Visual Coder: Clinical Coding in Translational Research

Fieran Mason-Blakley
University of Victoria
3800 Finnerty Road
Victoria, Canada
fmason@uvic.ca

Jens Jahnke-Weber
University of Victoria
3800 Finnerty Road
Victoria, Canada
jens@uvic.ca

Melanie Tory
University of Victoria
3800 Finnerty Road
Victoria, Canada
mtory@uvic.ca

Cliff McCollum
GenoLogics Life Sciences
4464 Markham Street
Victoria, Canada
cliff.mccollum@genologics.com

ABSTRACT

As the fields of translational research and personalized medicine evolve, the need clinicians and scientists have to exchange experimental results is growing. As different terminologies are used to describe research results in different fields, a tool which could provide assistance in the automated integration of scientific and medical experimental results would be facilitated by semantic tagging using terminology sourced from a domain spanning thesaurus[5]. We present Visual Coder, a visualization tool for performing clinical coding of anatomic pathology structured reports (APSR) for the purpose of multidisciplinary knowledge exchange. The tool provides a visualization of a translational research terminology server and a graphical user interface which provides a set of visual tools to facilitate clinical coding.

Categories and Subject Descriptors

D.2.12 [**Software Engineering**]: Interoperability

General Terms

Documentation, Design, Standardization

Keywords

Clinical Coding, Semantic Annotation, Translational Research, Knowledge Visualization

1. INTRODUCTION

The field of translational research informatics (TRI) provides informatics solutions for collaborating life sciences researchers and clinicians. One sub-domain attempts to provide mechanisms by which clinicians and scientists can automatically exchange results across disciplinary boundaries while at the same time maintaining the semantic integrity of those results. A primary hurdle in providing these solutions is the difference in terminology used by experts in different organizations or disciplines.

In order to support automated data exchange, the issue of semantic interoperability must be addressed. Clinical coding is an approach commonly used in medicine. This process involves annotating textual data with codes which refer to concepts that are stored in communal knowledge sources. There are many rich, purpose specific knowledge sources available for clinical coding. A few sources of note are the Systematic Nomenclature of Medicine Clinical Terms (SNOMED CT)[1], Logical Observation Identifiers Names and Codes (LOINC)[2] and the International Classification of Disease 10^{th} ed(ICD 10)[4]. The responsibility of performing semantic tagging would belong to a person with a different vocabulary in a clinical setting than it would in a laboratory setting; consequently, a tool which is meant to facilitate interdisciplinary knowledge exchange must support the vocabulary used by each potential user.

The Unified Medical Language System's Metathesaurus is the largest biomedical knowledge source of its kind[3]. It provides mappings between more than 100 biomedical ontologies, taxonomies and controlled terminologies including [1][2][4]. The term to concept associations provided by the Metathesaurus are precisely what is needed to perform semantic tagging in a translational research context.

Our primary contribution is Visual Coder, an extension of [8][6]. Visual Coder's terminology exploration features allow clinical coders to inspect the relationships between their native terminology and that used by other researchers by employing the UMLS Metathesaurus as a communal knowledge source while performing clinical coding of anatomic pathology structured reports (APSRs). By employing visualization we have allowed these experts to perform this task in a more exploratory way, accelerating the development of their understanding of foreign terminologies. By providing this understanding and by employing a domain spanning thesaurus we facilitate interdisciplinary knowledge exchange.

Presented in fig. 1 is a screen shot of Visual Coder. A user first opens an APSR for annotation. Visual Coder then parses the APSR into the tree structure labeled as A. The

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
Copyright 20XX ACM X-XXXXX-XX-X/XX/XX ...$10.00.

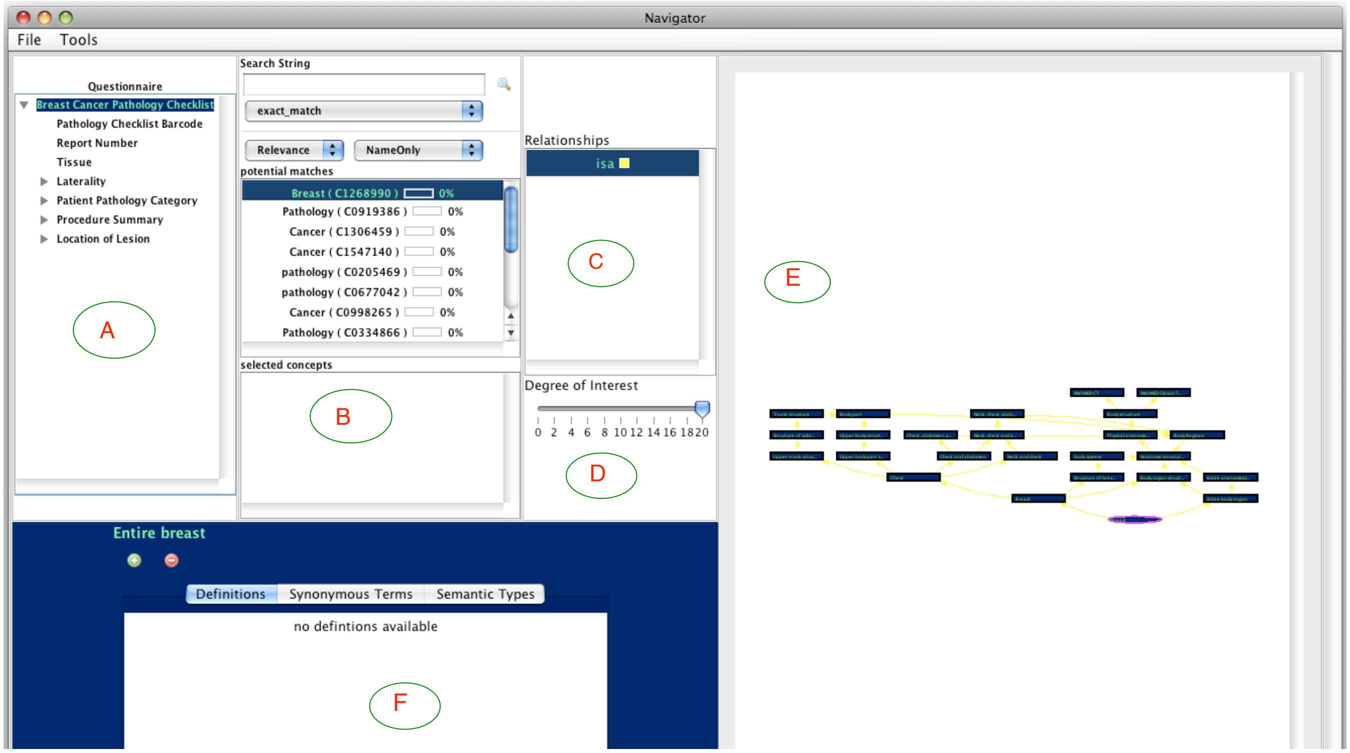

Figure 1: An overview of the application viewport

user then uses the search tools in B to uncover concepts which might potentially be used for annotation and then continues to explore the uncovered concepts using the visualization panel labeled as E with its corresponding controls labeled as C and D, and the definitions panel labeled as F. Annotated documents may be saved for future retrieval.

2. DOMAIN

A domain model was developed with the assistance of GenoLogics Life Sciences, our industry collaborators. During this process the players and entities of the target domain were identified. The domain model was further refined after our involvement with the IHE/HL7 APSR working group. The domain is described below.

Within a cancer clinic, a care team cares for a set of patients. During the course of treatment, tumor samples may be extracted from a patient. The care team develops an APSR which acts as part of a record of care for the patient and which also documents the properties of the tumor. A clinical coder who is a member of the care team may use a piece of software to perform clinical coding of the produced APSR. The data which is recorded in the APSR along with the annotations produced by the clinical coder are stored in an information repository. Primary investigators, e.g, academic biochemists performing life sciences research, are then allowed to access this information repository.

In our proposed solution, the clinical coder uses Visual Coder to perform the clinical coding of the APSRs produced in his clinic. The concepts he uses to annotate the APSRs during this task are fetched from a knowledge source through term based queries. The APSRs which the clinical coder annotates have a hierarchical structure. They can be decomposed into sections, fields and data. Annotations are stored in an information repository. The information repository may later be accessed to respond to primary investigator queries. These queries may be addressed by producing annotated standardized APSRs.

3. EVALUATION CRITERIA

We first demonstrate the efficacy of our visual clinical coding interface by displaying how it meets user requirements which were identified during an industrial collaboration with GenoLogics Life Sciences. We further evaluate this visualization component on a theoretical basis by employing the visualization theories of Ware[10] in the context of Shneiderman's visualization task list[7].

Visual Coder is evaluated in the context of the targeted usage scenario described below, by showing that the tasks related to the steps of this workflow are supported. Shneiderman's visualization task list[7] is assumed to be a reasonable representation of the visualization tasks which would be required to accomplish the steps presented in the workflow, especially when considering van Ham's[9] search, show context, expand on demand modification to Shneiderman's mantra of overview first, zoom and filter, then details on demand [7]. Below is an itemized overview of our hybrid approach.

1. open an APR template

2. inspect a section title or field title from the APR

3. use the terminology exploration tool to find the concept which was intended by this title

4. annotate the selected field with the intended concept

5. repeat this process until the document is sufficiently annotated to provide the desired degree of semantic interoperability with collaborating researchers

6. persist the annotations to an information repository

Task	**Description**
Search	Isolate a generalized subset of the terminology for refinement
Show Context	Show relationships between a core concept and its conceptual context
Filter	Hide uninteresting items
Expand on Demand	Allow the user to broaden over specified searches and also provides additional information about selections
Zoom	Focus on details which are pertinent to the task
Extract	Extract data subsets and query parameters
History	Rely on the system to persist actions previously taken for future reuse

Table 1: Hybrid Task List[7][9]

4. DEMONSTRATION

Initially a clinical coder opens an APSR which is represented in a local computable syntax. The APSR is displayed in A of the working area as shown in Fig 1. Visual Coder will automatically select the first APSR node (i.e. a section title, a data label, or a data value) for the user to annotate.

4.1 Search

Visual Coder then automatically searches the knowledge source for concepts which match this first selected node using pre-configured search strategies which are provided in B. It then automatically populates the list of potential matches list shown directly above B for the selected APSR node with this prefetched concept list. The first of these concepts is automatically selected for display in the visualization panel labeled E. The display shows a node link graph which places the selected concept as the root. The graph portrays the conceptual context of the selected concept.

The user can then use the contextual information provided by the visualization to determine the suitability of the suggested concept for the annotation of the selected APSR node. If the user is not satisfied with the first concept in the list of concepts proposed by the system for annotation, he/or she can either select the next concept in B, or even choose one of the other concepts which appeared in the visualization, E, as an annotation.

If the user exhausts all of the system's prefetched suggestions, then he/she can use the search strategies provided in B using his/her own input strings as opposed to the exact APSR node title which is used by default. This process again will populate the potential matches list with the concepts returned by the knowledge source in response to the user's queries.

4.2 Show Context

The relate task is fulfilled by the node link graph which is used to represent suggested concepts within their semantic context in E. Using nodes to represent concepts and arrows to represent relationships between them takes advantage of users' precognitive faculties to recognize association[10]. Curved arrows were used to represent the links between nodes as they are more more likely to be perceived as part of the same entity[10]. The various relationship types are categorized and aggregated using colour. The display uses only 8 colors so as not to overwhelm the pre-coginitive recognition of the user[10]. These colours are linked to the relationship filter in C by a legend of associated colour blocks. The colour legend shown in the screen shot displays a yellow square associated with the isa relationship type. This yellow square corresponds to the yellow arrows used in E to represent concept relationships. In more complex graphs however, a more extensive legend is shown which uses different colours for each relationship type. The inclusion of a brushing and linking feature has been considered as well. This feature would highlight elements in the display panel as the user passed the mouse cursor over the associated relationships in the visualization control panel.

4.3 Filter

Once a user has found a concept which shows promise as a potential APSR node annotation, he/she can take advantage of the filtering options provided in C and D to refine his or her search. Currently, a set of relationships are displayed to portray the conceptual ancestral relationships of the selected concept. The visualization control panel allows the user to show or hide concepts based on their associated relationship types. Currently, the graph does not include children of the root concept along the hierarchy paths. This is as a consequence of technical issues with the underlying relational model which have resulted in severe performance issues. In future work, it is expected that child, co-occurrence and other relationship sets will also be extracted from the knowledge source.

In addition to the relationship type filtering feature in C, a degree of interest filter, labeled D, is provided. With this feature a user can specify the desired volume of context they wish to view by limiting the concepts by their distance in edges from the graph's root concept.

4.4 Expand on Demand

The next point of focus would be the definitions panel, labeled F. As has been described, when a user selects a concept from the selected concepts list in B, the display, labeled E, is populated by the node link graph which represents the conceptual context of the selection. The selected concept is designated as the root of this diagram is selected by default.

This definitions panel, F, provides a more detailed description of the selected concept than is provided in the visualization panel, E. The display panel only provides a fixed length abbreviation of concept names with the full name being available through a mouse over tool tip(in addition to the context information of course). The details on demand panel provides the full concept name, concept definitions, synonymous terms to the concept title and semantic categorizations.

By default, the root concept of the node link graph is selected in the visualization. Consequently, the information published in the definitions panel describes this root concept. With the click of a mouse the user can select other nodes within the graph. This action will automatically repopulate the definitions panel with details about the newly selected concept.

Concepts which have been selected in the visualization panel can be added as annotations to the concept which has been selected in the APR tree by using the add annotation button in the definitions panel (the green plus icon). This action will move either move the selected concept from the potential matches list to the selected concepts list in B if the chosen concept is present there, or will simply add the concept to the selected concepts list if it is not. These annotations will be used in the final persistence step of the clinical coding workflow.

4.5 Zoom

The zoom task is supported by the navigation features available through the display panel in E. By using the mouse and various command keys, the user can pan across the displayed concept graph and can also zoom in and out of the display when they wish to see a particular corner of the graph in more detail. Additional features like a semantic zoom with which a user could change the root of their concept map based on a double clicking action will likely be considered in a future version.

4.6 Extract

At present the extract task is not supported; however, a plan to support extraction by employing the translation engine to produce standardized HL7 CDA compliant APSRs has already been developed and is currently being pursued. These transformations would be executed by performing a semantic mapping between an expertly annotated HL7 CDA compliant APSR to the annotation of the locally developed APSR.

4.7 History

The history task also is not supported but could be facilitated in future by the implementation of caching on the restful server, and also by using a more elaborate querying mechanism for prefetched annotation suggestions. Currently a variety of search routines are employed to provide the initial annotation suggestions. By monitoring user selections and the queries which uncovered them, a bias towards the better search strategies could be integrated into the application.

5. SUMMARY

A clinical coding tool was developed using a translational research knowledge source. The tool provides functionality to facilitate nearly all of the visualization tasks which have been specified by Shneiderman and Van Ham. We also propose potential mechanisms to fulfill the remaining relevant tasks.

A few technical challenges remain with this tool mostly revolving around the relational model used for the underlying database. Some usability issues remain as well, but these will be addressed based on user feedback. In future work, a user study through an industrial partnership would be a good way of uncovering weaknesses in both the provided interface and the underlying domain model.

We propose that the ideas presented in this tool can be exploited to provide more rapid, accurate and consistent coding of anatomical pathology structured reports. We assert that further involvement with industry partners like GenoLogics Life Sciences and with SDO partners like the IHE/HL7 Anatomic Pathology Structured Reports Working Group will yield strong and medically meaningful results.

6. REFERENCES

[1] Tim Benson. *Principles of Health Interoperability.* Springer, 2009.

[2] Clement and et al. Loinc, a universal standard for identifying laboratory observations: A 5-year update. *Clinical Chemistry*, 49:624–633, 2003.

[3] Theresia Gschwandtner, Katharina Kaiser, Patrick Martini, and Silvia Miksch. Easing semantically enriched information retrieval–an interactive semi-automatic annotation system for medical documents. *International Journal of Human-Computer Studies*, 68(6):370 – 385, 2010. Human-Computer Interaction for Medicine and Health care (HCI4MED): Towards making Information usable.

[4] Guoqian Jiang, Jyotishman Pathak, and Christopher G. Chute. Formalizing icd coding rules using formal concept analysis. *Journal of Biomedical Informatics*, 42(3):504 – 517, 2009. Auditing of Terminologies.

[5] J. Lahteenmaki, J. Leppanen, and H. Kaijanranta. Interoperability of personal health records. In *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pages 1726 –1729, 3-6 2009.

[6] X. Lin and D. Zhang. Visualization of knowledge structures. *Proceedings of the 11th international Conference information Visualization*, IV:476–484, July 2007.

[7] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. *Proceedings of the 1996 IEEE Symposium on Visual Languages*, pages 336–343, September 1996.

[8] E. Sundvall, M. NystrŽm, H. Petersson, and H. Ahlfeldt. Interactive visualization and navigation of complex terminology systems, emplified by snomed ct. *Studies on Health Technology and Informatics*, 124:851–856, 2006.

[9] Frank van Ham and Adam Perer. Search, show context, expand on demand: Supporting large graph exploration with degree-of-interest. *IEEE Transactions on Visualization and Computer Graphics*, 15:953–960, 2009.

[10] Colin Ware. *Information Visualization: Perception for Design.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.