

(Natural Language) Interaction with Graphical Representations of Statistical Data

Leo Ferres
HOTLab
Carleton University
Ottawa, ON, Canada
lferres@ccs.carleton.ca

Petro Verkhogliad
School of Computer Science
Carleton University
Ottawa, ON, Canada
vpetro@gmail.com

Louis Boucher
Dissemination Division
Statistics Canada
Ottawa, ON, Canada
Louis.Boucher@statcan.ca

ABSTRACT

Numerical information is often presented in graphs to take advantage of the human ability to quickly find visual patterns. Unfortunately, this medium is problematic for people who are blind or otherwise visually-impaired. To provide accessibility to graphs published in The Daily (Statistics Canada's main dissemination venue), we have developed *iGraph*, a system that provides short verbal descriptions of the information depicted in graphs and a way of also interacting with graphical information.

Categories and Subject Descriptors

H.5.2 [INFORMATION INTERFACES AND PRESENTATION]: Natural Language; I.2.1 [ARTIFICIAL INTELLIGENCE]: Applications and Expert Systems—*Natural language interfaces*; I.3.6 [COMPUTER GRAPHICS]: Interaction Techniques

Keywords

Accessibility (Blind and Visually Impaired), Natural language Interaction, Statistical Graphs

1. INTRODUCTION

The main objective of our research program is to make graphical information published in Statistics Canada's "The Daily" accessible to blind and visually-impaired populations and to allow querying and exploration of said information.

The work reported here builds heavily on two bodies of literature: (a) the literature on generating summaries of graphical data [2, 6, 4], among many others and (b) the literature on data exploration by sound (see [1], for example). The work reported in (a) comes mostly from the Natural Language Generation (NLG) community and aims at providing short descriptions of time-series and meteorological data, or the efficient integration of text and graphics. On the other hand, the literature in (b) provides tools to make graphs interactive by using time and sound. In the systems

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.

W4A2007 - Challenge, May 07–08, 2007, Banff, Canada. Co-Located with the 16th International World Wide Web Conference.

Copyright 2007 ACM 1-59593-590-8/06/0010 ...\$5.00.

proposed in (a) there is little or no interactivity, while those in (b) do not seize the richness of natural language. *iGraph-Lite* will hopefully contribute to the intersection of these two bodies of literature (see [3]): *iGraph-Lite* is an interactive system, and the interaction is done by (mostly) natural language. Blind (and indeed sighted) users can therefore navigate and ask questions about otherwise purely visually represented data, allowing not only (propositional) information extraction (as in (a)), but also exploration (as in (b)).

2. HOW THE SYSTEM WORKS

2.1 The input to *iGraph-Lite*

In our still-under-development system, when a statistician creates any kind of graph (usually in MS Excel), at the time of saving the worksheet an auxiliary file in XML is also generated. This file contains the most important information available in the graph, including titles, scales, values, etc and even colors and styles. This information is published at the same URL as the original graphical representation, except with the .xml extension. Thus, if a graph is at statcan.ca/c070130a.gif, the *iGraph-Lite* representation will be at statcan.ca/c070130a.xml.

This XML file generated by Excel is the primary input to *iGraph-Lite*. Once *iGraph-Lite* is pointed to the target graph's URL, it parses the input and a set of graph analysis classes construct another, much richer XML representation of the graph. These representations contain many different pieces of information such as the maximum values, shallow natural language processing on the titles for defining some context (for instance, if the x-axis title reads 'Month' and the values of the axis are '01', '02', '03', etc. we translate the numbers into more descriptive text such as 'January', 'February', 'March', etc.), the general shape of the graph, etc. Once the original XML has been completely analyzed by these processes, the resulting XML (inspired, by the way, in [5]) is input into the navigator, which takes care of natural language generation (NLG) and understanding (NLU).

2.2 NLG and NLU During Navigation

NLG and NLU are event-driven processes in *iGraph-Lite*. Every time a key is pressed during navigation, a message is sent to the NLG system. This message contains information on how to render a relevant natural language (NL) message. This NL message is read back to the user by a Text-To-Speech (TTS) engine. Below is a sample interaction with the graph in Figure 1, taken from The Daily, as the target graph. Everything in lines starting with [igr]:, the system

says by TTS.

```
[usr]: l http://statcan.ca/01244260e.xml
[igr]: I'm now looking at 01244260e.xml
[usr]: d
```

```
[igr]: Ok, I'm describing the whole graph now.
It's a line graph with one line. The main title
is 'operating profits dip in second quarter'.
The vertical axis is billions of dollars and
there is a sub-title that reads 'seasonally
adjusted'. The vertical axis goes from 0 to 70
(billions of dollars) and the horizontal axis
looks like quarters. It has a general upwards
trend and the maximum point is 57.6 in quarter
5, while the minimum is 25.9 in quarter 1.
```



Figure 1: A graph taken from “The Daily”

A few things should be noted here: *iGraph-Lite* has shortcut commands like ‘l’ (load graph) and ‘d’ (describe graph or segment, see below). The general description of the graph has many properties we have been able to gather in several studies with sighted, blind and visually impaired participants [3]. Presenting axes as vertical or horizontal, instead of x and y , presenting general information about the trend and reading the titles are some of these characteristics. Notice also that, in the description, the system “inferred” that the values of the abscissa were billions of dollars, while the values of the ordinate were quarters. Several commands exist in *iGraph-Lite* as it is, but one of the most important set of commands are the navigation commands. For lack of space we will discuss only two: the arrow commands (‘→’, ‘←’), the where ‘w’, and the skip ‘s’ commands. Consider the following exchange with *iGraph-Lite*.

```
[usr]: w
[igr]: You're at the beginning of the graph: Q1, value
25.9.
[usr]: →
[igr]: There is an small increase between Q1 and Q2
from 25.9 to 26.7.
[usr]: s 2 →
[igr]: Skipping 2 values. There is an small increase
between Q2 and Q1 from 26.7 to 28.9.
```

Again, there are several things to notice here. Most importantly, the arrow commands allows the user to move freely between or among points in order to be able to explore

the data set at any desired level of detail. Although it would be relevant to explore a small dataset point by point (or segment by segment (in ‘increase’, ‘decrease’ terms), this would make no sense for datasets that are thousands of data points long. The skip command helps in this respect by, unsurprisingly, skipping points and reporting the next segment. The ‘w’ (or ‘where’) command helps the user keep track of where in the graph he or she is positioned. Relevant choice of language (e.g. ‘small increase’) helps build a more intuitive mental representation [5] of the graph.

3. CONCLUSIONS AND FUTURE WORK

We have reported on our efforts to build *iGraph-Lite*, a tool to explore graphical representations of statistical data appearing in Statistics Canada “The Daily”, in order to make graphical information available and accessible to blind and visually-impaired users. This tool is inspired in the sound and NLG communities with respect to their use of natural language and interactivity. There are still several limitations with the system, although most of them have to do with merely enriching the set of messages and commands the system is able to communicate to the user. For instance, notice that the system did not find a correlation between quarters and the numbers (1996, 1997, etc.), which should have been inferred as years. This leads to a somewhat strange sounding combination (“There is an small increase between Q2 and Q1”) and some loss of information (Q2 is 1996 and Q1 is 1997). Dealing with these issues and building a more feature-rich system is our main research goal.

4. ADDITIONAL AUTHORS

Additional authors: Martin Lachance (Dissemination Division, Statistics Canada, email: Martin.Lachance@statcan.ca) and Antoine Chrétien (Dissemination Division, Statistics Canada, email: Antoine.Chretien@statcan.ca).

5. REFERENCES

- [1] L. Brown and S.A. Brewster. Drawing by ear: Interpreting sonified line graphs. In *Proceedings of ICAD 2003*, pages 152–156. ICAD, 2003.
- [2] M. Fasciano and G. Lapalme. Postgraphe: a system for the generation of statistical graphics and text. In *Proceedings of the 8th International Workshop on Natural Language Generation (INLG-96)*, 1996.
- [3] L. Ferres, A. Parush, S. Roberts, and G. Lindgaard. Helping people with visual impairments gain access to graphical information through natural language: The igrph system. In *Proceedings of the 10th ICCHP*, Lecture Notes in Computer Science. Springer-Verlag, 2006.
- [4] Vibhu O. Mittal, Giuseppe Carenini, Johanna D. Moore, and Steven Roth. Describing complex charts in natural language: a caption generation system. *Computational Linguistics*, 24(3):431–467, 1998.
- [5] S. Pinker. A theory of graph comprehension. In R. Freedle, editor, *Artificial intelligence and the future of testing*, pages 73–126. L. Erlbaum, Hillsdale, NJ, 1990.
- [6] J. Yu, E. Reiter, J. Hunter, and S. Sripada. A new architecture for summarising time series data. In *Proceedings of INLG-04 Poster Session*, pages 47–50, 2004.