Medical Informatics Europe 2002
G Surján et al. (eds)
IOS Press 2002

Step–By–Step Mark–Up of Medical Guideline Documents

Vojtěch SVÁTEK and Marek RŮŽIČKA

European Centre for Medical Informatics, Statistics and Epidemiology – Cardio and Department of Information and Knowledge Engineering, University of Economics, Prague, W.Churchill Sq. 4, 130 67 Praha 3, Czech Republic

Abstract. The quality of document–centric formalisation of medical guidelines can be improved using a decomposition of the whole process into several explicit steps. We present a methodology and a software tool supporting the step–by–step formalisation process. The knowledge elements can be marked up in the text with increasing level of detail, rearranged into an XML knowledge base and exported into the operational representation. Semi–automated transitions can be specified by means of rules. The approach has been tested in a hypertension application.

1. Introduction

Medical guidelines are standard means for dissemination of medical knowledge; large attention is currently paid to their *formalisation* and *computational processing*. Most approaches to formalisation assume extensive interaction between domain expert and knowledge engineer, and are *model–centric*: the formal model of the guideline is piece–by–piece populated with knowledge mentally abstracted from the document. The model–centric approach has been repeatedly used for the development of guideline–based decision–support systems in projects such as EON, GLIF, Asgaard, Proforma, PRESTIGE or Prodigy [2]; we omit most citations for the sake of brevity, they can be found e.g. in [4]. An alternative stream in guideline computerisation is *document–centric*: the original text is systematically *marked–up* with respect to the model and kept as structured document. The leader in this stream is probably the GEM methodology and model [6]; its authors claim that the mark–up–based approach is more appropriate for capturing (in addition to the decision structures) the 'healthcare–service' aspects of the guideline, such as its prospective audience or support with clinical evidence. A generic advantage of mark–up–based formalisation is the possibility to structure the documents down gradually, in *multiple steps*. Existing projects leave such phasing upon the developer; nonetheless, making the stepwise character of the process *explicit* brings several benefits:

- Different types of expertise (medical, document design, knowledge modelling, target formalism) are required for delimited steps only; this saves the costly time of experts.
- The process is more transparent. Thanks to fewer transformations performed in each step, it is easier to point out the knowledge added by the expert explicitly. Also, the risk of information loss is reduced, and subsequent verification is made easier.

The second point is particularly important for the *compliance analysis* task: comparison of the actual medical practice (reflected in EPRs) with the standards set by the guidelines.

In this task, in contrast to the decision–support task, we prefer to preserve the generic content of the guidelines rather than to adapt it to local conditions; see [7] for discussion.

2. Methodology of the Step–By–Step Approach

The transition from a plain text document containing knowledge to an operational representation includes multiple aspects: *generic linguistic expressions* expressing e.g. the structure of definitions, decisions or causalities have to be replaced with *standardised formal structures*; free–text terms referring to the same *domain concept* have to be *unified*; knowledge elements have to be *modularised*, i.e. made independent of the surrounding context. There are several ways how to map these different aspects onto a sequence of steps: the one we propose here assumes five levels of formalisation:

Input Text Format. We assume that a natural choice for the initial text format is XHTML: the XML version of HTML. The creation of an XHTML document merely requires common web page design skills; the documents can be viewed with web browsers, and their elements can be referenced using the XLink/XPointer technology.

Coarse–Grained Semantic Mark–Up. Large (from sentence–level up) and relatively closed chunks of text are semantically marked–up, and parts of the document that are not likely to be exploited in the target application (often, results of clinical studies and ad hoc illustrations) are removed. We assume that the coarse–grained mark–up can be done even by persons without (deep) medical expertise.

Fine–Grained Semantic Mark–Up. The basic elements are refined into a tree structure of sub–elements. Although the original text should remain more–or–less untouched, reformulation is often needed in order to pick up relevant phrases consistently out of a complex sentence. Elements can be characterised according to the amount of *external knowledge* added[1]. Since we proceed from sentence level to term level, it is natural to create a *Data Dictionary* characterising the important clinical parameters involved e.g. in decision structures and concept definitions (the ultimate clean–up of the terminology is however left to the next phase). Background knowledge is added so as to resolve ambiguous statements and provide missing aspects of knowledge elements.

XML Knowledge Base. The original document structure is abandoned in favour of systematic ordering. The context of occurrence of knowledge elements has to be wrapped into their own structure to achieve modularity. Cross–references are verified and updated if necessary. Where possible, natural–languages phrases are replaced with XML structures containing Data Dictionary terms. Some of these activities can be done semi–automatically; others will require involvement of the medical expert.

Operational Code. The last step consists of two parts. First, the XML knowledge base is *syntactically* converted from the XML format to the format of the target operational language. This can often be done fully automatically, using the declarative apparatus of XSL style sheets. Second, some sub–elements of the XML code may still contain natural language text, which has to be interpreted and operationalised.

We have tested the approach using a simple guideline model derived from a semiformal ontology described in [8]. It has four top–level elements: *procedural* statements (later refined to *scenarios*), definitions of and references to *concepts*, *causal relationships*, and *goals* to be achieved; we assume that the four general types cover the majority of important statements that occur in guidelines for long–term care for a particular clinical condition. In its aiming at modular, self–contained scenarios, our model is actually similar to Prodigy

[1] We have, for example, used an XML attribute `added` (shared by all top–level elements) with values `no` (text without modification), `interp` (text has been reformulated using the most likely linguistic interpretation), `parts` (part of the text has been added) or `whole` (the whole element has been added).

[2]; interestingly, both have been designed with emphasis on primary–care guideline applications.

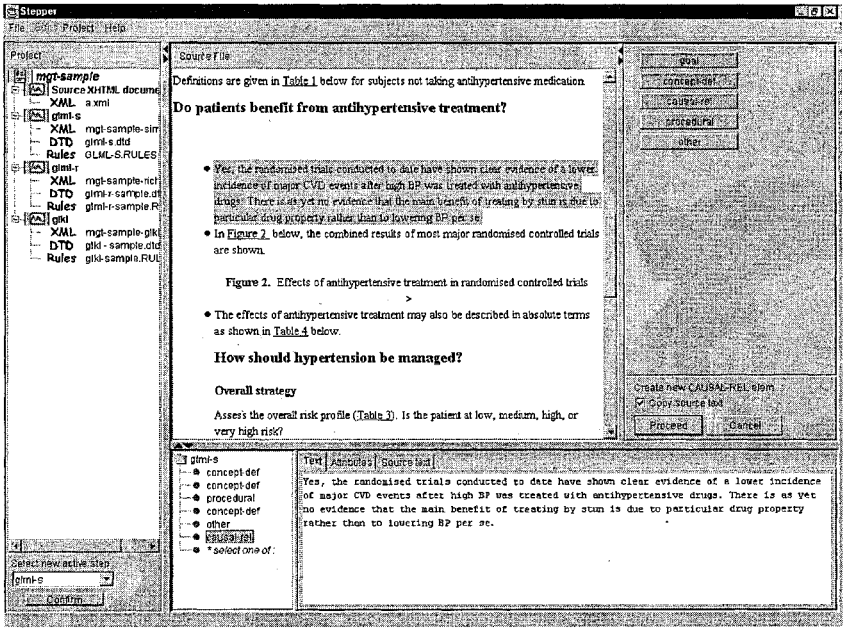

Figure 1: The mark–up interface of the *Stepper* system

The model has, in each of the levels of formalisation, a different shape – the elements evolve from free–text containers through thoroughly marked–up text into a knowledge base containing formalised items wrapped in XML, and, finally, into the computational representation (see [7] for details).

3. Tool Support

The beta version of the dedicated step–by–step mark–up editor has been developed (in Java) under the name of *Stepper*, with the following main functionalities:

- Support for the mark–up of knowledge elements *in a source text*, incl. specification of their *attribute values*.
- Fully automated generation and update of element–to–text and element–to–element *links* across the formalisation levels, and *retrieval* of knowledge elements arisen from the given text fragment and vice versa.
- Convenient creation and update of *transformation rules*, which enable to define operations such as element aggregation, decomposition, shift of element content into attribute, and even conditional setting of element value.

As soon as the rules have been defined, the users can carry out the mark–up (see Fig.1), fire the rules, and move information around the XML structures automatically built by the rules. The tree structures and buttons are generated in runtime from the DTD of the given formalisation level. In the latter transformation steps, the screen is divided horizontally into two parts corresponding to the 'source' and 'target' version of the document, each of them containing an XML tree and a pane for editing the attribute values.

Although we have tested the methodology and tool in connection with our own model, they can easily absorb another model via the DTDs. There is no hindrance, for example, to dividing the structure of the GEM model horizontally into the ‚coarse' and ‚fine' level, adding a ‚knowledge–base' level adopted to the particular application, creating the sets of transformation rules, and applying all of these on a particular guideline document.

4. Application

We have tested the methodology on the *WHO hypertension guidelines* [1], in the context of the European project 'Medical Guideline Technology', in 2000–2001. The document mark–up was a basis for the development of a compliance–analysis (and partially also decision–support) application [8]; the target language was OCML (Operational Conceptual Modelling Language, see [3]). Since the formalisation had to be carried out manually (the first version of the *Stepper* tool has been completed as late as in Autumn 2001), only the first three formalisation levels have been achieved completely, and the target application was thus based only indirectly on the semantic mark–up of the document.

Currently, in the EuroMISE Centre – Cardio (a new national–level research centre), we are both revisiting the hypertension application with the help of the *Stepper* tool, and starting to address another cardiolological application, namely, *unstable angina*. For hypertension, the formalisation of selected parts of the guideline document (setting the risk group of a patient) has been led through all the steps mentioned in section 2: the result is a simple interactive Java application *automatically generated* from the XML knowledge base. The methodology and the *Stepper* tool are judged intuitive by both medical and informatics staffs that are expected to use them.

5. Discussion

The *explicitation* of the stepwise character of medical text formalisation seems to be a unique feature of our approach. We will however try to line our research up with projects with similar objectives, focusing only on the most interesting points.

Shankar [5] attacks the problem of *rigid linking* between guideline text and elements of the abstracted model from the Information Retrieval perspective: instead of absolute addresses, the elements are associated with conceptual descriptions that enable to retrieve relevant portions of the document dynamically. Our solution to this problem consists in 'segmenting' each text–to–model link into multiple parts corresponding to transformation steps: if the document (or even the model) changes, only the adjacent part of the link has to be modified. We thus remain faithful to the document–centric paradigm while eliminating one of its drawbacks; conversely, Shankar's approach stands on the model–centric ground.

One of known advantages of document–centric approaches is the possibility to maintain *different parts of the same document* in different levels of formalisation, cf. [6]. This, in a sense, goes well together with our step–by–step view, since explicit formalisation levels can be more easily separated than ad–hoc (and thus implicit) formalisation levels. On the other hand, the *Stepper* tool does not (by its nature) support the mixing of different levels in the same document. Rather, the documents in later stages of formalisation may contain only some parts of the documents in earlier stages of formalisation.

An important aspect of free–text formalisation is linking to *terminological standards* such as ICD–10 or SNOMED. Although we have not implemented such linking in our prototype tool yet, we presume that it will be part of the fine–grained semantic tagging. For example, in our simple guideline model, the element for ‚concept definition' contains sub–

elements for ‚canonical' name and for ‚aliases', which can be used for distinguishing dictionary terms from ad–hoc ones.

6. Conclusions

In the paper, we have described the methodology, model and software tool for step–by–step transformation of knowledge–rich documents into a formal (or even operational) representation. The approach has been developed in the context of medical guideline formalisation, which still represents its principal application area. The explicitly defined formalisation levels guide the whole process, enable to reduce the involvement of the domain expert, and help to minimise the information loss.

Future work will, among other, address the capture of *consensual* background knowledge needed to operationalise vague statements and to fill gaps in knowledge (due to implicit knowledge assumptions); we assume the use of fuzzy measures for this purpose. Attention will also be paid to overcoming some technical limitations of the first version of the *Stepper* tool, and to the evaluation of other existing guideline models in the step–by–step framework.

The authors wish to express their thanks to Tomáš Kroupa who contributed to the development of mark–up languages, to the medical expert Jan Peleška, to Jana Zvárová, Director of the EuroMISE Centre, for her long–term support and inspiring comments on the project, and to Vilém Sklenák for assistance in typographical matters.

The research has been partially supported by the *project LN00B107* (European Centre for Medical Informatics, Statistics and Epidemiology – Cardio) of the Ministry of Education of the Czech Republic.

References

[1] WHO/ISH Guidelines for the Management of Hypertension. Journal of Hypertension, 17, 1999, 151–183.
[2] P. D. Johnson, S. Tu, N. Booth, B. Sugden and I. N. Purves: Using Scenarios in Chronic Disease Management Guidelines for Primary Care. AMIA Annual Symp., Los Angeles, CA, 389–393. 2000.
[3] E. Motta: Reusable Components for Knowledge Modelling: Principles and Case Studies in Parametric Design. IOS Press, Amsterdam, 1999.
[4] M. Peleg, A. A. Boxwala, O. Ogunyemi, Q. Zeng, S. W. Tu, E. Bernstam, L. Ohno–Machado, E. H. Shortliffe and R. A. Greenes: GLIF3: The Evolution of a Guideline Representation Format. AMIA Annual Symposium, Los Angeles, CA, (20 Suppl):645–649. 2000.
[5] R. D. Shankar, S. W. Tu, S. B. Martins, L. M. Fagan, M. K. Goldstein, and M. A. Musen. Integration of Textual Guideline Documents with Formal Guideline Knowledge Bases. In: AMIA 2001.
[6] R. N. Shiffman, B. T. Karras, A. Agrawal, R. Chen, L. Marenco and S. Nath: GEM: A proposal for a more comprehensive guideline document model using XML. JAMIA 2000; 7(5):488–498.
[7] V. Svátek, T. Kroupa and M. Růžička: Guide–X – a Step–by–step, Markup–Based Approach to Guideline Formalisation. In: (B. Heller , M. Loeffler, M. Musen and M. Stefanelli, eds.:) Computer–Based Support for Clinical Guidelines and Protocols, IOS Press, Amsterdam, 2001, 97–114.
[8] V. Svátek, A. Říha, T. Zíka, J. Zvárová, R. Jiroušek and Z. Zdrahal: Informal, Formal and Operational Modelling of Medical Guidelines. In: Hruška T., Hashimoto M. (eds.): Knowledge–Based Software Engineering. IOS Press (2000), 9–16.