Results Visualization in the XBrain XML Interface to a Relational Database

Zhengming Tang[2], Yana Kadiyska[2], Dan Suciu, PhD[2], James F. Brinkley, MD, PhD[1,2]

[1]Structural Informatics Group, Departments of Biological Structure and Medical Education and Biomedical Informatics,
[2]Computer Science and Engineering
University of Washington, Seattle WA 98195

Supported by Human Brain Project grant RO1 MH/DC02310.

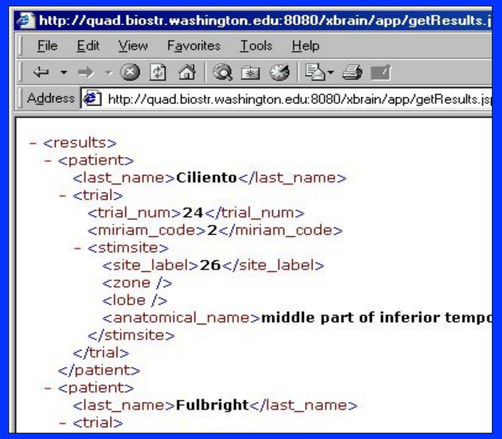

Figure 1: Original XML Result Output. No real patient names are in the database, only pseudonyms.

Introduction:

The University of Washington's XBrain application is used to dynamically export relational data over the web in XML format, as a prelude to data exchange. We describe additional tools to aid the human user in visualizing the dynamically generated XML results returned by the web application.

XBrain is a JSP web application that dynamically publishes data from a relational database containing language map data acquired during neurosurgery[1]. XBrain takes XQueries as input, hands it to a middleware application called SilkRoute, and returns XML results back to the user. Although XML is good for data exchange between applications, it is often not chosen for visualization of the data because it is not very human readable. Therefore, we focused on the users (neuroscientists) and developed more intuitive ways to visualize the XML output in XBrain.

Figure 1 shows sample XML output from a query that asks for all regions of the brain that showed an error of type semantic paraphasia (miriam_code = 2) during a language mapping study performed during neurosurgery. Figures 2-4 show these results reformulated in ways more intuitive to the users.

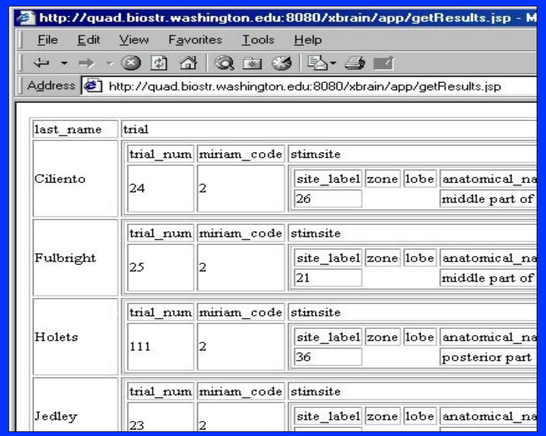

Figure 2: HTML Version

From initial interviews, we found that users wanted results in HTML and CSV (comma separated values, for import into Excel). We implemented these formats using XSLT, a well-known language for transforming XML into other formats. However, because the DTDs (Document Type Definition) of the XML results are not known beforehand, the XSLT must be generated dynamically, on a per user query basis. The generated XSLT is then used to transform the XML result into either HTML (Figure 2) or CSV (Figure 3).

The advantage of HTML and CSV is that they are fairly general: given any user query input, one can generate the XSLT transformations to convert the results into nested tables for HTML or a flat table for CSV. However, the same generality also revealed a major disadvantage of these formats: by assuming nothing about the results, the visualizations don't offer any interesting analysis of the data, leaving the analysis completely to the users.

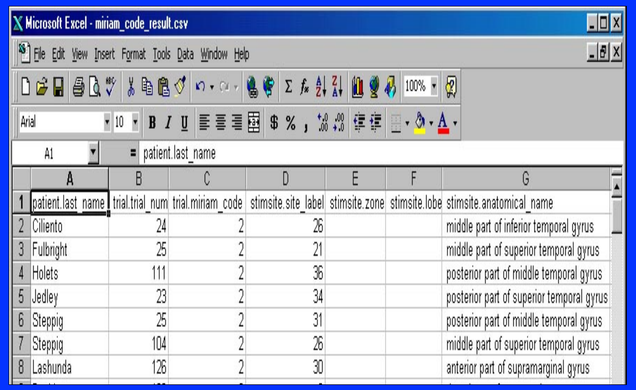

Figure 3: CSV version

In order to further make XBrain useful to the users, we focused on a common set of queries which find the anatomical regions in the brain where different language errors occur. Figures 4a & 4b display results from such a query using dynamically generated images. Users may then filter their query results and redraw the images based on the new constraints (Figure 5).

Conclusions:

We conducted informal user studies on a small number of researchers. All preferred the new visualization formats over XML. When users were asked to compare analyzing results in XBrain against their current manual methods using Excel and Photoshop, all preferred using XBrain. More importantly, users are interested in trends/patterns in the data and in being able to quickly visualize the results and add/update constraints. Thus, they can then explore various permutations and locate new and interesting trends using the new visualizations in XBrain.

[1] Tang Z, Kadiyska Y, Li H, Suciu D, Brinkley JF. Dynamic XML-based exchange of relational data: application to the Human Brain Project. In: Proceedings, AMIA Fall Symposium. Pp 649-653; 2003.

For more information, e-mail: < brinkley@u.washington.edu >

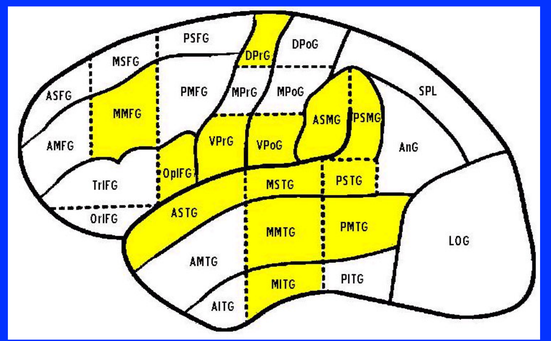

Figure 4a: Visualizations of the query result with yellow regions representing presence of region in the result XML.

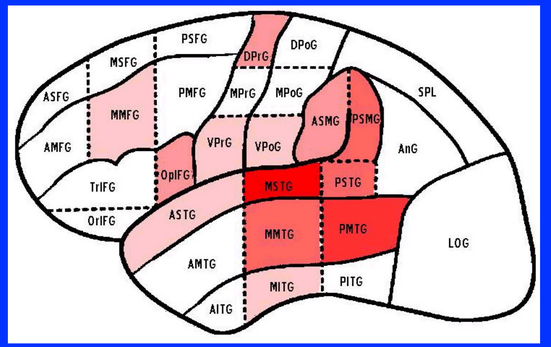

Figure 4b: Visualizations of the query result with highlighted region also representing the number of times the region occurs in the XML result.

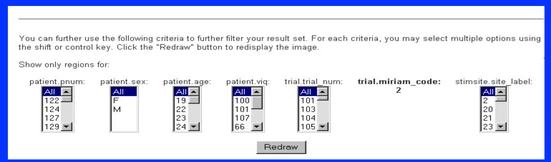

Figure 5: Additional constraints (auto-generated from results) to filter and redraw the images in Fig 4a & 4b.