The Journal of Open Source Software

UTDEventData: An R package to access political event data

HyoungAh Kim[1], **Vito D'Orazio**[1], **Patrick T. Brandt**[1], **Jared Looper**[1], **Sayeed Salam**[2], **Latifur Khan**[2], **and Michael Shoemate**[3]

1 School of Economic, Political and Policy Sciences, University of Texas at Dallas **2** Department of Computer Science, University of Texas at Dallas **3** School of Natural Sciences and Mathematics, University of Texas at Dallas

DOI: 10.21105/joss.01322

Software

- Review
- Repository
- Archive

Submitted: 13 March 2019
Published: 22 April 2019

Introduction

Political event data record interactions among social and political actors. Researchers use these data to understand relations among actors, predict outcomes of interest, and forecast trends (Schrodt, 2012). As automated technologies have become better able to extract events from text, event data projects and repositories have increased in number. For example, the Cline Center at the University of Illinois contains events from 1945 to 2015 coded from three different news sources (Althaus, Bajjalieh, Carter, Peyton, & Shalmon, 2017). The Integrated Crisis Early Warning System (ICEWS) program regularly delivers updates to their repository on Harvard's Dataverse (Boschee et al., 2015). The Temporally Extended, Regular, Reproducible International Event Records (TERRIER) database at the University of Oklahoma contains 73 million political event records from 1979 to 2016 (Halterman et al., 2017). At the University of Texas at Dallas (UTD), we have built an automated political event data system called "Spark-based Political Event Coding (SPEC)", that extracts and processes political event data from over 380 different English written news media on a daily basis, and has been doing so since October 2017 (Solaimani et al., 2016). Technologies for accessing event data resources exist (D'Orazio, Deng, & Shoemate, 2018), but not directly through R. The UTDEventData R library provides access to these event data and is flexible to access new sources as they become available.

The UTD political event data API server

The server is hosted in XSEDE's JetStream cloud and provides access to six datasets: three from the Cline Center, the ICEWS data, TERRIER data, and the real-time data coded by SPEC at UTD. We use a Flask API to handle web requests, data is accessed from MongoDB, and served in JSON format (Salam, Brandt, Holmes, & Khan, 2018). Users are required to obtain an API key for access http://eventdata.utdallas.edu/signup. More details about this repository, including how to obtain an API key, may be found at http://eventdata.utdallas.edu/.

The `UTDEventData` R package

`UTDEventData` enables direct access to the UTD event data repository through R. It includes functions to explore, to query, and to download event data from any table in

the repository. Exploratory functions include `DataTables()`, which provides a list of all available datasets, `tableVar()`, which lists the variables in a particular data table, and `previewData()`, which downloads a sample of the data for browsing. This centralization of event data resources, and a single interface for access, facilitates workflows for researchers working with these datasets. It also provides a convenient way to handle large datasets when only a subset is of interest.

Users can subset and download the data using one of two functions: `pullData()` or `sendQuery()`. The `pullData()` function is simpler, and provides a fixed way to retrieve data using only date and location information. `sendQuery()` is more flexible, and allows users to build up a query element-by-element. For example, helper functions such as `returnDyad()` returns the query element to ask for a pair of actors as source and target, and `returnRegExp()` allows for virtually any single query element using MongoDB regular expressions. These single elements are then combined using either their intersection (`andList()`) or their union (`orList()`), which forms the query to submit with the `sendQuery()` method.

A user is required to enter his or her API key as an argument for each of the aforementioned functions. For convenience, we have a reference class called `Table` that includes a field for storing the user's API key, and has methods that mirror the above functions `pullData()`, `tableVar()`, and `DataTables()`. With `Table`, users are only required to submit their API key once.

This R library has additional functions to cite the data table (`citeData()`), gauge the size of a query (`getQuerySize()`), and to download an entire data table directly to disk (`entireData()`). More information about the `UTDEventData` package, along with the package vignette, can be found at the project's Github page, https://github.com/KateHyoung/UTDEventData.

`UTDEventData` provides an accessible and user-friendly environment to access political event data. As automated methods develop, and event datasets proliferate in number and expand in magnitude, we expect new and larger tables to be added to the repository and accessed through our interface.

Acknowledgements

This R library has been developed upon the work funded by the National Science Foundation under Grant No. SBE-SMA-1539302.

References

Althaus, S., Bajjalieh, J., Carter, J. F., Peyton, B., & Shalmon, D. A. (2017). *Cline Center historical Phoenix event data. V.1.0.0.* Retrieved from http://www.clinecenter.illinois.edu/data/event/phoenix/

Boschee, E., Lautenschlager, J., O'Brien, S., Shellman, S., Starz, J., & Ward, M. (2015). *ICEWS coded event data.* doi:10.7910/DVN/28075

D'Orazio, V., Deng, M., & Shoemate, M. (2018). TwoRavens for event data. In *2018 IEEE International Conference on Information Reuse and Integration (IRI)* (pp. 394–401). IEEE. doi:10.1109/iri.2018.00065

Halterman, A., Irvine, J., Landis, M., Jalla, P., Liang, Y., Grant, C., & Solaimani, M. (2017). Adaptive scalable pipelines for political event data generation. In *2017 IEEE*

International Conference on Big Data (pp. 2879–2883). IEEE. doi:10.1109/bigdata.2017.8258256

Salam, S., Brandt, P. T., Holmes, J., & Khan, L. (2018). Distributed framework for political event coding in real-time. In *2nd European Conference on Electrical Engineering and Computer Science (EECS), Bern, Switzerland*. EECS. Retrieved from http://www.eecs-conf.org/

Schrodt, P. A. (2012). Precedents, progress, and prospects in political event data. *International Interactions*, *38*(4), 546–569. doi:10.1080/03050629.2012.697430

Solaimani, M., Salam, S., Mustafa, A. M., Khan, L., Brandt, P. T., & Thuraisingham, B. (2016). Near real-time atrocity event coding. In *2016 IEEE Conference on Intelligence and Security Informatics (ISI)* (pp. 139–144). IEEE. doi:10.1109/isi.2016.7745457