Comparing Methods for Generating Large Scale Political Event Data Sets *

Philip A. Schrodt
Parus Analytics
Charlottesville, Virginia, USA
schrodt735@gmail.com

Version 1.0 : October 6, 2015

*Paper presented at the Text as Data meetings, New York University, 16-17 October 2015. Partial results were earlier presented at the Conference on Forecasting and Early Warning of Conflict, Peace Research Institute, Oslo (Norway), April 2015 and the European Political Science Association meetings, Vienna (Austria), June 2015. The opinions expressed herein are solely those of the author and do not reflect those of any of the various data projects with which the author has been associated now or in the past—except, I suppose, KEDS—particularly those funded by the U.S. government. The derived data and programs used for both the data preparation and analysis are reasonably well documented and are available from the author, as is the new PETRARCH-coded Levant series, which I'll eventually get around to posting on Dataverse and `http://eventdata.parusanalytics.com/data.dir/levant.html` but haven't gotten around to at this point. A link to the most recent version of this paper can be found at `http://eventdata.parusanalytics.com/papers.dir/automated.html`

Abstract

This paper addresses three general issues surrounding the use of political event data generated by fully automated methods in forecasting political conflict. I first look at the differences between the data generation process for machine and human coded data, where I believe the major difference in contemporary efforts is found not in the precision of the coding, but rather the effects of using multiple sources. While the use of multiple sources has virtually no downside in human coding, it has great potential to introduce noise in automated coding. I then propose a metric for comparing event data sources based on the correlations between weekly event counts in the CAMEO "pentaclasses" weighted by the frequency of dyadic events, and illustrate this with two examples:

- A comparison of the new ICEWS public data set with an unpublished data set based only on the BBC *Summary of World Broadcasts*.
- A comparison of the TABARI shallow parser and PETRARCH full parser for the 35-year KEDS Reuters and Agence France Presse Levant series.

In the case of the ICEWS/BBC comparison, the metric appears useful not only in showing the overall convergence—typical weighted correlations are in the range of 0.45, surprisingly high given the differences between the two data sets—and showing variations across time and regions. In the case of TABARI/KEDS, the metric shows high convergence for the series with a large number of reports, and also shows that the PETRARCH coding reduces the number of material conflict events—presumably mostly by eliminating false positives—by around a factor of 2 in most dyads. In both tests, the metric is good at identifying anomalous dyads, Asia in the case of ICEWS and Palestine in the case of the TABARI-coded Levant series.

Finally, the paper looks at the degree to which the complex coding of PETRARCH can be duplicated using much simpler "bag of words" methods, specifically a simple pattern-based method for identifying actors and support vector machines for identifying events. While, as expected, these methods do not fully reproduce the more complex codings, they perform far better than chance on doing the aggregated classifications typically found in research projects, and arguably could be used as a first approximation for new behaviors where automated coding dictionaries are not available. The pattern-based actor and SVM models also strongly suggests that there may be a very substantial number of sentences which are currently not coded which actually contain events.

1 Introduction

Political event data were originally developed in the 1970s—primarily for the purpose of forecasting international conflict—under the sponsorship of the U.S. Department of Defense Advanced Research Projects Agency (DARPA) [Choucri and Robinson, 1979, Andriole and Hopple, 1988]. The approach experienced a long period of gestation and development largely under the sponsorship of social science programs in the U.S. National Science Foundation [Merritt et al., 1993], and in recent years was the focus of a major political forecasting effort, the Integrated Conflict Early Warning Systems (ICEWS) sponsored, once again, by DARPA [O'Brien, 2010]. After a long delay, the extensively-documented ICEWS data for 1996-2014 have been released for public use (http://thedata.harvard.edu/dvn/dv/icews) and monthly updates are promised.

ICEWS is just the latest addition to a large group of data sets focusing on political conflict events (see [Schrodt, 2012] for a review). In a small number of cases, notably the recently introduced Phoenix data set (http://phoenixdata.org) from the Open Event Data Alliance (OEDA; http://openeventdata.org), which is updated daily and has a data processing pipeline which is completely open source, these are general purpose. Other widely used data sets, including the long-established COW and MIDS data (http://www.correlatesofwar.org/) up to more recent entrants such as the ACLED (http://www.acleddata.com/,[Raleigh et al., 2010]) and Uppsala (http://www.pcr.uu.se/research/UCDP/) conflict data sets, the GTD terrorism data (http://www.start.umd.edu/gtd/[START, 2012]), and the SCAD protest data [Salehyan et al., 2012], focus on specific types of events, but for the purposes of most of this discussion these can also be considered event data. In fact, given that the event coding ontologies such as WEIS, COPDAB, IDEA and CAMEO used in the general event data sets such as ICEWS and Phoenix cover a limited subset of behaviors primarily associated with violent conflict, the distinction between these and the more focused data is just one of degree, not one of kind.

This paper will address four issues related to the use of event data in political forecasting. First, I will discuss the "data generating process" (DGP) that applies to event data, with a particular focus on the differences between machine and human coding. I argue that while at one point these differed primarily in the extent to which machines could code individual sentences as well as humans, that issue has probably been resolved and the important distinction is now how humans and machines use multiple news sources.[1] Second, I propose and provide an illustrative example of a metric for systematically comparing multiple streams of general event data, which also demonstrates the surprisingly limited gains of large-scale multiple sourcing, but substantial gains in the computationally-intensive full parsing. Finally, I look at the degree to which simple "bag of words" methods can reproduce the results obtained with the more complex processing of Petrarch.

[1] For purposes of this paper, "source" refers to a textual news source such as Reuters, Agence France Press, *The New York Times*, Xinhua, al-Jazeera or whatever. This is not to be confused with the source *actor*, a component of the data.

2 The data-generating process for event data

About twenty years ago I explored the issue the DGP for event data in some detail [Schrodt, 1994]. At the time I wrote that article, automated coding was just beginning to be developed and in most ways simply mirrored the human coding process except that a computer program was substituted for the human coder. Some of the points made in that original article are still relevant, but a critical one has changed: the use of multiple news sources. Early human coded series were generally single source,[2] and this continued to the early KEDS, PANDA and IDEA data sets produced by fully-automated methods, which originally used Reuters, later supplemented by a small number of additional sources.

In contrast, thanks first to data aggregators such as Lexis-Nexis, Factiva, and later Google, and now to the web generally, contemporary efforts use a large number of sources: both ICEWS and Phoenix use about 200. In the case of human coding, coders usually have access to the full panoply of web-based search tools, both media sources and reference sources such as [admit it...] Wikipedia. The way these multiple sources are used, however, differs significantly and that will be the focus of this discussion.

2.1 The Platonic ideal: incidents meet ontologies

All event data coding is based on two fundamentals: there are *incidents* that occur in the world that correspond to categories in a coding *ontology*. The ideal data set would have a single properly coded *event* for every incident where the ontology had a corresponding code. Event data ontologies all actually specify multiple items to be coded from an incident. For example in the basic WEIS/CAMEO ontology [Schrodt et al., 2009] an "event" includes a date, source actor, target actor, and event code. Many contemporary data sets, including ICEWS and Phoenix, also code for location, and many the human-coded data sets such as COW and ACLED have a large number of fields. But the principle here is straightforward: every codeable incident generates an event according to an ontology.

That's the ideal, which is unachievable. Let's now look at the key places where errors come in.

2.2 Only some incidents broadcast generate news stories

Probably the single biggest filter in the entire process of generating event data from incidents on the ground is whether an incident generates a textual record that can be coded at all [Davenport and Ball, 2002]. This can be affected by at least the following factors:

[2] COPDAB claimed to use multiple sources, and may have for some periods when the data were collected, but the differences in density compared to the single-sourced WEIS make this claim almost impossible to sustain, particularly in the absence of event-by-event documentation of those sources. The COW family of single-event-type datasets was definitely multiply-sourced, with coders doing extended research in a variety of publications.

- "Newsworthiness": media coverage exists to sell newspapers—or at least to sell some news stream—rather than to provide input for analysts and social scientists. Editors are always asking "so what, who cares?"
- Whether a reporter witnesses or is informed of the incident. Many high conflict areas are effectively off-limits to reporters, certainly on a day-to-day basis.
- The character of the incident: routine events that are easy to cover such as meetings and press conferences get a lot of coverage, as do sensational "when it bleeds it leads" stories. Things in between—for example protracted negotiations or low-level conflict—are much less likely to be covered consistently.
- Explicit or implicit censorship: freedom of the press varies widely and idiosyncratically. Censorship may be overt—China's "great firewall" [King et al., 2013]—or reporters may simply know that they are asking for trouble if they cover certain topics.
- Rashomon effects: An incident will be reported in multiple ways, either through differences in point of view, but often simply because of the difficulty of gathering information: it can take days to figure out how many people were killed in a car bombing.
- Wealth effects: news coverage generally "follows the money."

An incident, in short, can generate anywhere from zero to hundreds of texts. The zero report cases, of course, will never enter into our data.[3] For the cases which do enter, the results from human and automated coding diverge quite substantially.

2.3 Human coding: human readers reconcile multiple sources

The human coding process can be dealt with more quickly since almost everyone reading this paper has had some experience with it. Unless constrained by coding protocols—which is to say, pretending to be a machine—a human coder will take the cloud of texts generated from a single incident by multiple source and try to distill a single event.[4] This typically will be done against an extensive background of knowledge about the event, and will involve at least:

[3] The news environment also contains some texts that refer to incidents that did not occur at all. In the "mainstream" sources I use for coding the PITF Atrocities Data Set (ADS; `http://eventdata.parusanalytics.com/data.dir/atrocities.html`), such cases appear to be very rare: I certainly find reports that are too fragmentary to code, and there seems to be a some correlation between the physical distance one is from an incident and the gravity of it (that is, second-hand reports seem more dramatic), but the number of purely false reports is almost certainly much less of an issue than non-reports. In government controlled-media, however, this is a huge issue: the Russian media have created a counter-narrative on Ukraine completely at odds with US, European and Ukrainian government sources, which has been true of propaganda since time immemorial. These counter-narratives are probably fascinating in their own right and by no means confined to Vladimir Putin, as the fantasies of U.S. right-wing media, and increasingly, presidential candidates, demonstrate. With the appropriate selection of conflicting sources, event data could in fact be very productively used to study this sort of thing systematically.

[4] For purposes of simplicity, let us assume the ontology is such that in the Platonic ideal, an incident generates one and only one event.

- sorting out reports that are irrelevant, redundant or for various reasons are less than credible
- extract relevant information from the remaining texts
- generating a mental “median narrative” out of the entirety of the texts
- applying the ontology to this to determine the appropriate event coding

In reading reports of events, humans have a very strong cognitive tendency to assemble narratives: see the extended discussions of this in [Taleb, 2010, Kahneman, 2011]. Human episodic processing is cognitively tuned to ”connect the dots” and seek out missing links, and redundant information is invisible, particularly to experienced analysts.

That’s all great—and after all, event data coding started with human coding and consequently has largely been aligned with human cognitive capabilities, if not motivations—so let’s just use human coders! Which is precisely what some systems, including my own atrocities coding for the Political Instability Task Force, do. The problem is that human coding projects have limited capacity, particularly when dealing with real-time data. Consequently, we turn to machines, where we find a situation that is very different.

2.4 Machine coding: ontological receptors latch on to stories

With current technology available to social scientists,[5] automated coding works a bit like an immune system shown (sort of) in Figure 1: we have texts floating around that might correspond to events, and event detectors—instantiated as dictionaries—for what a sentence that reports an event might look like. When an event-detector matches a sentence, rather as a key fits a lock, we have an event. The systems for doing this have become more complex over time: the KEDS program was producing useable, which is to say publishable, data on the Middle East with dictionaries containing about a thousand political actors and a couple thousand verb phrases, whereas the ICEWS data uses an actor dictionary with over 100,000 entries and the Phoenix system has over 15,000 verb phrases. But all of the automated coding systems I’m aware of, including KEDS, IDEAReader, Tabari, JABARI, Petrarch and ACCENT, work pretty much along these lines.

The newer systems are probably approaching the accuracy of human coding teams[6]—the BBN coder has been systematically assessed as having a *precision* of about 80%, which is to say that 80% of the cases where ACCENT codes something into a particular category, it belongs in that category. Whether these systems are quite at the level of human coding projects is not settled, both because of the lack of a systematic gold standard against which

[5] That is, if we had available something like the IBM Watson system, we could probably do better. And now, in fact, in principle we could, as IBM has recently made Watson available as a commercial service, though the URL for the site stretches from here to West Virginia so for a reference just Google `"IBM Watson"`. Watson-level NLP: cool. However, I suspect it is not cheap. So for practical purposes, we don’t have this capability.

[6] Which is frequently not particularly high [Ruggeri et al., 2011], particularly when multiple institutions are involved and students, rather than professional coders, are hired to do the work.

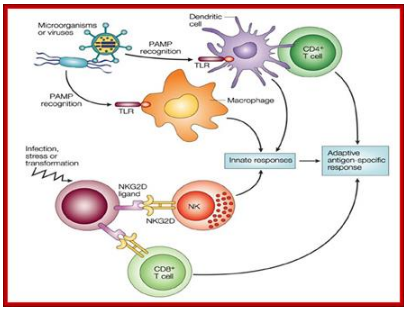


Figure 1: Immune response

to evaluate the automated systems but also because it is probably the case that the intercoder reliability for human systems, particularly those working over long periods of time (that is, have substantial coder turn-over and re-training) and across multiple institutions is probably considerably lower than the usually-reported figure of 80%.[7]

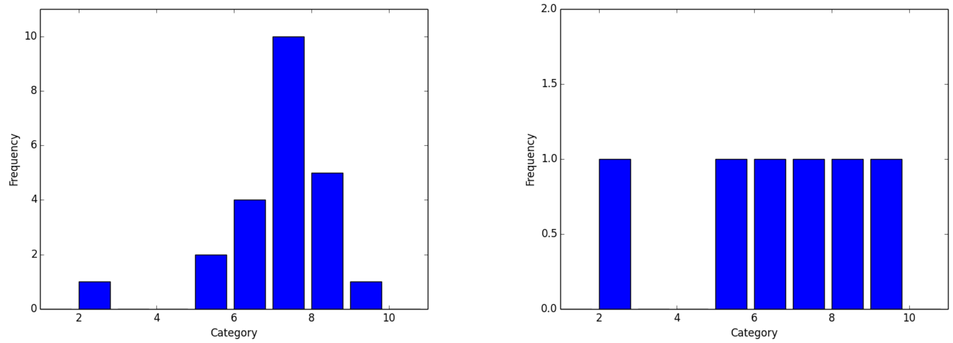
However, the machine-coding environment does not automatically ignore redundant information, so when more texts are available, these will generally produce more events, even from a single incident. This duplication is an issue even in single-source data sets, since news feeds that are providing near-real-time coverage will give updates from an incident, for example a car bombing, as more details emerge about the story, though often these texts will be sufficiently similar that they will generate the same event code. In multiple-source series, duplicates are extremely common because news wire stories are repeated by newspaper subscribing to the service either verbatim or with slight editing—and alas, even slight editing will sometimes change the coding, particular at levels below the primary code—and multiple news wires (in addition to local sources) will be covering the same story.

There is a literature in computer science on "near duplicate detection" that can automatically cluster stories that are likely to refer to the same event, and this technology is used, for example, in Google News (`https://news.google.com/`) and European Media Monitor (`http://emm.newsbrief.eu/overview.html`). To the best of my knowledge this has not been used to date in the automated coding of event data. Instead, the most common (and computationally simple) approach is to use the "One-A-Day" (OAD) filter that we originally developed to deal with multiple reports in a single source.

In the ideal world, OAD filtering would not be problematic, and would simply reduce multiple mentions of a single incident to a single mention: this was the intention. We are not, however, in an ideal world, but rather one where there is coding error—which is to say, non-zero entries off the main diagonal of the classification matrix—and instead OAD filtering has the effect illustrated in Figure 2 (note the change of scale between the two subfigures): all of the distinct events, correct or incorrect, generated by the stories generated by the incident produce a single event in the filtered set.[8] The scariest aspect of Figure 2: *most* of the events generated

[7] Or a Kronebach's alpha of 0.8 or whatever: my own sense is that this number is so far off from what occurs in long-term projects producing tens of thousands of events that the differences in particular metrics are not important.

[8] At various times arguments have been made that the frequency of repetition is a measure of the im-



(a) What was generated

(b) What remains

Figure 2: Effect of One-A-Day filtering

by any incident will have been incorrectly coded!

Let's put this together mathematically: Let S be a set of sources, T the set of texts generated by an incident and C be the classification matrix (see Appendix):

In this framework:

Probability an incident generates at least one text: $1 - \prod_{i \in S}(1 - p_i)$
where p_i is the probability that source i generates at least one text

Probability a text generates at least one correct coding: $1 - \prod_{i \in T}(1 - C_{i,i})$
where T is the set of texts generated by the incident

So far, so good, though note that both of these measures increase rapidly at the size of S and T increases, and are asymptotic to 1.0: that is, the pattern we see in Figure 3. The problem occurs when we look from the other side at the false positives:

Expected number of incorrectly coded events generated by incident: $\sum_{i \in T}(\sum_{j \neq i} C_{i,j})$

This points to a core problem: as the number of sources (and hence texts) increases, we see diminishing returns on the likelihood of a correct coding, but a linear increase in the number of incorrectly coded events. If a filter is used which eliminates multiple positives, after a certain point, increasing sources does nothing to increase the number of true positives—this is already essentially 1.0—but can increase the number of false positives up to the point where all categories are filled.

This is not a good situation.

portance of an event, and this could be incorporated into models. This is not an actively bad idea, but it will come with a great deal of noise: placement of news wire stories is partly a function of importance, but it is also a simple function of the amount of space a venue has available on a particular day. This would also amplify the bias towards event occurring in areas that are already getting extensive coverage: compare for example the number of stories dealing with the Charlie Hebdo attacks in Paris to the contemporaneous attacks in Nigeria, Iraq and Syria which had far more casualties. The approach *might* contribute some additional signal, but I doubt it will have a large effect.

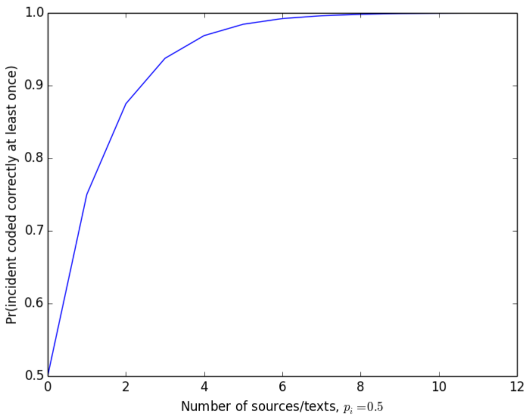


Figure 3: Probability of generating at least one correct event as a function of the size of S or T

Looking at Figure 2, the reader's initial reaction is probably that the answer is simple: look at the events generated by an incident, and simply choose the highest-frequency code. Unfortunately, we don't see something like Figure 2 which is sorted by *incident*, we only see the resulting *event codes.* An extreme filter, for example, might extend the OAD concept to allow only one type of event per dyad per day, but this would eliminate cases where, for example, two actors are simultaneously engaging in material and verbal conflict, or where an actor is simultaneously meeting with another actor and criticizing it—both situations are common. Furthermore, this assumes that the assignment of actors is accurate, and the assignment of targets in particular remains a weak point in many automated coding systems. The solution instead probably involves developing technology either for pre-filtering to cluster the texts, or else technology for using source information in filters, though note if this is applied too strictly—that use, only use codes that occur in multiple sources—this will favor news wire stories that were simply reprinted, and remove "missed dots" information provided only by a single, probably local, source.

3 A uniform metric for the comparison of event data sets

There is an old saying that a person with one watch always knows the time, but a person with two watches is never sure. Given the complexity of the DGP outlined above, how do we know which data set is "correct."

The short answer, of course, is that we never will. It would be useful, however, to have some idea of the extent to which different data sets—and the sources on which those data sets are based—converge. To date, this has largely been done by comparing the correlations on some major dyads. My suggestion is that we extend this to look at a weighted correlation of the most important dyads in the data set, where the weight is based on the proportion of events accounted for by the dyads. Specifically, I will compute the measure under the following criteria

- Find the actors which are involved in 80% of the events in the two data sets and look at all undirected dyads[9] involving these actors[10]
- Compute pentaclass[11]counts and Goldstein-Reising[12] totals by week, then run correlations between these vectors for the two data sets

The composite weighted correlation is

$$wtcorr = \sum_{i=1}^{A-1} \sum_{j=i}^{A} \frac{n_{i,j}}{N} r_{i,j} \qquad (1)$$

where

- A = number of actors;
- $n_{i,j}$ = number of events involving dyad i,j
- N = total number of events in the two data sets which involve the undirected dyads in A x A[13]
- $r_{i,j}$ = correlation on some measures: typically counts and Goldstein-Reising scores aggregated at a weekly or monthly level

I've implemented these calculations in a Python program which, as always, I'm happy to share.

[9] Obviously it would be trivial to modify the code to deal with directed dyads but dyadic symmetry is extremely strong in event data and this is unlikely to make much difference.

[10] There is probably some way to construct an artificial situation where a dyad could be part of the 80% of dyadic events and at least one of the actors not be part of the 80% monadic measure, but in practical terms this will almost never occur. One does find, however, that many of the dyads generated from the monadic list have no events in one or both of the data sets.

[11] The concept of a "pentaclass" is found in Phoenix and are similar to the "quad counts" long used in the KEDS research (and coded in the ICEWS data) except that the high-frequency "Make Public Statement" and "Appeal" categories go to a new '0' category. The remaining classes are the same as the quad counts: 1 = verbal cooperation, 2 = material cooperation, 3 = verbal conflict and 4 = material conflict.

[12] The so-called "Goldstein" weights found in most CAMEO-coded data sets are not actually those of [Goldstein, 1992], who developed weights for the *WEIS* ontology, but rather a similar set of weights developed for CAMEO by Uwe Reising, a University of Kansas graduate student who needed these for a thesis.

[13] The total is taken over the dyads rather than the size of the data because in contemporary data sets a very large number of events are internal: the primary actor code is the same for both the source and target. Dyads outside the set A x A will also be a factor but a much smaller one because of the extremely thin tail of the distribution.

4 Application 1: ICEWS multi-source data versus a BBC single-sourced data set

The first application of the metric will be to the recently-released ICEWS public data set, and an unreleased PETRARCH-coded data set from the Cline Center for Democracy at the University of Illinois/Urbana-Champaign (`http://www.clinecenter.illinois.edu/`) that is based only on BBC *Summary of World Broadcasts*(BBC-SWB). The Cline Center data set is still under development and while it will eventually go to 2014, at the present time the overlap of the two data sets is only 1996-2008. Using ISO-3166-alpha3 codes, the countries accounting for 80% of the events, in order of monodic frequency, are USA, RUS, CHN, IND, JPN, ISR, IRQ, PSE, IRN, GBR, PAK, TUR, AUS, KOR, AFG, FRA, TWN, IDN, PRK, DEU, UKR, EGY, THA, GEO, PHL, MEX, NGA, ZAF, SRB, ESP, CZE, LBN, SDN, BRA, COL, HRV, ITA, AZE, SVK, BGD, SYR, UGA, KEN, POL, LKA, ARG, VNM, MYS, NPL, SAU, ROU, CAN, VEN, NZL and ZWE

Figures 4 and 5 show the correlations between the two data sets by year on various metrics; the temporal level of aggregation is the week. Three things are immediately obvious from the comparison. First, while the correlation of the count totals in Figure 4 is generally fairly high, we can see from Figure 5 that this is mostly due to the two lowest of the pentaclass categories, which deal with announcements and appeals. That same factor is the reason that the correlations in the counts are somewhat higher than the correlations in the Goldstein-Reising scores. This is further confirmed in Figure 5, where there are considerably higher correlations in code classes 0 and 1 than in classes 2, 3, and 4. Class 4—material conflict—actually has the lowest correlation.

Second, the various measures generally track each other fairly well over time. There are idiosyncratic differences in the changes by year but these are not dramatic: the curves are more or less parallel.

Finally—and the one possibly useful piece of information from this largely illustrative analysis—the correlations are clearly lower in 1996 and 1997 than in the remainder of the series. This is consistent with a number of issues that have been raised about the consistency of the ICEWS data over time, and particularly the first two years of the data.[14]

Tables 3 and 4 show the dyads that have the highest and lowest correlations across the entire period as measured by the total counts. As would be expected, highest correlation dyads are primarily high visibility states, with Russian (RUS) and China (CHN) accounting for fully 72% of these, and interestingly the RUS-CHN dyad having the single highest correlation. Curiously, the USA does not occur in this list, which might indicate differences between the BBC-SWB and ICEWS coverage, possibly due to the DARPA-funded ICEWS by statute not being able to monitor the USA, though the documentation says that only internal events—which I'm not including in this analysis—were eliminated. The low average correlation

[14] Concerns have also been raised about the last four years of the data, but I currently don't have any overlap where I can test this.

cases,[15] in contrast, are generally random, though these may be disproportionately occurring with Asian states: for example 54% of the low frequency cases involve states that were the Asian focus of the original ICEWS research—CHN, IND, JPN, AUS, KOR, TWN, IDN, PRK, THA, PHL, BGD, SYR, LKA, VNM, MYS, NPL, and NZL—while those inter-Asia dyads are only 9% of the dyads being tracked.

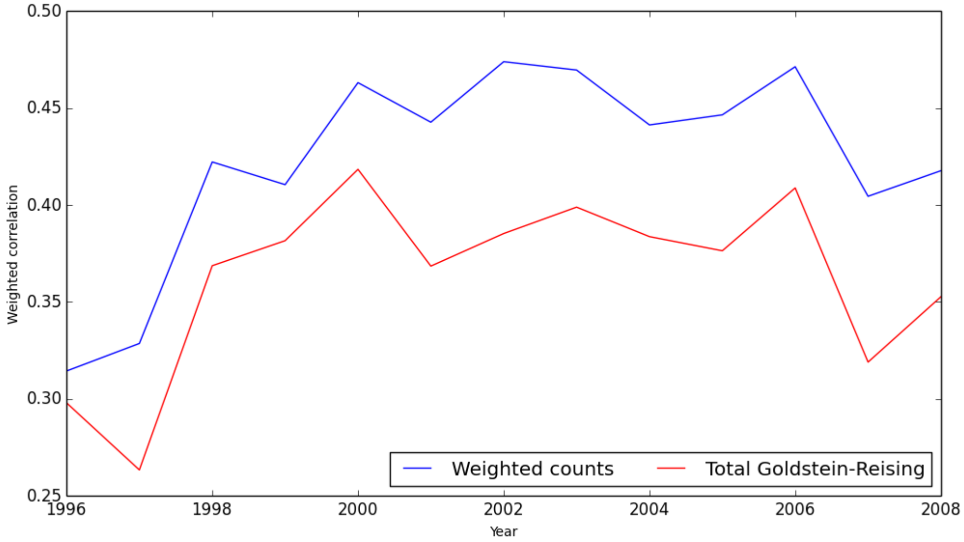


Figure 4: Weighted correlations by year: weighted counts and Goldstein-Reising totals

Table 1: Fifty dyads with highest average correlation on total counts

RUS-CHN 0.76	CHN-ZAF 0.72	CHN-EGY 0.67	CHN-PAK 0.66	CHN-DEU 0.66
CHN-SYR 0.66	CHN-HRV 0.65	CHN-JPN 0.64	RUS-JPN 0.63	UKR-HRV 0.63
RUS-IRN 0.61	CHN-FRA 0.60	CHN-ROU 0.60	CHN-IND 0.59	CZE-HRV 0.59
CHN-GBR 0.59	CHN-MEX 0.59	RUS-PSE 0.59	CHN-LKA 0.59	CHN-VNM 0.59
HRV-ROU 0.58	CHN-PSE 0.58	RUS-IND 0.58	RUS-DEU 0.57	TUR-POL 0.57
CHN-TUR 0.57	IRN-PAK 0.56	CHN-IRN 0.56	IRN-TUR 0.56	RUS-VNM 0.56
IRN-SYR 0.56	CHN-BRA 0.55	CHN-ESP 0.55	RUS-GBR 0.55	TUR-UKR 0.55
DEU-ROU 0.54	USA-CHN 0.54	RUS-CAN 0.54	CHN-AUS 0.54	RUS-EGY 0.54
CHN-ARG 0.54	RUS-ISR 0.54	TUR-ROU 0.54	RUS-SYR 0.54	RUS-POL 0.54
UKR-SVK 0.54	TUR-GEO 0.53	RUS-ROU 0.53	PSE-PAK 0.53	RUS-KOR 0.53

The correlations between these two series may appear relatively low, but keep in mind we would not expect them to be all that high because they are comparing data generated from a *single* BBC-SWB news stream with the 200 or so sources used in ICEWS. There are also methodological differences between the sets: for example ICEWS codes into the BBN CAMEO-B dialect whereas the BBC-SWB was coded into the original CAMEO (albeit this probably makes relatively little difference at a pentaclass level of aggregation), and the coding

[15] These are only the correlations that could be computed. Quite a few dyads had no observations in one or both of the data sets: these were treated as zero.

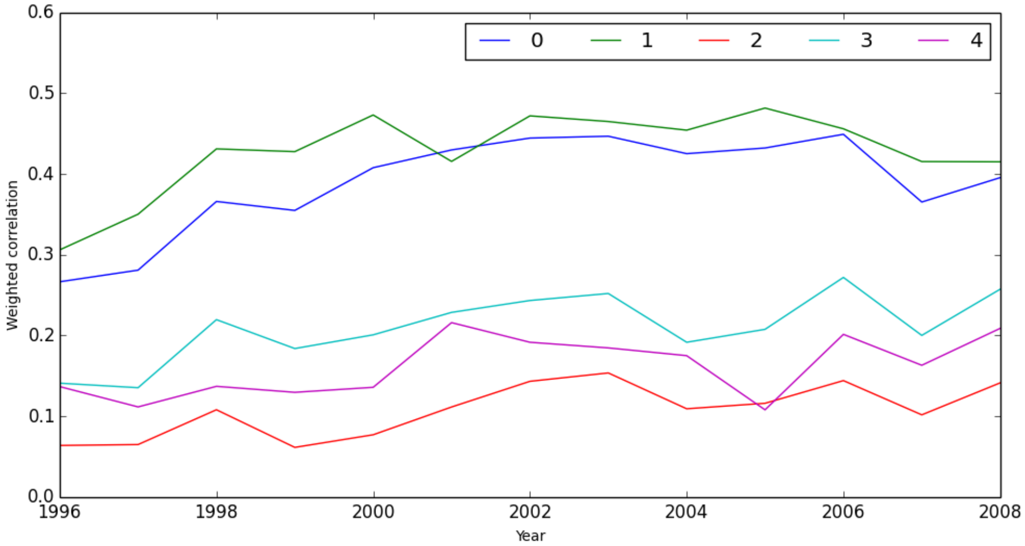


Figure 5: Weighted correlations by year: Pentacode classes

Table 2: Fifty dyads with lowest average correlation on total counts

MEX-SAU -0.0090	AUS-ITA -0.0086	GBR-VEN -0.0060	ISR-BGD -0.0060	AFG-SYR -0.0050
BRA-POL -0.0047	AFG-LKA -0.0045	SAU-NZL -0.0043	AUS-CZE -0.0042	CZE-LKA -0.0038
IDN-AZE -0.0037	ITA-NZL -0.0031	PRK-SAU -0.0030	IRQ-ZWE -0.0030	IND-ARG -0.0029
NPL-CAN -0.0028	PHL-LKA -0.0028	BRA-ITA -0.0027	VNM-SAU -0.0025	ESP-MYS -0.0025
NGA-LBN -0.0025	NGA-ITA -0.0025	PHL-ARG -0.0024	PSE-GEO -0.0024	IRN-NPL -0.0023
AZE-MYS -0.0022	GEO-SYR -0.0022	EGY-MEX -0.0022	BGD-SYR -0.0021	CAN-NZL -0.0020
TWN-EGY -0.0020	PRK-KEN -0.0019	COL-BGD -0.0018	PRK-LBN -0.0018	EGY-VEN -0.0018
CZE-VEN -0.0016	KOR-GEO -0.0016	KOR-VEN -0.0015	TUR-VEN -0.0015	NGA-VNM -0.0015
PHL-KEN -0.0015	SVK-SAU -0.0015	AFG-BRA -0.0015	SVK-ZWE -0.0015	AFG-VEN -0.0015
GEO-SAU -0.0015	KOR-ZWE -0.0015	SYR-ARG -0.0015	PSE-MEX -0.0014	ZAF-NZL -0.0014

engines are different (BBN ACCENT for ICEWS and the OEDA Petrarch-1 for the BBC-SWB series). The fact that the highest correlation dyads are probably around the level of human inter-coder reliability would be to be a very high convergence given these differences, and completely consistent with the argument I made in Section 2: the gains from adding additional sources fall off extremely quickly.

The documentation for ICEWS [n.a., 2015] provides only the following information on de-duplication:

> Duplicate stories from the same publisher, same headline, and some date are generally omitted,[2] but duplicate stories across multiple publishers (as is the case in syndicated new stories) are permitted.
>
> Footnote 2: Due to changes in data processing there are an increased number of inadvertent duplicate stories in the underlying story set starting around April 2014.

Putting aside the ambiguity of terms such as "generally omitted" and "inadvertent," it appears that ICEWS is doing very little de-duplication, and this document explicitly states that events reported by multiple sources that are subscribers to wire services will be repeated. It is possible that repetition also occurs within a publisher, depending on how literally they mean "same headline," since often the headline of a repeated story will be changed with "Revised" or "Update." This approach will also tend to increase the apparent precision of the coder: wire service stories tend to be easier to code than local stories, and not removing duplicates artificially inflates the precision by counting the same correctly coded story multiple times.

In terms of further research along these lines, the really interesting comparison, ICEWS against the OEDA real-time Phoenix data, will need to wait for another year or more until we have a significant overlap given the one-year ICEWS embargo. The second thing that would be useful is looking at how data sources compare on covering *internal* events, since that is the major focus of most current event-based research. A final point which would probably be useful, though it is less clear exactly how this should be done, is developing metrics for comparing general event data sets with event-specific sets (e.g. ACLED and the Uppsala UCDP data for conflict; SCAD for protest; GTD for terrorism).

Ultimately, however, multiple news sources—a recent development in event data in general—have both advantages and disadvantages, and these vary with the source. We need to systematically assess the degree to which specific sources are adding information rather than merely adding noise and inconsistency, particularly when certain geographical areas are over- or under-sampled. This is an issue both across event data sets and within single data sets over time: the daily density of events in ICEWS, for example, differs by at least a factor of two for reasons that are almost certainly due to changes in the sources.

Alternatively, multiple sourcing may require more sophisticated filters at the original level of the texts to eliminate duplicates—or at least classify stories into sets of likely duplicates—prior to coding. This would also open the possibility of using multiple codings to improve the coding accuracy, that is, by coding all of the stories in such a set, then assigning the coding that occurs most often. This requires quite a bit more computing power than we currently use, however, and assessment of a new body of literature in computational text analysis.

5 Application 2: Tabari shallow parsing versus Petrarch full parsing on the KEDS Levant Data

The second application will compare the shallow-parsing approach used in the TABARI coder to the full-parsing approach used in the new PETRARCH-1 coder.[16]

TABARI (https://github.com/philip-schrodt/TABARI-Code) has an internal parser which takes the any English-language sentence as input, applies a relatively small number of rules to identify actors, compound phrases, and subordinate phrases, and then applies a set of around 16,000 verb phases (https://github.com/philip-schrodt/TABARI-Dictionaries) that were large developed for the coding of the Levant[17] to determine the event. The combination of the shallow-parsing approach and the fact that TABARI is written in C/C++ means that it is very fast, coding around 5,000 sentences per second.

PETRARCH (https://github.com/openeventdata/petrarch) has been developed as a successor to TABARI and works with fully-parsed sentences in the Penn Treebank format; in most applications, including the example here, these are produced using the Stanford Core NLP system (http://nlp.stanford.edu/software/corenlp.shtml). PETRARCH also uses a more robust dictionary format (https://github.com/openeventdata/Dictionaries) that has been partly organized around the WordNet synonym sets (https://wordnet.princeton.edu/) and the general-purpose list of country-level names and adjectives, geographical locations and political actors in `Country-Info.txt` (https://github.com/philip-schrodt/CountryInfo-1).

Full-parsing should produce more accurate data, but this comes at a considerable computational cost: on my hardware, the Java-based Core NLP produces Treebank parsed sentences at a rate of about 10 sentences per second[18] and the Python-based PETRARCH codes at a rate of only about 300 sentences per second. In the contemporary computing environment of widely available cluster and cloud computing resources, in principle the speed on individual computers should not be a constraint but, realistically, moving programs into cluster or cloud environments is invariably more involved than it appears at first, so if it was possible to use the faster and simpler TABARI , this would be preferable.

This comparison uses the recently-updated TABARI -coded Reuters and Agence France Presse (AFP) data set available at http://eventdata.parusanalytics.com/data.dir/levant.html;

[16] As noted later, in the OEDA PETRARCH-1 will shortly be superseded by the more sophisticated PETRARCH-2 coder, though the comparisons with TABARI are likely to still be similar. In this discussion "PETRARCH" refers to PETRARCH-1.

[17] Specifically Egypt, Israel, Jordan, Lebanon, Palestinians and the Palestinian Authority, and Syria.

[18] This was achieved by simultaneously running two 2Gb instances of the program on a quad-core iMac with 3.2Ghz Intel i5 cores: your mileage may differ. The Core NLP team has recently released a much faster, if slightly less robust, "shift reduce parser" (http://nlp.stanford.edu/software/corenlp.shtml#srparser) but the parsing here used the default "ParseAnnotator" model.

the Reuters series covers 15 April 1979 to 30 March 2015; the AFP data analyzed here cover 5 May 1991 to 30 March 2015.[19] The same texts—which only involve "lede" sentences—were then processed with the Core NLP/Petrarch system to generate a parallel series covering the same period. In light of the comments made in Section 2 on the unintended consequences of One-A-Day filtering, the full sets of events were used, rather than filtered. The set of countries analyzed was ISR, PSE,[20] LBN, EGY, SYR, JOR, USA, IGO, TUR, FRA, GBR, DEU; note that due to the search terms used to generate the initial texts, in the cases of USA, IGO, TUR, FRA, GBR, DEU these are only stories that also mentioned one of the Levant countries in the headline or first paragraph—"HLEAD"—of the story. The patterns in the AFP and Reuters series were very similar, so only the longer Reuters series is discussed here.

Table 3: Twenty dyads with highest weighted average correlation

ISR-LBN (9871) 0.7684	ISR-PSE (39655) 0.7554	JOR-TUR (75) 0.6798	EGY-SYR (924) 0.6798
ISR-USA (14722) 0.6689	JOR-FRA (188) 0.6567	SYR-JOR (591) 0.6503	EGY-TUR (251) 0.6327
EGY-USA (4727) 0.6318	LBN-USA (3300) 0.6096	ISR-EGY (5399) 0.608	SYR-USA (4301) 0.6054
ISR-GBR (1075) 0.5929	ISR-IGO (4480) 0.5923	PSE-USA (10980) 0.5914	EGY-JOR (737) 0.583
JOR-USA (2435) 0.5724	EGY-FRA (594) 0.5718	ISR-JOR (2424) 0.5682	ISR-FRA (1068) 0.558

Table 4: Twenty dyads with lowest weighted average correlation

LBN-DEU (219) 0.403	PSE-IGO (5414) 0.3631	PSE-JOR (4632) 0.3577	USA-DEU (282) 0.3505
IGO-TUR (243) 0.3361	FRA-GBR (90) 0.3343	ISR-DEU (599) 0.3326	LBN-JOR (166) 0.321
USA-FRA (492) 0.3146	IGO-GBR (335) 0.3111	TUR-DEU (38) 0.2983	PSE-LBN (4574) 0.2861
IGO-FRA (384) 0.2549	LBN-TUR (61) 0.248	PSE-FRA (3532) 0.2473	PSE-SYR (3654) 0.2237
IGO-DEU (106) 0.2235	PSE-GBR (3445) 0.1275	PSE-TUR (2964) 0.0919	PSE-DEU (2973) 0.0701

As with the analysis in Section 4, the highest correlations are generally in dyads with large numbers of observations, and the low correlations are concentrated in a small set of problematic actors, specifically PSE (40% of the cases) and IGO (25% of the cases) and low-frequency cases such as FRA-GBR, TUR-DEU and LBN-TUR. As shown in Table 5, there is a strong, though by no means perfect, correlation between the total number of events in a dyads and the weighted correlation for that dyad.[21]

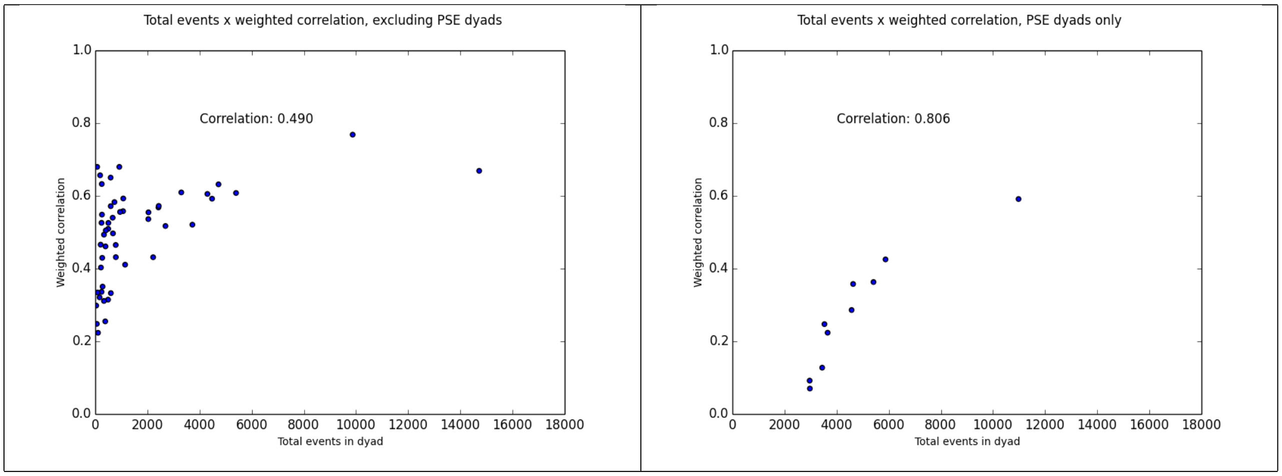
The dramatic difference in the PSE coding can be further seen by looking at the Class 4 (material conflict) codings for individual dyads in Table 6. While the Tabari totals (second number) are plausible for high-frequency dyads such as ISR and LBN, where in fact there was a great deal of Palestinian military activity, this is not the case for actors such as USA, FRA, GBR (United Kingdom) and DEU (Germany), where any activity would be limited

[19] The "AFP" dataset at `http://eventdata.parusanalytics.com/` is a composite of AFP and Reuters with Reuters used to fill in periods when AFP was not available.

[20] This also includes the non-ISO-3166 PAL code used to refer to Palestinians and Palestinian groups prior to the establishment of the Palestinian Authority.

[21] The graph does not include dyads where counts were so low that the weighted correlation was not computed because some pentaclasses had no observations: these were JOR-DEU, TUR-FRA, TUR-GBR, FRA-GBR and FRA-DEU.

Table 5: Total counts by weighted correlation by dyad.



to a relatively small number of terror attacks in the early part of the sequence. Here the Petrarch numbers look far more plausible, and we can conclude that Petrarch is almost certainly picking up dramatically fewer false positives than Tabari , at least for this case.

Table 6: Class 4 totals for PSE dyads listed in the order (Petrarch , Tabari)

ISR: 5373, 10192	LBN: 242, 1660	EGY: 125, 1228	SYR: 43, 1131
JOR: 67, 1189	USA: 220, 1375	IGO: 77, 1171	TUR: 4, 1084
FRA: 33, 1098	GBR: 30, 1107	DEU: 10, 1091	

The situation is further complicated, however, when we look at Table 7, which shows scatterplots of the total counts (the sum of all weeks) by dyad: for purposes of scaling, the ISR-PSE dyad is not plotted,[22] and also note that the y-axis scale in the figures for Classes 3 and 4 are twice- and four-times the x-axis scale. When the PSE dyads are treated separately, there is almost a perfect linear relationship: all of these have an $r > 0.99$ except for Class 0 ($r = 0.97$) and Class 4 with ISR-PSE excluded ($r = 0.93$). While PETR very consistently is coding fewer events than TAB in Classes 0, 3, and 4, over the 35-year period there is an almost perfect linear relationship across dyads in the total yield of events in the two systems.

What is remarkable—which is to say, worrisome—about the figures in Table 7 is that the line of the PSE cases is pretty closely displaced by a constant (that is, the slopes are essentially the same): this about 400 in Classes 0 and 3, 750 in Class 1, 150 in Class 2, and 1000 in Class 4 (the 1000-event offset is also evident in Table 6). While the possibility of some sort of processing artifact cannot be excluded (though why PSE and not LBN, another high-frequency case?), *something* appears to be generating a roughly constant number of additional events—in all likelihood false positives—across PSE interactions with all of the other actors, and doing so in a fashion that produces the same pattern in all of the pentaclasses, but with different offsets depending on the class.

[22] The ISR-PSE dyad has the following counts in the order (Petrarch ,Tabari):
Class 0 (4289,2329) Class 1 (3322,6141) Class 2 (1390,1447) Class 3 (1913,3259) Class 4 N=(5373,10192)

My guess is that this probably is probably due to the actor dictionaries rather than the shallow-vs-full parsing. The TABARI Levant dictionaries were developed over a period of some twenty years and there were ample opportunities for weird stuff to get in there, whereas the PETRARCH coding uses the much more general `Country-Info.txt` dictionaries which were largely assembled from sources such as the CIA World Factbook rather than the adjustments by individual coders. I have not had the opportunity to explore this yet but the eventual answer is likely to be interesting.[23]

Beyond this one puzzling anomaly, it is probably safe to conclude from this exercise that PETRARCH is substantially reducing false positives overall—which is what we expected the transition to full-parsing to do—though to firmly establish that would require some time-consuming case-by-case comparisons. In the high-frequency cases, the TABARI /PETRARCH correlations on aggregate counts are at levels comparable in inter-coding agreement for human coders, but these correlations drop for the cases with lower frequencies.[24] My take-away is that while Core NLP/PETRARCH is decidedly more difficult to work with—between formatting and the coding itself, that PETRARCH coding took me the better part of three days—we are getting a reasonable return on this. That said, it is unlikely that results using data from full-parsing coders such as ACCENT and PETRARCH will give a dramatically different view of the world than we had from earlier sparse-parsing coders, at least when one is looking at high-frequency dyads such as ISR-PSE and ISR-LBN in long time series, where much of the focus of event data research has been.

6 Application 3: Parsing compared to "bag of words" methods

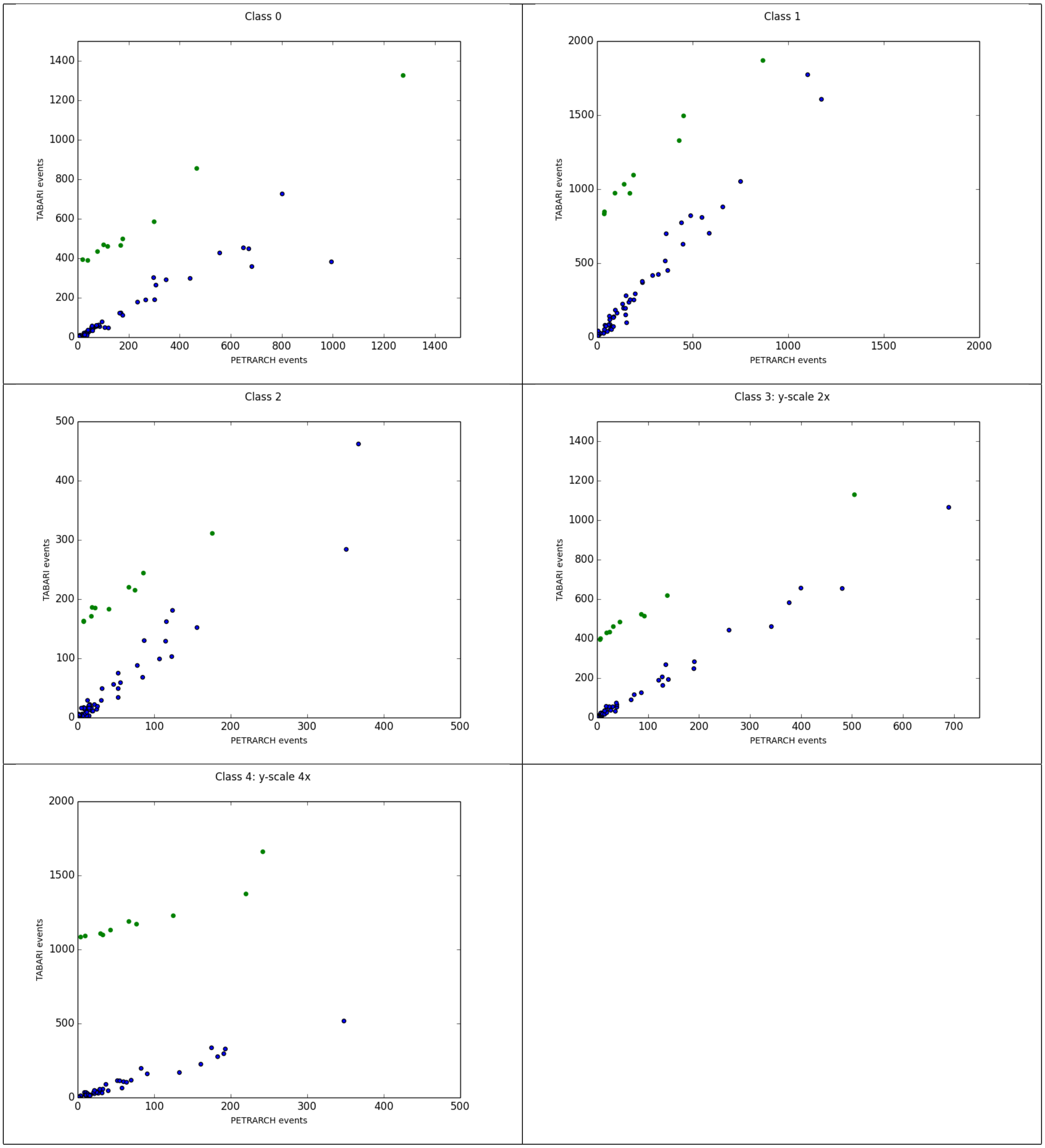
The full parsers used in ICEWS and PETRARCH are computationally intensive and, given the wide variety of ways that a particular political interaction might be described, potentially quite brittle. Furthermore, as noted earlier, these methods may be far more detailed than is needed: most extant uses of event data aggregate them to a high level such as pentacode classes, and event data tends to be highly reciprocal—in some cases reflecting events on the ground, in others simply the nature of news coverage—and consequently directionality may be providing relatively little information. The question, then, is how much is actually being gained from the parsing?

In order to test this, I used two "bag of words" approaches to extract events from lede sentences without using any parsing. The target texts were the ledes used to generate the AFP and Reuters Levant sequences; to reduce the amount of processing I just looked at the period 2005-2014; this has 114,220 ledes for AFP and 44,098 ledes for Reuters; the

[23] One possibility is confusion created by the pre-Oslo PAL code versus the post-Oslo PSE code, which may be reducing PETRARCH codings in the 1979-1996 segment of the series. But one would expect this to generate a proportional difference, not a constant one. If this is the case, it should be apparent in a comparison of the sequences by year, as was done in Section 4

[24] though are still almost always statistically significant, to the extent anyone cares about significance tests in this design, and one probably shouldn't

Table 7: Total counts by dyad, excluding ISR-PSE. Green markers are dyads involving PSE; blue are all other dyads.



AFP and Reuters texts are completely distinct during this period. The “gold standard” was the PETRARCH coding, so the comparison here is whether these methods can duplicate PETRARCH, not whether they can duplicate human coding.

6.1 Extracting actors using simple patterns

Actors and agents were extracted by simply looking for phrases from the `CountryInfo`-based dictionary.[25] The following algorithm was used

1. Identify all of the phrases in the sentence that correspond to either a country or agent code in the dictionary
2. The dyad for the lede is the two codes that occur most frequently. If two codes have the same frequency—this is quite common, particularly in the second-most-common frequency—use the code which occurs earliest in the sentence
3. If an actor code is followed by an agent code, combine the actor and agent
4. If multiple codes are found, use these—in other words, the system prioritizes inter-actor dyads. If only a single code is found but occurs two or more times, and one or both of those actor codes have agent codes attached, generate a single-actor event.

This is actually not dramatically different than the approach used in TABARI except that TABARI anchored the search on a verb, initially looking after this for the target.

Tables 8 and 9 look at the results of this approach for the AFP and Reuters cases where the PETRARCH coding had identified events; the N refers to the total number of actors identified in the event, or twice the number of events (that is, some ledes generate multiple events). The *alpha* comparisons check for the number of times codes identified in the PETRARCH coding also occur in the pattern-based coding for the 3-character (actor) and 6-character (actor and optional agent) codings. *none found* is the number of cases where PETRARCH found an event but none was found based on the dictionaries: These would be cases where the actors had been identified by PETRARCH using either the names of leaders or the IGO and militant group dictionaries that were also used in the PETRARCH dictionaries. The *order* rows look at whether both the codes correspond and the order corresponds: the divisor for computing the percentage is thus $N/2$.

The results are strikingly similar across the two data sets. At the non-directed primary actor level, there is a 82% agreement between PETRARCH and the dictionary-only method, and this is with dictionaries based almost only on geographical regions. This agreement drops by more than half, however, once the primary agents are considered, so that the syntactical parsing is giving considerable leverage here. This is also true for the comparison of the

[25] `http://eventdata.parusanalytics.com/software.dir/dictionaries.html` I used only the country and geographical place names, not the leaders, except that heads of state and government were added for the US and Iran. I also included vocabulary for the United Nations from the PETRARCH international actors dictionary. To standardize the system to 3-character actor codes, ‘PSEGZA’ (Gaza) was converted to ‘GZA’ and ‘IGOUNO’ (United Nations) was converted to ‘UNO.’

Table 8: AFP results [N = 86,450]

alpha-3	70271	81.29%
alpha-6	30367	35.13%
none found	5968	6.90%
order-3	11705	27.08%
order-6	0	0.00%

Table 9: Reuters results [N = 31,256]

alpha-3	25538	81.71%
alpha-6	10212	32.67%
none found	2314	7.40%
order-3	4184	26.78%
order-6	0	0.00%

directed dyads, where there is only about a 33% agreement, though this number is biased downwards somewhat by the fact that "symmetrical" events of the form "X visits Y" and "Y hosts X" would automatically get at least one of the combinations incorrect since the dictionary-based algorithm only generates a single directed event. In no cases in either data set did the dictionary-based method get the ordering correct.[26]

One interpretation of this would be that actor identification in the "classical" event data sets such as WEIS and COPDAB was, in fact, a rather simple problem, and this would also explain both the high inter-coder reliability scores reported in those studies, as well as the relatively high accuracy scores reported in some of the early machine-coding work, including [Schrodt and Gerner, 1994, King and Lowe, 2004], which worked in pre-CAMEO frameworks. The challenge, it appears, comes in as one tries to get more nuance out of the stories. Conversely, if one is just looking at relatively high levels of aggregation, pretty much anything with decent generic dictionaries is going to work.

Classifying events

To classify events without parsing, I used support vector machines (SVM) on a vector of the most common 2000 words found in the first 30,000 lead sentences following the elimination of

- Capitalized words: this is for purposes of generalization as it eliminates bias that would otherwise be introduced because certain actors are more likely to be engaged in certain categories of behavior; this also removes acronyms;
- Numbers;

[26] and frankly, this looks like it must be a bug but the code is quite simple and parallels that used to compute the 3-character case, so it is difficult to see what a bug could occur, particularly since we know from the $alpha - 3$ row that there is a pretty good agreement in the 6-character codes generally.

- 1- and 2-letter terms—this mostly removes common stop words such as 'a', 'to' and 'be' and residual 's' and 't' after the removal of apostrophes by the routine which removes punctuation;
- Words in a 580 word stop list largely derived from the Python "Goose" module plus some news-story-specific vocabulary such as "stories," "leading," "press," and "verified": this file is available on request.

Side Note: There were surprisingly large differences in these lists between AFP and Reuters: 517 distinct words for a list of first 2000 words from first 10,000 ledes, and 682 words from a list of first 30,000 ledes. Note that because the total size of each list is fixed, once one eliminates the words in common, the number remaining is the same for the two sets even though the words, obviously, are different.

The SVM used is the default `LinearSVC` in $scikit - learn$[27] which implements a set of "one-vs-the-rest" classifiers.[28] The models are assessed using a standard test/training-sample scheme in two configurations:

- Train on the first part of the data, using sufficient data (about a year) that there are 400 instances of each category; the non-events are sampled during this period with a random probability of 7.5%, also with a maximum of 400 instances. "400" is used to [*very* optimistically] approximate the number of "gold standard" cases one might generate through human coding if this were used rather than PETRARCH. The test cases are the remainder of the data.
- Train on roughly the first half of the data, specifically the period 2005-2009, and test on the remainder of the data, with non-events now sampled at a 25% rate, which gets the number of Category 5 cases in roughly the same range as Categories 0 and 1.[29]

In the test cases I also distinguished between cases where no actor dyad was found in the pattern-based search (designated category 5) and cases where a dyad was found but PETRARCH didn't code an event (designated category 6).

In the 400-training-case situation, Tables 10 and 12, AFP and Reuters produce quite similar results. In both instances, the training set classification is nearly perfect, which is what one would expect to see given that the dimensionality of the feature vector is the same size as the number of training cases (2000) and PETRARCH rarely produces pathological cases where nearly identical sentences receive distinct pentaclass codes. In the AFP case, the accuracy

[27] `http://scikit-learn.org/stable/modules/svm.html`

[28] From the documentation:

```
LinearSVC(C=1.0, class_weight=None, dual=True, fit_intercept=True, intercept_scaling=1,
loss='squared_hinge', max_iter=1000, multi_class='ovr', penalty='l2', random_state=None,
tol=0.0001,verbose=0)
```

[29] This balancing appears to be important: I originally just sampled these cases at 7.5% but this under-sampled category 5 in the training cases, which reduced the test cases accuracy in categories 5 and 6 to 24.6% and 14.7% respectively, and the overall accuracy to 34.8% Over-sampling non-events at 50% had the opposite effect with the category 5 and 6 accuracy being too high—around 80%—at the expense of accuracy in the other categories.

in the test set is in a fairly narrow range from 31.8% for Category 0 (comments) to 42.4% for Category 3 (verbal conflict); the Reuters accuracy is generally about 4% lower, except for the non-event category 5, which is quite substantially higher at 50%. That accuracy then leads to the overall Reuters accuracy being 37% compared to the 33.5% for AFP despite the lower accuracy on the true event categories.

The "overall accuracy", however, assumes that all of the category 6 cases are actually category 5—that is, these are non-events even though they contain actor-dyads—but category 6 is probably picking up quite a few cases that Petrarch mis-coded as false negatives; this would also be consistent with the fact that the category 6 accuracy is 10% lower than the category 5 accuracy. It also, of course, assumes that the Petrarch classifications are all correct, which we know not to be the case. Adjusting for these factors, the overall accuracy of this approach would probably be somewhere in the high 30% range for AFP and low 40% for Reuters.

Despite this convergence in the small training set case, the two news sources diverge quite substantially when a 50/50 training/test split is used (Tables 14 through 16). In the AFP case, the classification accuracy in the training set drops to 63% and shows substantial variation between the categories. The Reuters training-case accuracy, in contrast, is generally in the mid-70% level, with 77.8% accuracy overall, and less variation except for the very high 87.3% accuracy in category 1 (verbal cooperation). In the test set, however, this pattern is reversed, with higher accuracy in the AFP case, which increases to above 50%, again with the exact number unknown since some of the "errors" in category 6 probably correspond to false negatives in the Petrarch coding. In the AFP case, accuracy on category 1 (verbal cooperation) is a remarkably high 68.9%, and accuracy for non-events where no actor-dyad was found is 65%. The remaining categories are in the 40% range except for the low-frequency category 2 (material cooperation), which is below 20% and also has the lowest accuracy in the training set.

Reuters, in contrast, generally has about 6% lower accuracy than AFP in the test cases, though it actually does slightly better (21% vs 18.7%) on the lower-frequency category 2, and has an unusually high drop compared to AFP of almost 20% on category 3. The most likely explanation for this pattern is that over time Reuters has been undergoing some shifts in editorial styles, and possibly was enforcing a style sheet more strictly during the 2005-2009 period than during the 2010-2014 period.[30]

In both of the data sets, the classification errors have somewhat more of a pattern in the 50/50 test than in the 400-training case. In the latter (particularly for AFP, Table 11), the classification errors are surprisingly evenly distributed across categories, again possibly an effect of the low ratio of training cases to the size of the feature vectors. In the 50/50 case (Table 15), there is a fairly strong tendency for categories 0 and 1 to mis-classify into each other—and in the older "Quad category" scheme these are the same class—and there are few misclassifications into the low-frequency category 2, but in particular the two "material"

[30] Somehow I managed to acquire a copy of the book-length Reuters style sheet in the 1990s: As would be expected of an elite British institution—or is "elite British" an oxymoron when applied to journalists?—it is quite amusing in places, and one can almost smell the gin-and-tonic, but it is totally scattered and we couldn't figure out how to get anything useful from it for coding purposes.

categories, 3 and 4, mis-classify into categories 0 and 1 more frequently than they mis-classify into each other.

Table 10: Training set: AFP first 400 cases in each category
Correct: 98%

	0	1	2	3	4	5
0	387	8	3	1	2	0
1	2	397	0	1	2	0
2	0	1	393	2	4	0
3	1	0	0	396	4	0
4	0	1	4	5	392	0
5	0	0	0	0	0	400

Table 11: Test set: AFP remaining cases
Correct: 33.48%

	0	1	2	3	4	5	True cases	Category accuracy
0	3860	1183	1555	1476	1496	2538	12108 (10.50%)	31.88%
1	2558	5466	2325	1631	1855	2623	16458 (14.27%)	33.21%
2	413	276	1316	404	438	713	3560 (3.09%)	36.97%
3	816	383	562	2554	667	1037	6019 (5.22%)	42.43%
4	696	431	948	945	2948	1824	7792 (6.76%)	37.83%
5	3682	2232	4231	3065	4958	10723	28891 (25.06%)	37.12%
6	5699	3661	5905	5460	8020	11737	40482 (35.11%)	28.99%

Table 12: Training set: Reuters first 400 cases in each category
Correct: 97.8%

	0	1	2	3	4	5
0	380	10	2	5	4	0
1	3	396	0	1	1	0
2	3	0	396	1	0	0
3	4	4	1	391	1	0
4	3	1	5	1	391	0
5	0	1	0	0	0	399

While most of these accuracy rates are relatively low, they are still obviously far above chance, so quite a bit of accurate classification is occurring here. Curiously, there is relatively little relationship between the frequency of the cases in the data and the classification accuracy: for example in the 50/50 case for AFP category 0 has about twice the frequency as category 3 but both have 47% accuracy; a similar relationship is found between categories 1 and 5 at about 65% accuracy.

Table 13: Test set: Reuters remaining cases
Correct: 37.07%

	0	1	2	3	4	5	True cases	Category accuracy
Test set 0	1291	339	740	656	418	952	4396 (11.54%)	29.37%
1	841	1123	725	575	383	718	4365 (11.45%)	25.73%
2	220	108	530	169	126	288	1441 (3.78%)	36.78%
3	194	99	251	651	204	261	1660 (4.36%)	39.22%
4	221	72	399	325	841	476	2334 (6.12%)	36.03%
5	1362	454	1709	1041	1198	5808	11572 (30.36%)	50.19%
6	1727	905	2164	1735	1926	3885	12342 (32.39%)	31.48%

Table 14: Training set: AFP 2005-2009
Correct: 63.12%

	0	1	2	3	4	5	Category accuracy
0	5120	1424	161	397	426	1585	56.18%
1	893	10656	149	216	236	1124	80.28%
2	344	537	923	105	228	719	32.32%
3	557	447	72	2561	255	696	55.82%
4	403	498	108	303	3479	1323	56.90%
5	1225	1732	259	613	1202	8473	62.74%

Table 15: Test set: AFP 2010-2014
Correct: 53.4%

	0	1	2	3	4	5	True cases	Category accuracy
0	1950	633	93	239	209	1012	4136 (11.27%)	47.15%
1	426	3371	98	168	131	699	4893 (13.33%)	68.89%
2	183	196	207	50	94	374	1104 (3.01%)	18.75%
3	265	251	29	954	143	365	2007 (5.47%)	47.53%
4	212	260	58	192	945	613	2280 (6.21%)	41.45%
5	856	769	246	550	839	6046	9306 (25.35%)	64.97%
6	1830	2058	353	994	1625	6119	12979 (35.36%)	47.15%

Table 16: Training set: Reuters 2005-2009
Correct: 76.85%

	0	1	2	3	4	5	Category accuracy
0	2051	259	47	80	132	307	71.31
1	177	2650	15	30	28	134	87.34
2	55	53	775	10	34	79	77.04
3	114	61	15	820	54	103	70.27
4	106	48	24	42	1581	196	79.17
5	345	234	55	88	246	2651	73.25

Table 17: Test set: Reuters 2010-2014
Correct: 45%

	0	1	2	3	4	5	True cases	Category accuracy
0	1098	491	163	147	178	583	2660 (11.14%)	41.28%
1	418	1274	130	74	142	441	2479 (10.38%)	51.39%
2	155	127	175	55	74	249	835 (3.50%)	20.96%
3	206	127	63	257	98	235	986 (4.13%)	26.06%
4	147	107	76	95	459	327	1211 (5.07%)	37.90%
5	997	866	549	314	776	4370	7872 (32.97%)	55.51%
6	1480	1114	483	506	1132	3120	7835 (32.81%)	39.82%

I would draw three major conclusions from this exercise

1. While the SVM classifications diverge substantially from those of Petrarch in the test cases, and I see little reason to employ an SVM classifier over Petrarch in situations where dictionaries are available, they are certainly not random and would probably provide a useful signal for behaviors where dictionaries were not available, for example in some of the categories identified in the topic modeling exercise in [Schrodt and Bagozzi, 2013] or in coding specialized categories such as protests or IED use where the CAMEO dictionaries are not well developed.
2. It is almost certainly the case that a combination of phrase-based actor identification and SVM classifiers could be used to identify texts that might contain events not covered in the existing dictionaries, which would be a major resource for new systems for automatically developing event dictionaries.
3. Given that these are the two major English-language wire services, the differences between AFP and Reuters were surprisingly large, as was the difference in Reuters over time, and the differences between these traditional wire services and newer media—for example CNN, Xinhua and al-Jazeera, to say nothing of local sources such as capital-city newspapers—is likely to be even greater. Source-specific effects may be considerably higher than we have been assuming.

As is usually the case in machine learning schemes, there are roughly a zillion parameters with which one could experiment to improve the accuracy: balancing the samples clearly makes a difference, and I simply used the defaults on the kernel and several parameters of the `LinearSVC` procedure. In addition, I have looked only at the lede sentences, and one could probably get a noticeable gain in accuracy by coding, for example, the first four sentences, though one might also want the possibility—as Petrarch does routinely, in around 25% of ledes in the Levant data—of coding multiple events from a sentence. This is somewhat awkward to do in SVM but is more straightforward in some other classification models such as LDA.

7 Future developments

In earlier versions of this paper presented in April and June of this year, I closed the paper with a "what is to be done" section which elaborated on an earlier agenda I suggested at `https://asecondmouse.wordpress.com/2015/03/30/seven-observations-on-the-newly-released-icews-data/`. Indicative of just how quickly this technology is moving, parts of that agenda have now been implemented and, most important, we have now secured a major source of funding for a stable development path for open-source event data for at least the next three or four years. Consequently I'm going to reorient that discussion from "what is to be done" to "what is being done."

7.1 PETRARCH-2

During the summer of 2015, Clayton Norris, an intern at Caerus Associates who is a dual major in computer science and linguistics at the University of Chicago, re-wrote the core algorithms of Petrarch so that they look like something from computational linguistics rather than from a couple of NLP hackers coming out of political science; this also increased the speed of the program by a factor of 10 and involved substantial changes to the event dictionaries. This version is not quite in release yet and we haven't done full testing to compare these results with the version of Petrarch we used over the past year but this should be done in the next few weeks: initial results indicate it should produce significant improvements in actor assignment, particularly for targets.

7.2 Mordecai

Over the spring and summer of this year, Andrew Halterman and John Beieler at Caerus Associates also produced the `Mordecai` geolocation program,[31] which uses an ensemble of several existing open source geolocation and NLP tools to produce a system which seems substantially more accurate than anything we have to date, and at the very least dramatically reduces instances where an event in Moscow, Russia is geolocated to Moscow, Idaho, or Damascus, Syria to Damascus, Virginia. Yes, the existing systems are that bad. `Mordecai` is not the last word on geolocation, but a major improvement over anything available six months ago.

7.3 NSF RIDIR funding

In the past month, a consortium of university and private sector research centers have secured a large, three-year grant from the U.S. National Science Foundation's Resource Implementations for Data Intensive Research in the Social Behavioral and Economic Sciences (RIDIR) program.[32] While the precise agenda for this effort is still evolving, in part because of progress made between the time the proposal was submitted and where we are now, this should produce at least the following:

- Stable support for the real-time Phoenix data system
- Several long-time-frame databases using data from Lexis-Nexis: the consortium has acquired a license to download very large volumes of text from LN and Andrew Halterman has written an LN-approved program—appropriately named after the Roman goddess of sewers—to do this
- Native language coders, using the CoreNLP/Petrarch-2 framework, in at least Spanish and Arabic, with French and Chinese as an additional possibilities. Despite the problems of multiple sources, work to date, particularly Osorio's work on using a

[31] `https://github.com/caerusassociates/mordecai`

[32] `http://www.nsf.gov/awardsearch/showAward?AWD_ID=1539302&HistoricalAwards=false`

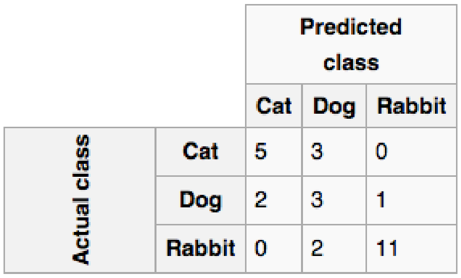
Spanish-language automated coder to analyze the drug war in Mexico [Osorio, 2014], strongly suggests that some non-English sources are going to contribute very substantially to event coverage.

- Development of systems for automated actor and event dictionary development, and possibly a user-friendly open collaborative platform for both dictionaries and gold standard cases.
- Extensions of CAMEO—which was developed to study mediation [Schrodt and Gerner, 2004], not as a general-purpose political event ontology—to cover a broader set of political events (e.g. routine democratic processes such as elections, parliamentary coalition formation, and legislative debate) as well as providing a better system for coding ethnic and religious groups.
- Development of event-specific modules for PETRARCH-2 which can extract features beyond simple source-target-event triples. Protests will probably be the first topic we work on.

While the ICEWS project invested a very substantial amount of funding in event data development—multiple tens of millions of dollars—and after an extended delay made quite a few of those resources public, it was hampered by using the twentieth-century model of proprietary development, which massively inflated the cost of the project, as well as having to deal with the inescapable U.S. Department of Defense requirements for the nearly continuous production of incoherent and unreadable PowerPoint^{TM} slides. The OEDA-RIDIR project, in contrast, focuses working in twenty-first century open source software ecosystems and has the lean and agile flexibility of an NSF-funded research program. Fully seven years elapsed between the beginning of the ICEWS and the public release of its data; in contrast PETRARCH-1 and the Phoenix data pipeline were written and made operational in a period of about nine months, and PETRARCH-2 in a period of four months. Coupled with the ever-increasing availability of news reports on the web, the future of event data looks very promising.

Appendix: The Classification/Confusion Matrix

A classification matrix—apparently the preferred term nowadays, or at least the term that has won the Wikipedia battles, is “confusion matrix”[33]—is a contingency table with the true values of the cases as the rows and the “predicted”—in our case, the resulting codes—as the columns. Figure 6 shows a simple example.



		Predicted class		
		Cat	Dog	Rabbit
Actual class	Cat	5	3	0
	Dog	2	3	1
	Rabbit	0	2	11

Figure 6: Classification/confusion matrix (Source: Wikipedia)

In the case of event data, the matrix would be constructed from a sample of texts (for most automated coding systems, sentences), the rows would be the “gold standard” codes, presumably determined by human coders, and the columns would be the code that was assigned by the system. In addition to rows and columns corresponding to the codes, we would also have a final row for texts which should not receive a code at all—in practice, these will have a very large number of cases—and a final column for cases where the case should have been assigned a code, but was not. For our purposes, it is easier to standardize these by the sample size, so the row entries will sum to 1.0: that is, any text will either be assigned a code, or not coded at all. The classification matrix C is therefore

> $C_{i,j} =$ probability that a text in gold standard category i will be coded as category j, plus a probability that it will not be coded at all

In a good coding system, we would expect the largest elements of this matrix to be on the main diagonal, and the more likely errors to be “near misses”: Schematically, the classification matrix looks like Figure 7 where the highest loadings are for accurate classification (darkest squares) and diminishes as one gets away from the main diagonal. While SVM classifiers do succeed in putting the largest numbers of cases on the main diagonal, except for categories 0 and 1, we are *not* seeing this idealized pattern in the mis-classifications.

The situation is, of course, a little more complicated than this, particularly when human coding and automated coding are compared. If properly trained, motivated and alert, humans rarely assign actors incorrectly since distinguishing subjects and objects is a fundamental language processing task humans are extremely good at. Machines, in contrast, make this error frequently, particularly on oddly constructed sentences or in situations where the object is ambiguous.

[33] “error matrix” is another possibility though unlikely to catch on since this phrase is already widely used in statistics for something totally different.

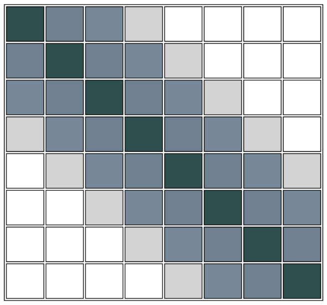


Figure 7: Classification matrix

I would like to say that humans are less likely to make crazy misclassifications of the event types than machines, but over the years I've seen human coders make a lot of really weird interpretations, and with contemporary automated systems using full parsing and very large dictionaries, I'm not entirely sure this is still true, and this is a particularly difficult issue when dealing with large coding teams where the level of expertise and commitment varies substantially. The point where humans are most likely to err is to miss events entirely.

The classification matrix will also differ, of course, depending on the source texts. If the classification matrix were known, and consistent in a large sample, it provides a nice statistical correction is one is using count data. Let g_i be the vector gold-standard classification of the texts and o_i be the vector of classification observed from the coding. By definition, $o = Cg$ so applying the correction $o' = C^{-1}o = g$ would presumably provide a much better estimate of the true counts than o. While this is a very simple correction to apply, I've never seen it done—including in my own work—since I don't know of any cases where C has been estimated with any degree of confidence.

Alas, unless I'm missing something, this absence of useable information for corrections also applies to the evaluation of the BBN ACCENT coder. For reasons that may have been linked to contractual requirements, BBN chose not to compute the classification matrix, but rather the "precision" for each *coded* category. That is, they selected on the dependent variable—the coded outcome—rather than sampling on the texts which were being coded, and consequently we only know that in the BBN sample (they don't provide the full matrix in any case) the columns, rather than the rows, would sum to 1.0. This is better than nothing, but is still an unfortunate design choice. Selection on the dependent variable usually is.

As the discussion in Section 2 indicates, to understand the DGP, we need to estimate the entire classification matrix for various coding systems. The reason this has not been done is that it involves human coding, which is complex, slow and expensive. What we need and do not have is a standard set of "gold standard" cases with known inter-coder reliability that can be shared without running into intellectual property issues, which could probably be provided by the Linguistic Data Consortium GigaWord news files. Then use a set of coders with documented training protocols and inter-coder performance evaluation, and do full accuracy assessments, not just precision assessments. Sustained human coder performance is typically about 6 events per hour—though probably much faster on true negatives, which are very common in a complete set of news texts—and we will need at least 10,000 gold standard cases, double-coded, which comes to a nice even $50,000 for coders at $15/hour, double this amount for management, training and indirect costs, and we're still at only $100,000, well within the range of a research grant in the social sciences, and we may be able to do some of this under the RIDIR funding.

References

[Andriole and Hopple, 1988] Andriole, S. J. and Hopple, G. W. (1988). *Defense Applications of Artificial Intelligence.* Lexington, Lexington MA.

[Choucri and Robinson, 1979] Choucri, N. and Robinson, T. W., editors (1979). *Forecasting in International Relations: Theory, Methods, Problems, Prospects.* W.H. Freeman, San Francisco.

[Davenport and Ball, 2002] Davenport, C. and Ball, P. (2002). Views to a kill: Exploring the implications of source selection in the case of guatemalan state terror, 1977-1995. *Journal of Conflict Resolution*, 46(3):427–450.

[Goldstein, 1992] Goldstein, J. S. (1992). A conflict-cooperation scale for WEIS events data. *Journal of Conflict Resolution*, 36:369–385.

[Kahneman, 2011] Kahneman, D. (2011). *Thinking Fast and Slow.* Farrar, Straus and Giroux, New York.

[King and Lowe, 2004] King, G. and Lowe, W. (2004). An automated information extraction tool for international conflict data with performance as good as human coders: A rare events evaluation design. *International Organization*, 57(3):617–642.

[King et al., 2013] King, G., Pan, J., and Roberts, M. E. (2013). How censorship in china allows government criticism but silences collective expression. *American Political Science Review*, 107(2):1–18.

[Merritt et al., 1993] Merritt, R. L., Muncaster, R. G., and Zinnes, D. A., editors (1993). *International Event Data Developments: DDIR Phase II.* University of Michigan Press, Ann Arbor.

[n.a., 2015] n.a. (2015). ICEWS coded event data read me.pdf. http://thedata.harvard.edu/dvn/dv/icews.

[O'Brien, 2010] O'Brien, S. P. (2010). Crisis early warning and decision support: Contemporary approaches and thoughts on future research. *International Studies Review*, 12(1):87–104.

[Osorio, 2014] Osorio, J. (2014). The contagion of drug violence. spatio-temporal dynamics of the mexican war on drugs. Annual Conference of the Midwest Political Science Association, Chicago.

[Raleigh et al., 2010] Raleigh, C., Linke, A., Hegre, H., and Karlsen, J. (2010). Introducing ACLED: An armed conflict location and event dataset. *Journal of Peace Research*, 47(5):651–660.

[Ruggeri et al., 2011] Ruggeri, A., Gizelis, T.-I., and Dorussen, H. (2011). Events data as Bismarck's sausages? intercoder reliability, coders' selection, and data quality. *International Interactions*, 37(1):340–361.

[Salehyan et al., 2012] Salehyan, I., Hendrix, C. S., Hamner, J., Case, C., Linebarger, C., Stull, E., Williams, J., White, K., and Williams, S. (2012). Social conflict in Africa: A new database. *International Interactions*, 38(4):503–511.

[Schrodt, 1994] Schrodt, P. A. (1994). Statistical characteristics of events data. *International Interactions*, 20(1-2):35–53.

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

[Schrodt and Bagozzi, 2013] Schrodt, P. A. and Bagozzi, B. (2013). Detecting the dimensions of news reports using latent dirichlet allocation models. International Studies Association.

[Schrodt and Gerner, 1994] Schrodt, P. A. and Gerner, D. J. (1994). Validity assessment of a machine-coded event data set for the Middle East, 1982-1992. *American Journal of Political Science*, 38:825–854.

[Schrodt and Gerner, 2004] Schrodt, P. A. and Gerner, D. J. (2004). An event data analysis of third-party mediation. *Journal of Conflict Resolution*, 48(3):310–330.

[Schrodt et al., 2009] Schrodt, P. A., Gerner, D. J., and Yilmaz, Ö. (2009). Conflict and mediation event observations (CAMEO): An event data framework for a post Cold War world. In Bercovitch, J. and Gartner, S., editors, *International Conflict Mediation: New Approaches and Findings*. Routledge, New York.

[START, 2012] START (2012). Global terrorism database. National Consortium for the Study of Terrorism and Responses to Terrorism.

[Taleb, 2010] Taleb, N. N. (2010). *The Black Swan: The Impact of the Highly Improbable Fragility*. Random House Digital, 2 edition.