User-Adapted Image Descriptions from Annotated Knowledge Sources

Berardina De Carolis and *Fiorella de Rosis*

Intelligent Interfaces, Department of Computer Science,

University of Bari, Italy

{decarolis, derosis}@di.uniba.it

Abstract. We present the first results of a research aimed at generating user-adapted image descriptions from annotated knowledge sources. This system employes a User Model and several knowledge sources to select the image attributes to include in the description and the level of detail. Both 'individual' and 'comparative-descriptions' may be generated, by taking an appropriate 'reference' image according to the context and to an ontology of concepts in the domain to which the image refers; the comparison strategy is suited to the User background and to the interaction history. All data employed in the generation of these descriptions (the image, the discourse) are annotated by a XML-based language. Results obtained in the medical domain (radiology) are presented, and the advantage of annotating knowledge sources are discussed.

1 Introduction

The amount of heterogeneous information available on the Web is growing exponentially; this growth makes increasingly difficult to find, access, present and maintain information. From research about how to make these tasks easier, methods for making machine understandable multimedia web resources have emerged: these methods require associating semantics to information, through the use of metadata. The description of such metadata is typically based on a domain conceptualization and a definition of a domain-specific annotation language. An annotation can be loosely defined as “any object that is associated with another object by some relationship” (from the W3C Annotation Working Group). In particular, XML is a standard, proposed by the W3C, to create mark-up languages for a wide variety of application domains; developing such languages favours universal storage and interchange formats, re-use and share of resources for web distributed knowledge representation and programming [11].

Metadata annotation of web resources is essential for applying AI techniques for *searching* and *extracting* relevant information (by improving a semantic contextualized search), for *maintaining* web resources (by keeping them consistent, correct and

up-to-date). More recently, there is a tend to employ it, as well, for *automatic document generation* especially when user-adaptation has to be considered.

Introducing annotations in a NLG systems requires two main steps:

1. *defining annotations for knowledge sources* in the application domain and for the intermediate results of the generation process; whenever possible, already existing and shared annotation languages should be employed (especially as far as application domain data are concerned);
2. *revising the NLG algorithms* so as to enable every generation module to read annotated data and to produce annotated results.

As far as user adaptation is concerned, annotating resources increases the possibility of finding information of interest to a particular user and to denote which particular piece of information is relevant for a particular interaction context. Annotating the steps of the generation process (for instance, the discourse plan) enforces a distributed vision of the process and enables rendering the final output as a function of the device through which the user interacts. This vision become particularly attractive when the resource to be described and explained to the user is an image. There are millions of images on the web that could be accessed for different uses and purposes, understanding their semantics would give the possibility of using them in several ways: for instance, for searching an image, for extracting useful information related to it, for creating image ontologies, for describing them, or relevant portions of them, verbally or textually, and so on.

In this paper, we will focus on this last aspect and in particular on the generation of user-adapted image descriptions in web-based consultation systems eventually accessible using different devices.

For this purpose, we need: i) to "understand" images, ii) to organize them into appropriate ontologies, iii) to define a user modelling component that formalizes the user features that are relevant for adapting the description, and iv) to generate the description more appropriate to the user and to the interaction context. This adaptation process may be seen as follows: given an knowledge base of images together with the related metadata describing them, a user model containing information about the user level of knowledge in the application domain, a list of already seen images during the interaction and the interaction context:

- select the attributes to include in the description and the level of detail of their description;
- select the appropriate description strategy" (an image can be described individually or by comparison with an image in the ontology that is known to the User);
- define the appropriate way to present the relevant information according to the context (i.e. web vs. wap);

To test our approach, we choose the medical domain in which image-based examples are very common to describes normal anatomical sites as well as particular pathologies. In particular, in order to show example of how it works in real domain application, we choose the context of hypertext for consulting medical guidelines ARIANNA (for more details see [3]). ARIANNA is a system aimed at dynamically generating user adapted hypermedia presentations of medical guidelines. Our medi-

cal partners envisage using this system to instruct students and to spread guidelines among general practitioners and specialists. In addition to the guideline, the prototype we developed is able to dynamically generate user adapted explanations of concepts involved in the clinical decision process: in this context, the User may ask to see some example about the explained concept, to better understand it; this example may be described either individually or by comparison with other cases, that the User is presumed to already know. As we work in the radiological domain, most of the examples to show are illustrated by images; therefore, our goal is to automatically generate context-dependent image descriptions, and we need "understanding" images to this purpose. The potential users may be classified as i) *students*, who may learn diagnostic and therapeutic procedures to follow in specific situations; ii) *doctors* with several degrees of competence, who may apply correct diagnostic and therapeutic procedures; iii) *patients*, who may get information about the scope and the efficacy of the health treatment they have to undergo.

In this context we used an image annotation tool to generate an XML structure correspondent to the metadata associated to it. For this purpose, we defined a XML-based mark-up language for radiological images and we developed an algorithm for interpreting its semantics. Starting from a set of annotated images, the information contained in the user model and given a communicative goal that formalises the User request of seeing an image example, a discourse plan is produced. This plan is built by taking into account the user's information needs and her background knowledge, and specifies the information content and the structure of the description text [3,10]: it is written also as an XML-structure, according to a mark-up language that we defined for this purpose. The annotated plan is the input of the surface generator that, according to the interaction context and to the User characteristics, decides how to render it.

In the following Sections, we will describe our approach by focusing, in particular, on how we use the annotation in the NLG process and by discussing the impact that an XML-based annotation may have on this process.

2 Generation of Image Descriptions

The explanation facility of ARIANNA uses two main strategies to generate the concept description that is appropriate in a given context: the concept position in a taxonomy of medical concepts and its relation with "similar" concepts that the User knows. If the User does not know other "similar" concepts, the generated text provides a complete description of the concept itself, in which its position in the taxonomy is specified by describing the relations with its ancestors. If, on the contrary, the User knows other concepts in the taxonomy (for instance because she has just seen their description), an explanation by comparison with the most similar of them is provided. To select the reference concept, a 'degree of similarity' between concepts is measured, by considering the attributes they have in common; the comparison then includes in the description the 'commonalities' and of the 'alignable' and 'non alignable' differences [9]. Only properties appropriate to the User level of knowledge

are mentioned in the text: commonalities are presented first, alignable differences second and non-alignable differences at the end. This strategy corresponds to what we consider a systematic description of concepts, which is typical of learning tasks, as opposed to information-seeking ones [7]

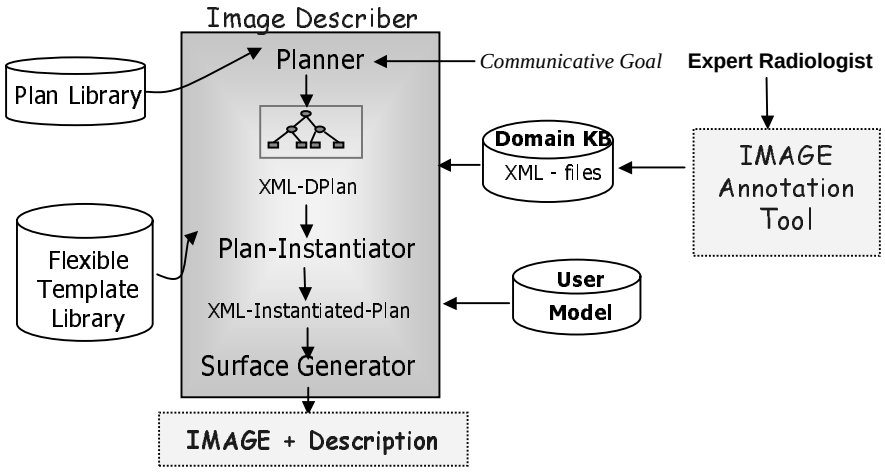

Fig. 1. The Architecture of Image Describer

As we mentioned in the Introduction, in the context of these explanations, the User may ask to see an example about the explained concept; these examples are in the form of radiological images, that have to be illustrated through some natural language text. In our first prototype of ARIANNA, image descriptions were pre-stored comments; this required our radiologists to provide a text for every example image and did not allow us to tailor it to the context. We therefore thought about applying, to produce image descriptions, strategies similar to those we applied in the case of concept explanations, so as to generate automatically texts by also taking into account adaptivity to the User knowledge. However, this goal required that our generator be able to "understand" images: let's see how we did it.

3. Understanding the Image

Understanding a image means extracting the features that characterize the information needed for its description: typically, these features are regions with their shape, texture, edges and so on. Since we do not use automatic image recognition techniques to extract these features, we use metadata to describe the image components, their attributes and the relationships among them. To build these metadata, we use an annotation tool (Inote [8]) in Java that is available on line and provides a way of annotating images with a XML-based mark-up language. Inote allows the User to attach textual annotations to an image and to store them in a text file as XML data,

through a XML structure that organizes them appropriately. With this tool, our medical partners can mark-up a digital radiological image by directly "writing on it" and without altering it; once a image has been loaded, the borders of one or more regions in the image may be outlined interactively, and a number of attributes may be associated with each of them. Regions are called “details” and attributes “annotations”, and may be given a name; a text may be associated with every annotation of every detail, by filling a text field. The details may be organized into as many “overlays” as needed. Inote’s mark-up language is very general, and may be applied to every kind of image. To tailor it to radiological images, we defined an ad hoc markup language that allows us to identify overlays and details in our images, with their attributes, in a univoque and unambiguously interpretable way. A radiological image has some “General Properties” that identify it: the technique with which the image was produced, the body region on which the exam was performed and the diagnosis. Its main information content then consists in a list of details that correspond to the regions of interest (anatomic structures); a set of characteristics (morphology, density, etc.) is associated with each of them.

```
<overlay>
    <title>parenchymal organs</title>
    <detail>
        <title>liver</title>
        <annotation>
            <title>position</title>
            <text>left</text>
        </annotation>
        <annotation>
            <title>morphology</title>
            <text>ellipsoidal</text>
        </annotation>
        <annotation>
            <title>volume</title>
            <text>normal</text>
        </annotation>
        <annotation>
            <title>margins</title>
            <text>regular</text>
        </annotation>
            </detail>
</overlay>
```

Fig. 2. An example of XML structure produced by Inote.

The first overlay in the Inote file then defines the “General Properties”; it is followed by other overlays, representing groups of visible details.

For instance, in the CT of abdominal organs, the following overlays may be defined:

- parenchymal organs
- hollow organs
- vascular structures
- muscular structures
- skeletal structures

The overlay named ‘parenchymal organs’ includes, as details, the organs in the image that belong to this category: the liver, the spleen and the lung parenchyma.

For each organ or detail, the following attributes may be specified: position in the image, relation with other parts, morphology, volume, density and margins. Each of them corresponds to an annotation. The example in Fig. 2 is a portion of the XML structure that was produced by Inote for a CT-scan (Computerised Tomography) of the abdomen. Figure 3 shows how this information was introduced, with Inote’s graphical interface. In particular, the expert radiologist after the graphical marking of the liver, is entering the annotation for the ‘morphology’ attribute.

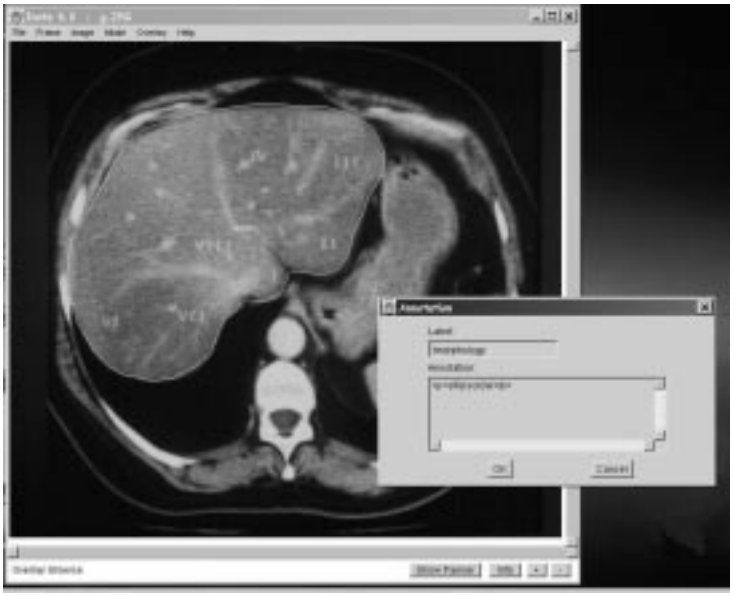

Fig.3: An example of a CT-scan annotation with Inote.

4 Planning the Image Description

The XML structure produced by Inote represents the knowledge base for our description generator. Before generating texts, our XML-application has to interpret the Inote tags and the detail and the overlay to which every annotation belongs, so that sentences describing the image can be built correctly. The generation decides the text structure according to the discourse plan that corresponds to given communicative goal: for instance, "Describe (System User I)", where I denotes a specific image in the domain KB. According to this goal and to the User characteristics, a presentation plan is selected from a library of non-instantiated plans that are represented as XML structures too; the generic plan is, then, instantiated by filling the slots of its leaves with available data in XML-domain-files. The DTD definition of our Discourse Plan Markup Language is shown in Fig.4. In this specification, a discourse plan is identified by its name; its main components are the nodes, identified by a name, containing mandatory attributes describing the communicative goal and the rhetorical elements (role in the RR of its father and rhetorical relation) attached to it. Then the 'info' element, that is not mandatory, describes additional information, related to a node, concerning the focus of the discourse and the complexity of the sub-tree de-

parting from it. These optional information elements are not used in this particular application, but they are necessary in other NLG systems developed by our research group [4, 5]. The XML-based annotation of the discourse plan is driven by two reasons, the first is that in this way it is possible to build a library of standard explanation plan that can be instantiated when needed and used by several applications working in several contexts; the second one is that we have chosen to use XML has a standard interface between all the modules constituting our generators, favouring in this way the distribution of resources and computation.

DPML 1.0 – Discourse Plan Markup Language

```
<!DOCTYPE d-plan[
<!ATTLIST d-plan name CDATA #REQUIRED>
<!ELEMENT node (node*, info*)>
 <!ATTLIST node name CDATA #REQUIRED goal  CDATA #REQUIRED
                role (root|nucleus|sat) #REQUIRED RR CDATA #IMPLIED>
<!ELEMENT info EMPTY)>
<!ATTLIST info focus  CDATA #REQUIRED compl (H|M|L) #REQUIRED >
]>
```

Fig4. Discourse Plan Markup Language DTD.

A small portion of the XML-Instantiated-Plan that was produced for describing the C.T. scan of the abdomen in Figure 3 is shown in Fig. 5. In this case, the XML-annotated plan has been instantiated according to the information relative to 'img1.xml' (as it is possible to notice from the goal of the tree root 'Explain(image, img1.xml)').

```
<d-plan name="CT-abdomen.xml">
  <node name="n1" goal="Explain(Image, img1.xml)" role="root" RR="Sequence">
    <node name="n2" goal="Describe(General Features, image)" role="nucleus" RR="ElabGenSpec">
      <node name="n4" goal="Inform(diagnosis,normal liver)" role="nucleus" RR="null"/>
      <node name="n5" goal=" Describe(Exam, C.T.)" role="sat" RR="Joint">
          <node name="n6" goal="Inform(name, C.T. Abdomen)" role="nucleus" RR="null"/>
          <node name="n8" goal="Inform(level, spleen)" role="nucleus" RR="null"/>
      </node>
    </node>
    <node name="n3" goal="Describe(Specific Features, image)" role="nucleus" RR="OrdinalSequence">
      <node name="n9"goal="Describe(ComplexStructure-1, parenchymal_organ)" role="nucleus"
      RR="OrdinalSequence">
          <node name="n10" goal="Describe(detail,liver)" role="nucleus" RR="ElabGenSpec">
            <node name="n12" goal="Describe(attribute,liver)" role="sat" RR="Joint">
              <node name="n13" goal="Inform(position,left)" role="nucleus" RR="null"/>
              <node name="n16" goal="Inform(rel_position,medialpart_abdomen)" role="nucleus" RR="null"/>
              <node name="n17"goal="Inform(morphology,ellipsoidal)"role="nucleus" RR="null"/>
              <node name="n18" goal="Inform(volume,normal)" role="nucleus" RR="null"/>
              <node name="n19" goal="Inform(margins,regular)" role="nucleus" RR="null"/>
            </node>
              <node name="n11" goal="Inform(name,liver)" role="nucleus" RR="null"/>
          </node>
          ...
  </d-plan>
```

Fig. 5. An example of XML-Instantiated-Plan.

5 Rendering the Image Description

This functionality of our Image Describer is very simple; the XML-Instantiated-Plan is the input of a Surface Realizator that, using flexible templates, produces the image explanation as an HTML file. This process is mainly driven by the Rhetorical Relations (RR) between portions of the plan. The plan is explored in a depth-first way; for each node, a linguistic marker is placed between the text spans that derive from its children, according to the RR that links them. For instance, the following sentence: "Inside the parenchyma, tubular shaped, hyperdense and white images are visible (the superhepatic veins)." Is obtained from an template for the *ElabGenSpec* RR in which the satellite, corresponding to the application of the *Joint* template to the following attributes <position>, <shape>, <density> and <colour>, is followed by the nucleus stating the name of the object in focus that is put between brackets (the superhepatic veins, in this case). The decision of rendering this template in this way, is driven by common patterns we extracted from a corpus of explanation written by expert radiologists and, from the same corpus, we extracted also the generation rules for the templates corresponding to other RRs.

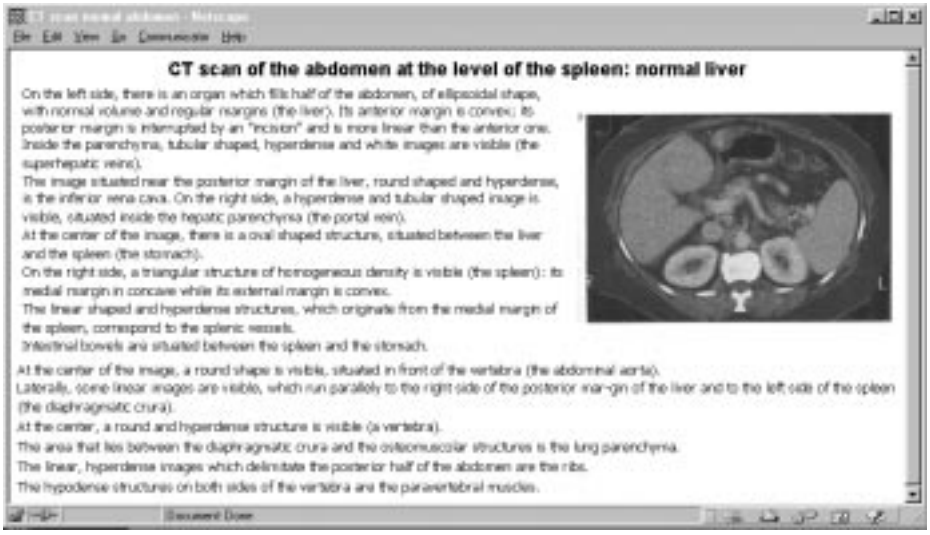

Fig 6. An example of image description.

At present, we generate the text in HTML; however, our approach is general enough to produce descriptions in different formats and, therefore, for different interaction contexts. It is also domain independent, since it is only driven by the Rhetorical structure of the discourse plan. We choose to develop our surface generator instead of using existing standard techniques, such as XSLT stylesheet templates, because these approaches did not allow us to produce complex textual description matching the style of the corpus that we analysed. This limit it is also underlined in

Cawsey and colleague papers [1,2], that used XML coupled with XSLT stylesheets for generating user tailored tabular presentations, from selected metadata, of online resources. In addition, in this system, as we will seen later on, we generates also description by comparison with other images, and this requires a more complex reasoning that is difficult to reduce to XSLT application to an XML file.Fig. 6 shows an example of the description that was generated from the discourse plan in Fig. 4.

6. Comparing Images

Let's now see how we generate the description of a image by comparing it with a reference image. The general strategy we apply is similar to the one we applied to compare concepts in ARIANNA. For every detail in a overlay, we mention first commonalities, second alignable differences and finally non-alignable differences. In the case of image descriptions, we distinguish, at the moment, three types of comparisons, that depend on what the User already knows and on the images she has already seen. Then, given a Image I to be described to a User U and a Reference-Image RI, three different comparison plans may be activated:

Comparison 1. KnowAbout(U, RI) AND Remember(U, RI) ⇒ Exec(S, cplan_1);
If the user, according to its background knowledge, profession and level of expertise or according to what she has already seen, knows RI and is presumed to remember its description, the first comparison plan (cplan_1) is applied. This plan corresponds to the following strategy: for each overlay and for each detail, only the attribute values of I that are different from the ones in RI are mentioned (alignable differences). After them, the values of the attributes that are not present in RI are presented (non-alignable differences). This plan is applied, for instance, to describe pathological cases to radiologists.

Comparison 2. KnowAbout(U,RI) AND ¬Remember(U,RI) ⇒ Exec(S, cplan_2);
If the user knows RI but does not remember it in all its details, the second comparison plan (cplan_2) is applied. This plan corresponds to the following strategy: for each overlay and for each detail, the attributes of I that take different values from those of RI are mentioned, by describing both values (for I and for RI). After them, also in this case, non-alignable differences are presented. This plan is applied, for instance, to general proactitioners.

Comparison 3. ¬KnowAbout(U,RI) ⇒ Exec(S, cplan_3);
If the user does not know RI, the third comparison plan (cplan_3) is applied. This plan corresponds to the following strategy: for each overlay and for each detail, all attributes in the two images are described, by emphasizing commonalities, alignable and not-alignable differences. This plan is applied, for instance, to students.

Let us see some examples of comparisons that were generated with our system: in all these examples, the reference image is a CT scan of the abdomen for a 'non-

pathological' case, while the image to be described is a case of hepatic cirrhosis, obtained with the same technique. The first text is generated by cplan_3: alignable differences are emphasized in italics, while there are no 'non alignable differences' between the two images; only the first part of the text is shown, for space reasons.

CT scan of the abdomen at the level of the spleen: hepatic cirrhosis.

As in the non-pathological case, the liver is the organ situated on the left side of the image, which fills half of the abdomen, of ellipsoidal shape. *In this case, however, its volume is reduced, its density is inhomogeneous and its margins, instead of being regular, are lobulated.* Like in the normal case, the anterior margin is convex while the posterior one is more linear and is interrupted by an incision. Superhepatic veins are visible inside the parenchyma; they are tubular shaped, hyperdense and white, like in the normal case.
As in the normal vein, the inferior vena cava is situated near the posterior margin of the liver, round shaped and hyperdense. The portal vein lies inside the hepatic parenchyma: it is hyperdense and tubular shaped like in the normal case, *but is enlarged.*
As in the normal case, the stomach is visible at the center of the image, between the–liver and the spleen, and is oval-shaped.
The spleen is visible on the right side; it is triangular and has a homogeneous density, like in the normal case, *but it is enlarged.* Also in this case, its medial margin is concave while the external one is convex.
The splenic vessels, which originate from the medial margin of the spleen, are linear and hyperdense, like in the normal case.
Also in this image, between the spleen and the stomach, intestinal bowels are visible.
……..

If c_plan2 is applied to the same case, the following text is obtained:

CT scan of the abdomen at the level of the spleen: hepatic cirrhosis.

If compared with a non-pathological case, the volume of the liver in this image is reduced, its density is inhomogeneous and its margins, instead of being regular, are lobulated. The portal vein is enlarged and the spleen is enlarged too.
……..

---

7 Conclusions and Future Work

In this paper, we presented the first prototype of Image Descriptor, a software to generate image descriptions from annotated knowledge sources: this prototype was built in Java using the IBM-XML4J parser and will be integrated in a system (ARIANNA) that dynamically generates hypermedia presentations of clinical guidelines; ARIANNA is already in use and an experimental evaluation study has been performed, to check how physicians react to it. The methods and the techniques we employed for generating image descriptions aim at favouring sharing and re-use of information. In particular, annotating images has several advantages: first of all, it enables retrieving images from Web databases according to ad hoc criteria; in addition, once a image has been retrieved, it may be described in a natural language text

whose content, structure, and style may be adapted to the context in which retrieval was made.

The annotation of linguistic resources favours, in general, their re-use and distribution: their semantics can be interpreted and rendered in different ways according to the interaction context; for instance, plain text, HTML or WML. Our research efforts go in this direction: we plan to introduce, in ARIANNA, a Conversational Agent with the role of an "Explainer" that supports the User at different levels; we already developed a similar Agent in another context, the generation of 'Animated User Manuals' for software applications [4]. In passing from hypertexts to Animated Agents, most of the techniques described in this paper will not change: for instance, the DTD for representing discourse plans is the same, and therefore also the planning component remains invaried; we only add a 'Sentence Planner' that revises the XML-plan files and substitute the surface text generator with a module that generates what we call the "Agent's behaviours".

We claim that, to enable sharing of resources and methods among various research centers and to produce outputs in context and application-dependent forms, establishing standards in the NLG field is a promising approach. This may foster reuse of methods in different applications and settings: let's think about new UMTS phones or wearable computers, whose particular graphical interface will require revising the generation methods that many of us developed so far. The work described in this paper is a step in this direction.

Acknowledgments

This work was founded by the CNR grant 21.15.01 on the topic: "Digital Processing of Radiological Images" and by the NationalCo-founded Project on "Intelligent Agents: Knowledge Acquisition and Interaction".

References

1. Cawsey, A. Presenting tailored resource descriptions: Will XSLT do the job? In Proceedings of the 9th International WWW Conference, 2000.
2. Cawsey A., Bental D., Bruce E. and McAndrew, P.: Generating resource descriptions from metadata to support relevance assessments in retrieval. Proceedings of RIAO 2000.
3. De Carolis, B., de Rosis, F., Andreoli, C., Cavallo, V. and De Cicco, M.L.: The dynamic Generation of Hypertext Presentations of Medical Guidelines. The New Review of Hypermedia and Multimedia, 67-88 (1998).
4. De Carolis, B., de Rosis, F., Pizzutilo, S.: Generating User-Adapted Hypermedia from Discourse Plans. Fifth Congress of the Italian Association of Artificial Intelligence (AI*IA 97), Roma , (1997).
5. B. De Carolis, C. Pelachaud, I. Poggi, Verbal and non verbal discourse planning. Workshop on Achieving Human-like Behaviors. Autonomous Agents 2000. ACM Press.
6. de Rosis, F., De Carolis, B., Pizzutilo, S.: Automated Generation of Agent's Behavior from Formal Models of Interaction. To appear in proceedings of AVI 2000, Palermo, Italy (2000).

7. Hammond, N. and Allinson, L.: Extending Hypertext for Learning: an Investigation of Access and Guidance Tools. People and Computers V, HCI 89, Cambridge University Press (1989).
8. Inote: Image Annotation Tool. http://jefferson.village.edu/iath/inote.html.
9. Markman., A.B. and Gentner., D.: Commmonalities and Differences in Similarity Comparisons. Memory and Cognition, 24, 2 (1996).
10. Moore, J., D. Participating in Explanatory Dialogues. Interpreting and Responding to Question in Context. ACL-MIT Press series in NLP, (1995).
11. W3C: eXtensible Markup Language (XML). http://www.w3.org/xml/
12. Marcu, D.: Extending a Formal and Computational Model of Rhetorical Structure Theory with Intentional Structures à la Grosz and Sidner. The 18th International Conference on Computational Linguistics COLING'2000, Luxembourg, July 31-August 4, 2000.