and presently
remains a speculative note deemed worthy of further investigation.

7.2. Annotation viewer

An ‘Annotation Viewer’ (Fig. 8(b)) is available wherein the conceptual range of any arbitrary semantic type form the UMLS-SN ontology within the TG documents may be obtained. The viewer exploits the existing annotation indexer and/or a previously serialized index (if one is available). For every semantic type within the presently loaded ontology (or sub-ontology if profiles are active), the steps illustrated in Listing 2 are performed to determine the range of a selected semantic type or property.

Fig. 8. OntoMOVE information retrieval interface I. (a) TG information retrieval. (b) Annotation viewer.

Fig. 9. OntoMOVE information retrieval interface II. (a) Profile selector. (b) Semantic report.
Listing 2: Establishing semantic scope.

```java
/*
   INPUT : A semantic type (concept or property) from the requirement profile (R).
   OUTPUT: A set of concepts that are either directly or indirectly related to a named concept/property (ResultSet).
*/

//Main Function to get Semantic Scope
1. function getSemanticScope () : ResultSet
{
    ResultSet = {}; // In the following, we briefly explain the algorithm in Listing for establishing the semantic scope of semantic type from the profile within a annotated resource repository: firstly, if a primitive concept is selected, then it is simply added to the established semantic scope denoted by ResultSet. If a property is selected instead, then the task is to find the Domain (domainConcept) and Range (rangeConcept) of the selected property and perform this algorithm for each of them. In the final step of the algorithm, the ResultSet that is obtained is essentially a set of concepts from the profile. This set is complemented with all concepts that have been defined to be equivalent (using OWL: sameAs property) to any concept present in the original result set. This final step (potentially) widens the semantic scope of the results by including the synonym concepts.
}
}

1.1 if ( R is Concept )
{
    ResultSet = ResultSet ∪ getSubConcept ( R );
}

1.2 else if ( R is Property )
{
    domainConcept = domain(R);
    ResultSet = ResultSet ∪ getRelatedConcept ( domainConcept );
    rangeConcept = range(R);
    ResultSet = ResultSet ∪ getRelatedConcept ( rangeConcept );
}

tempSet = NULL;
Loop through each Concept in ResultSet
{
    C = next ( ResultSet );
    tempSet = tempSet ∪ getEquivalentConcepts( C );
}

1.4 return ( ResultSet ∪ tempSet )
}

//Helper Function to get Related Concepts ResultSet
2. function getRelatedConcepts ( Concept R ) : ResultSet
{
    PartialResultSet = NULL;
    if ( R is Primitive )
    {
        PartialResultSet = PartialResultSet ∪ R;
    }
    else if ( R is UNION OR R is INTERSECTION )
    {
        Loop through each Concept in R
        {
            C = next ( R );
            PartialResultSet = PartialResultSet ∪ C;
        }
    }
    return PartialResultSet;
}
```

In the following, we briefly explain the algorithm in Listing for establishing the semantic scope of semantic type from the profile within a annotated resource repository: firstly, if a primitive concept is selected, then it is simply added to the established semantic scope denoted by ResultSet. If a property is selected instead, then the task is to find the Domain (domainConcept) and Range (rangeConcept) of the selected property and perform this algorithm for each of them. In the final step of the algorithm, the ResultSet that is obtained is essentially a set of concepts from the profile. This set is complemented with all concepts that have been defined to be equivalent (using OWL: sameAs property) to any concept present in the original result set. This final step (potentially) widens the semantic scope of the results by including the synonym concepts.

Once the set of concepts resulting from the application of above mentioned steps is obtained, the annotation index is queried to retrieve the list of annotation objects that exist for each of the concept included the result set. Finally, the information in each annotation object is used to dynamically generate an HTML-document that provides a list of documents represented by the list of annotations (see Fig. 8(b)).

8. Analyzing contextualization capability

OntoMOVE’s performance can be evaluated along two fronts. Firstly, the sub-ontology extraction mechanism, which underlies the OntoMOVE framework, can be evaluated independently from its proposed application in the form of profiling and contextualization for the medical information retrieval domain. Here, the focus is on OntoMOVE’s capability to generate sub-ontologies that are semantically complete, optimal and independent of a base ontology. Secondly, performance of OntoMOVE’s proposed application toward semantic requirement profiling and Contextualization in the medical information retrieval systems domain may be evaluated. Here, OntoMOVE’s capability to contextualize or restrict an information retrieval request to only those parts of the overall repository that are semantically related to the user’s area of interest is highlighted. Considering the medical domain-specific scope of this paper, we restrict ourselves to the evaluation along the latter application front.7

8.1. Scope of analyses

We restrict the evaluation to the Contextualization capability of the system, as opposed to the empirical investigation of the precision and recall abilities of the information retrieval phase. Here, we are primarily interested in ensuring the consistency of two-way Contextualization—specialising a context or further expanding it by the refinement of the requirements specification. The manner in which such a refined, i.e., expanded or further specialized, con-
text is exploited by a information retrieval module, and its resulting precision and recall abilities, is independent of the Contextualization module per se. Hence, here we focus on the Contextualization precision and recall abilities, is independent of the Contextualization text is exploited by a information retrieval module, and its resulting Annotation metrics.

<table>
<thead>
<tr>
<th>Statistic</th>
<th>(a) Annotation and concept coverage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total concepts</td>
<td>UMLS-SN</td>
<td>135</td>
</tr>
<tr>
<td>Total annotations</td>
<td></td>
<td>1156</td>
</tr>
<tr>
<td>Concept coverage</td>
<td></td>
<td>95/135</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Annotations per concept</th>
<th>Annotations per document</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max</td>
<td>226</td>
<td>43</td>
</tr>
<tr>
<td>Mean</td>
<td>8.56</td>
<td>6.8</td>
</tr>
<tr>
<td>Standard deviation</td>
<td>23.41</td>
<td>6.67</td>
</tr>
</tbody>
</table>

Table 3

8.2. Base setup

For evaluation purposes, we use the setup illustrated in Table 3. The base medical domain ontology (i.e., UMLS-SN; see Section 5.2) being used consists of a total of 135 concepts. Using these concepts, a total of 170 documents from the overall Medical Therapeutic Guidelines have been annotated, resulting in a total of 1156 annotations. Note that from the overall 135 concepts in the base ontology, a majority of 70% have corresponding annotations in the document-set, with the remaining concepts being those that did not have any semantically related resources in the medical TG document-set. The absence of corresponding resources to annotate results from the fact that the (test) TG document-set chosen for the analysis is a small fraction (approximately 10%) of the complete TG collection.

8.3. Annotation quality

From the viewpoint of a conventional information-retrieval centric analyses, which is not the objective here, a limitation of our prototype is that our annotations are not performed by a domain-expert. As such, the annotation process also lacks a global annotation strategy. A general measure of the optimality or correctness of a particular distribution is not quantifiable, that being governed by the nature of the document-set being annotated. As a guideline toward preventing over/under representation of some medical resources within the resource repository, we adopt the heuristic that the annotations be (approximately) normally distributed throughout the document-set. However, note that this is simply a work-in-progress heuristic and will be no more applicable once a domain expert is involved (see Section 9). The heuristic also ensures that every document in the test-set is annotated in order to provide coverage to every identifiable semantic content present in it. Given the availability of annotation statistics, such as in Fig. 10(a), this, or potentially other annotation heuristics, are easily achievable within the framework.

Table 3(b) consists of basic statistical data relevant to the annotation process. Fig. 10(a) reflects the overall frequency distributions for the number of annotations per document that resulted from the annotation process for the document-set under consideration. Both, the concept and annotation coverage data as well as associated statistics are generated within the system for the singular reason that they are reflective of the qualitative aspects of the annotation process. This functionality is useful for fine-tuning the annotation strategy, which, with or without a domain expert, is generally considered to be a key component in the quality and improved accuracy of the information retrieval [13,36].

8.4. Contextualization capability

Toward the main evaluation task, we utilize a base requirement specification that approximately spans the entire UMLS-SN ontology. Based on this requirement specification, a complete and independent sub-ontology (referred to as ontology A) is created. The same process is repeated 3 times, each time using the sub-ontology from the preceding stage as the base ontology for the next stage. The result is a collection of four sub-ontologies (referred to as A, B, C and D) or user profiles that share the following relationship: \([A \supset B \supset C \supset D]\). The rationale behind this approach is that since the semantic scope (see Section 6) of the overall base ontology spans the entire test document-set, further specialization of the base ontology should result in a corresponding contraction of the semantic scope within the document-set. The results in Fig. 10(b) illustrate that this is indeed the resulting behaviour, i.e., the semantic scope of profile A, which consists of 107 concepts, spans 168 documents, whereas with profile D, the scope is narrowed down to 113 documents. Note that although multiple, non-related profiles can be applied toward a certain Contextualization task, we use the afore-discussed process of incremental specialization with one profile in order to make the results comparable since multiple profiles without non-overlapping requirements are not comparable.

Once the semantic-scope for a profile within the overall document-set is derived, the resulting scope essentially provides a means for the retrieval of semantically valid information for the user. Table 4(a), consisting of results for individual search queries, reflects the effects of Contextualization of the user’s interest or requirement for sample queries. We highlight the results for 4 queries (called Q1, Q2, Q3 and Q4) and additionally, also include the average results for a total of 21 queries. For instance, the results for profile A in row one of Table 4(a) indicate that from a total of 168 documents, queries Q1 results in 67 relevant documents whereas for profile C, which is a specialized profile in relation to A, the result drops down to 55 documents. The overall trends for all the profiles and the respective queries are shown in Table 4(a). Similarly, the effects of Contextualization when multiple, disjoint profiles are applied are shown in Fig. 11. For ease of reference, the results obtained when comparing both related and non-related

<table>
<thead>
<tr>
<th>Profile</th>
<th>Concepts</th>
<th>Documents in context</th>
</tr>
</thead>
<tbody>
<tr>
<td>E</td>
<td>59</td>
<td>158</td>
</tr>
<tr>
<td>F</td>
<td>26</td>
<td>113</td>
</tr>
<tr>
<td>G</td>
<td>79</td>
<td>117</td>
</tr>
<tr>
<td>E ∪ G</td>
<td>138</td>
<td>169</td>
</tr>
</tbody>
</table>

Table 4

Effects of context specialization and expansion.
(i.e., disjoint) requirement profiles are illustrated together. Here, we compare three profiles, $E$, $F$, and $G$, which are related in the following way: $[E \supset F]$ and $[E \cap G = \emptyset]$. The objective of using the setup in Table 4(b) and Fig. 11 is to show that Contextualization can operate both ways, i.e., for further (incremental) specialization of interest (as exemplified with the setup in Table 3(a) and Fig. 10(b) involving profiles $A$, $B$, $C$, and $D$) as well as for the expansion of the semantic scope with the document collection. Note from the setup in Table 4(b) that profiles $E$ and $G$ are disjoint and that $[|E \cup G| = |E| + |G|]^{10}$ holds, this relationship being important for our present exemplary purposes. The following should be noted from Fig. 11, which illustrates the results for 14 test queries: (a) Where a profile is a specialization of another (e.g., $F$ is a subset of $E$), the resulting documents with the specialized profile are consistently lesser than those for the general profile across all queries (this was illustrate in Fig. 10(b) as well). (b) When independent profiles are applied in conjunction (e.g., $E \cup G$), the results are consistently greater than those for any of the individual profiles applied in isolation across all queries. Note that the total number of results for the combined profiles need not be a sum-total of the results for the individual profiles since a document may consist of annotations of concepts from both the profiles (this is exemplified in Fig. 6 with overlapping ovals). Obviously, as mentioned previously, disjoint profiles (e.g., $E$ and $G$) are not comparable in isolation. As such, the respective data for profiles $E$ and $G$ in Fig. 11 are not relevant.

9. Summary and outlook

Although our methodology of sub-ontology based semantic requirement profiling and information retrieval is generally applicable in any domain where it is possible to annotate resource-sets, as an exemplar, we have designed OntoMOVE so as to be specifically applied in the medical information systems domain. OntoMOVE utilizes semantic web [5] standards like RDF(S), and the Web Ontology Language (OWL) in addition to medical domain standards and controlled vocabularies encompassed by the UMLS Knowledge Sources (UMLSKS), namely the UMLS Metathesaurus and the UMLS Semantic Network. The system offers advanced profiling of a users field of specialization and/or interest by exploiting materialised sub-ontologies produced by MOVE. Our methodology consists of making the semantic content present in the medical information sources explicit and building a system that takes advantage of the explicitly represented knowledge. At the core of our system lies
9.1. Expert-driven annotation and empirical study

The current focus of our work in this paper is to utilize a subontology profiling technique to perform contextual search and to filter relevant documents based on the user's profile as defined by the sub-ontology. As such, our goal is to prune document corpus based on our sub-ontology Contextualization approach. It is expected that this process will be further refined in the future, whereby the relevance of the document collections are then ranked based on their relevance to the user profile. Because our focus here is to demonstrate a novel approach of document filtering based on a sub-ontology extraction technique, our evaluation technique is currently experimental-based, and our future aim will be to perform the evaluation on real domain experts in the area. Indeed, this will also be necessary to perform a realistic empirical study from an information retrieval viewpoint since the quality and/or correctness of information retrieval and their benchmarking in terms precision and recall is not possible without domain-expert driven annotation strategy [13,36] and result ranking.

9.2. Annotation automation potentiality

As further work, we also regard it important to at least semi-automate the annotation process by utilizing the deep annotation technique that is suited to scientific (Medical or Bioinformatic) ontologies [20]. Toward this end, dynamic interaction via web-services with a corpus of Medical domain terms, the UMLS Knowledge sources and the Medical information sources (in a semi-structured form) seems essential. Work is in progress to develop built-in functionality, similar to that utilized via OntoMat, to create manual annotations using our custom annotation vocabulary. Although essentially similar to OntoMat in methodology, this built-in functionality will differ in two regards: (a) instead of an RDF based annotation schema, we develop our custom OWL ontology based schema, (b) the annotation schema itself will not be static, as is the case with OntoMat, thereby allowing users to specify their own annotation types. Most importantly, these extensions also facilitate the provision of the entire information retrieval methodology in one single application.

9.3. Ontological reasoning capability

Other interesting extension involves exploiting the ontological reasoning facilities that are available for OWL described resources using existing tools description logic reasoners such as Racer [42]. Finally, the Contextualization in the present system is performed in a built-in sequential manner. We are working toward utilizing the distributed version (see [8]) of our framework that has been designed for use in a distributed cluster environment. We propose to achieve this goal via the medium of web-services, i.e., a web-service performs as a mediator between the distributed framework operating in a Linux cluster environment and the Java-based OntoMOVE application.

9.4. Integration within an information retrieval system

The present work categorically focused on semantic requirement profiling, and resource Contextualization and its empirical investigation. We regard that any specialized ontology and annotation based information retrieval system will benefit from our Contextualization approach. For instance, as illustrated in the Contextualization analyses in Section 8, incremental specialization and expansion of the context is consistently achievable by the application of our profile derivation approach. These may in turn be utilized together or in isolation by an information retrieval system to fine-tune/control its search space and improve the quality of its results from a semantic viewpoint. In this context, an important functionality that need development is the automation of the requirements specification (Section 5.1) stage so that any arbitrary application may communicate its needs automatically to our system. Once this is achieved, in conjunction with the aforementioned task of involving expert-driven quality annotation, a conventional precision and recall study of the entire Contextualization-backed information retrieval phased would be feasible, and beneficial.

Acknowledgments

We acknowledge the financial and infrastructural support of the Victorian Partnership for Advanced Computing (VPAC, Australia) for conducting this research. VPAC, Australia: http://www.vpac.org/. M.B. also acknowledges funding provided by the Alexander von Humboldt Foundation, Germany. W.R. acknowledges the support from the Australian National University AIGRP collaborative research.

References
