# Semantic based recipe suggestion

Luigi Iannone, Ian Blacoe, Valentina Tamma, Floriana Grasso
{L.Iannone,I.W.Blacoe,V.A.M.Tamma,F.Grasso}@csc.liv.ac.uk

In this abstract we briefly sketch out the benefits gained and the challenges we faced in using semantic web technology to support the generation of recipe suggestions for patients with specific nutritional needs, as well as for normal people wishing to follow healthy food guidelines. This system is part of the PIPS EU project (FP6/IST eHealth Integrated Project No. 507019 – http://www.pips.eu.org).

Suggesting a recipe or, at least, determining whether it suits the patient's nutritional needs and does not conflict with their specific clinical situation, raises a variety of problems that can be categorized as follows:

- Knowledge acquisition: Where and how can we extract the necessary knowledge? How to tackle the heterogeneity of these sources?
- Knowledge Representation: Is there an adequate representation framework for describing the knowledge we need?

The required knowledge is heterogeneous along different dimensions: source, domain, permanence, etc.. In order to describe a recipe we need to individuate ingredients and their quantities as well as cooking instructions, and then, in order to assess its suitability for a given individual, we need to be aware of the individual's clinical situation, such as the drugs they are currently taking. Interaction among such drugs, the ingredients, and possible allergies must be taken into account. In addition, nutritional guidelines represent another different kind of knowledge that takes a part in the assessment of a recipe. All these types of information potentially originate from different areas both physically and logically, and may not all be in state in which they can be readily used. In fact, whilst on one hand there are ongoing efforts to formally express medical knowledge (allergies, drugs and interactions), on the other hand, recipes with their ingredients and cooking instructions are currently less permanent and, in general, are not formalized. Therefore, in order to acquire the latter kind of knowledge we devised a sub-system for scraping web resources and annotating them semantically (i.e.: in a formal language).

This system works as multi-layered analysis, annotation and reasoning engine which applies different sets of rules in consecutive phases to each resource in order to generate relevant metadata. The first layer aims to remove all the layout information from a document, reducing it to an abstract text based representation of the resource we want to annotate (in our case a recipe, with its ingredients, cooking instructions and serving advice). This is accomplished by means of a set of rules that syntactically analyse web resources. At this stage the abstract, though text based and, therefore still informal, document undergoes a further annotation process layer. This layer uses rules to identify instance occurences within the resource and outputs formal annotations. In the case of recipes, ingredients are related to the proper entities in the knowledge-base formally describing the nutritional domain. Once the formal annotations have been created, a further layer examines them with reference to a set of rules in order to exclude misidentified instances (filter phase). Finally, the filtered annotations are re-examined and reasoning over the ontology is applied so as to assert properties of instances that are implied by the ontology, but not directly obtained from the resource itself. Using Semantic Web (SW) technologies as the Knowledge Representation framework allows for decoupling the business knowledge of our application from the reasoning underlying it. With a traditional approach, business logic and control mechanism are tangled together causing poor maintainability and preventing reuse. We based our annotations on the OWL-DL ontology language, whereas the medical knowledge, nutritional guidelines, and their interactions are based on both OWL-DL ontologies and SWRL rule-bases in order to overcome expressivity limitations of Description Logics. Besides maintainability, using formal languages to describe the domain enables automatic derivation of implicit knowledge from the facts specified in the domain description, as well as within the knowledge base. The fact, for instance, that a recipe is not suitable for an individual needs not to be explicitly expressed within a knowledge base, but can come out from the fact that the individual is allergic to some ingredients in the recipe, or that there are some undesirable interactions between some ingredients and some drugs they are currently taking.

We adopted the SPARQL query language, as it is currently the only standard for querying SW knowledge-bases. Further, it can be easily be combined with reasoning, as there are robust frameworks (e.g.: HP Jena Semantic Web Toolkit) that allow to query both explicit and implicit facts within a SW knowledge-base. Utilising SPARQL we developed a GUI that enables users to easily create semantic queries, based on the ontological knowledge, that returns matching instances and associated resources.