Three's a Charm?: Open Event Data Coding with EL:DIABLO, PETRARCH, and the Open Event Data Alliance

Philip A. Schrodt, John Beieler and Muhammed Idris

Parus Analytical Systems, Pennsylvania State University
schrodt735@gmail.com, john.b30@gmail.com, muhammedy.idris@gmail.com

March 27, 2014

Why is event data suddenly attracting attention after 50 years?

- Rifkin [NYT March 2014]: The most disruptive technologies in the current environment combine network effects with zero marginal cost
- Key: zero *marginal* costs even though open source software is still "free-as-in-puppy"
- Examples
  - Operating systems: Linux
  - Statistical software: *R*
  - Encyclopedia: Wikipedia
  - Scientific typesetting and presentations: LaTeX

EL:DIABLO
Event Location: Dataset in a Box, Linux Option

- Full modular open-source pipeline to produce daily event data from web sources
- Scraper from white-list of RSS feeds and web pages
- Event coding from any of several coders: TABARI, PETRARCH, others
- Geolocation: Penn State "GeoVista" project coder, UT/Dallas coder
- Conventional reduplication keeping URLs of all duplicates
- Additional feature detectors are easily added
- Designed for implementation in Linux cloud (e.g. Linode: $20/month)

PETRARCH

- Python
- Full parsing using the Penn Treebank format and Stanford Core NLP. This handles the noun/verb/adjective disambiguation that accounts for much of the size of the TABARI dictionaries
- Synonym sets from WordNet
- Identifies actors even if they are not in the dictionaries
- Extendible through program "hooks": "issues" facility
- Probably will code at about 150 sentences per second, about a tenth the speed of TABARI but cluster computing is now readily available
- Unknown: how well will the TABARI dictionaries—based on shallow parsing—translate to the more precise full parsing?

Advantages of Python

- Open source (of course...tools want to be free...)
- Standardized across platforms and widely available and documented
- Automatic memory management (unlike C/C++)
- Generally more coherent than perl, particularly when dealing with large programs
- Text oriented rather than GUI oriented (unlike Java)
- Extensive libraries but these are optional (unlike Java)
- It may be possible to integrate C code at critical points for high-performance applications

Sources for historical texts

- LDC Gigaword 2000-2010; easily licensed
- SPEED (Cline Center, University of Illinois at Urbana-Champaign)
- Usual proprietary sources, but these are awkward and expensive
- Collective resources: in the US, coded data on facts does not inherit the IP constraints of the source
- Discussion of legal issues for US: http://asecondmouse.wordpress.com/2014/02/14/the-legal-status-of-event-data/

Open Event Data Alliance

- Institutionalize event data following the model of CRAN and many other decentralized open collaborative research groups: these turn out to be common in most research communities
- Provide at least one source of daily updates with 24/7/365 data reliability. Ideally, multiple such data sets: "one data set to rule them all" is *soooo* twentieth century
- Establish common standards, formats, and best practices
- Open source, open collaboration, open access
- Will not give annual awards nor proliferate committees

Extending the event ontologies

CAMEO and IDEA were derived from earlier Cold War event ontologies (WEIS, COPDAB, World Handbook) and consequently miss substantial amounts of political behavior that is currently relevant.

- natural disaster
- disease
- criminal activity
- financial activity
- refugees and related humanitarian issues
- human rights violations
- electoral and parliamentary activity

Improve the named entity recognition

- There is a very large literature on this in NLP
- Actors have a power-law distribution, so investing work in the most frequent names with have a high payoff
- The value of names in the long tail is less clear, given that infrequent names are almost always introduced in a context that provides the state and role. However, we need feature extractors to get these.
- Computational methods may not be more efficient than simply using trained coders. Crowd-sourcing probably is not efficient (even if it is possible)

Thank you

Email:
schrodt735@gmail.com
john.b30@gmail.com
muhammedy.idris@gmail.com

Slides:
`http://eventdata.parusanalytics.com/presentations.html`

Software: `https://openeventdata.github.io/`

Papers:
`http://eventdata.parusanalytics.com/papers.html`