

Automated Learning of Event Coding Dictionaries  
for Novel Domains with an Application to  
Cyberspace

by

Benjamin James Radford

Department of Department of Political Science  
Duke University

Date: \_\_\_\_\_

Approved:

\_\_\_\_\_  
Michael D. Ward, Supervisor

\_\_\_\_\_  
Scott de Marchi

\_\_\_\_\_  
Kyle Beardsley

\_\_\_\_\_  
Mark J.C. Crescenzi

Dissertation submitted in partial fulfillment of the requirements for the degree of  
Doctor of Philosophy in the Department of Department of Political Science  
in the Graduate School of Duke University

2016

ABSTRACT

Automated Learning of Event Coding Dictionaries for Novel  
Domains with an Application to Cyberspace

by

Benjamin James Radford

Department of Department of Political Science  
Duke University

Date: \_\_\_\_\_

Approved:

\_\_\_\_\_  
Michael D. Ward, Supervisor

\_\_\_\_\_  
Scott de Marchi

\_\_\_\_\_  
Kyle Beardsley

\_\_\_\_\_  
Mark J.C. Crescenzi

An abstract of a dissertation submitted in partial fulfillment of the requirements for  
the degree of Doctor of Philosophy in the Department of Department of Political  
Science  
in the Graduate School of Duke University  
2016

Copyright © 2016 by Benjamin James Radford  
All rights reserved except the rights granted by the  
Creative Commons Attribution-Noncommercial Licence

# Abstract

Event data provide high-resolution and high-volume information about political events. From COPDAB to KEDS, GDELT, ICEWS, and PHOENIX, event datasets and the frameworks that produce them have supported a variety of research efforts across fields and including political science. While these datasets are machine-coded from vast amounts of raw text input, they nonetheless require substantial human effort to produce and update sets of required dictionaries. I introduce a novel method for generating large dictionaries appropriate for event-coding given only a small sample dictionary. This technique leverages recent advances in natural language processing and deep learning to greatly reduce the researcher-hours required to go from defining a new domain-of-interest to producing structured event data that describes that domain. An application to cybersecurity is described and both the generated dictionaries and resultant event data are examined. The cybersecurity event data are also examined in relation to existing datasets in related domains.

I dedicate this thesis to my fiancé, Yaoyao, and to my parents, Heidi and James.  
I couldn't have done it without your love, support, and patience.

# Contents

<b>Abstract</b>	<b>iv</b>
<b>List of Tables</b>	<b>ix</b>
<b>List of Figures</b>	<b>x</b>
<b>List of Abbreviations and Symbols</b>	<b>xii</b>
<b>Acknowledgements</b>	<b>xiii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Cyberspace and Cyber Conflict . . . . .	2
1.2 Data on Cyberspace . . . . .	6
1.3 Event Data . . . . .	9
1.4 Conclusion . . . . .	10
<b>2 From Cybersecurity Documents to Event Data</b>	<b>12</b>
2.1 Data Creation in Political Science . . . . .	14
2.1.1 Manually Coded Datasets . . . . .	14
2.1.2 Machine Coded Datasets . . . . .	15
2.1.3 Machine-learned Datasets . . . . .	17
2.2 Raw Cybersecurity Data: Lots of Text . . . . .	17
2.2.1 Softpedia Corpus . . . . .	18
2.2.2 Technical Corpus . . . . .	18
2.3 CAMEO and the Need for Novel Dictionaries . . . . .	21

2.4	Preparation for Automated Dictionary Generation . . . . .	24
2.4.1	Cleaning Data, Sentence Parsing, and Named Entity Recognition	24
2.4.2	Triplet Extraction . . . . .	27
2.5	Conclusion . . . . .	28
<b>3</b>	<b>A Semi-Supervised Method for Generating Novel Event-Data</b>	<b>30</b>
3.1	Why automate dictionary creation? . . . . .	30
3.2	A method for creating and extending dictionaries . . . . .	31
3.3	Word Embeddings . . . . .	32
3.4	Preprocessing the corpus . . . . .	35
3.5	Learning the corpus . . . . .	37
3.6	Outlining an ontology . . . . .	38
3.7	Term extraction and post-processing . . . . .	40
3.8	Remaining challenges . . . . .	41
3.9	Conclusion . . . . .	44
<b>4</b>	<b>Introducing CYLICON, a Cyber Event Dataset</b>	<b>45</b>
4.1	Dictionary Generation Process . . . . .	46
4.2	A Cybersecurity Ontology . . . . .	47
4.3	CYLICON Event Data Overview . . . . .	48
4.4	Example Events from CYLICON . . . . .	53
4.5	Evaluating Event Data . . . . .	57
4.6	Remaining Challenges . . . . .	59
<b>5</b>	<b>Exploring CYLICON</b>	<b>64</b>
5.1	ICEWS and CYLICON . . . . .	64
5.1.1	Geographic distribution of events . . . . .	65
5.1.2	Events of Interest . . . . .	68

5.2	UCDP/PRIO and CYLICON . . . . .	71
5.3	Crime and CYLICON . . . . .	73
5.4	Conclusion . . . . .	78
<b>6</b>	<b>Automated Classification of Actors for Event Coding</b>	<b>80</b>
6.1	Approach . . . . .	81
6.2	Data . . . . .	82
6.3	Results . . . . .	86
6.3.1	Word2Vec Models . . . . .	86
6.3.2	Unsupervised classification . . . . .	89
6.3.3	Supervised classification . . . . .	91
6.4	Conclusion . . . . .	93
<b>7</b>	<b>Conclusion</b>	<b>95</b>
7.1	CYLICON and Event Coding . . . . .	95
7.2	The Future of Automated Event Coding . . . . .	96
7.3	Alternative Approaches to Cyber Data in Political Science . . . . .	97
7.4	Final Thoughts . . . . .	99
<b>A</b>	<b>Dictionary samples</b>	<b>100</b>
<b>B</b>	<b>Verb dictionary performance</b>	<b>103</b>
<b>C</b>	<b>CYLICON data</b>	<b>105</b>
<b>D</b>	<b>Scoring rules</b>	<b>123</b>
D.1	Actor scoring . . . . .	123
D.2	Event scoring . . . . .	125
	<b>Bibliography</b>	<b>126</b>
	<b>Biography</b>	<b>135</b>

# List of Tables

2.1	The 30 most frequent sources of news in the technical corpus. . . . .	21
2.2	Common triplet elements from cybersecurity corpora . . . . .	28
3.1	Skipgram example . . . . .	35
4.1	Seed phrases for verb dictionary . . . . .	61
4.2	Seed phrases for agent and actor dictionaries . . . . .	62
4.3	Seed phrases for issue dictionary . . . . .	62
4.4	Seed phrases for synsets . . . . .	63
5.1	ICEWS top dyads . . . . .	67
5.2	CYLICON top dyads . . . . .	67
6.1	ICEWS data summary . . . . .	82
6.2	Most similar words . . . . .	85
6.3	Algebra on word vectors . . . . .	85
6.4	Model performance in unsupervised task . . . . .	87
A.1	Sample of the verb dictionary . . . . .	101
A.2	Sample of the agents dictionary . . . . .	102

# List of Figures

1.1	The Phoenix pipeline . . . . .	10
2.1	Softpedia stories over time. . . . .	19
2.2	Technical corpus stories over time. . . . .	20
2.3	Softpedia stories in 2014 . . . . .	20
2.4	Common cybersecurity-related named entities . . . . .	26
3.1	Dictionary learning pipeline . . . . .	33
4.1	Cyber events over time . . . . .	48
4.2	Cyber events by type . . . . .	49
4.3	Spatial distribution of cyber actors . . . . .	50
4.4	Top country dyads . . . . .	51
4.5	Action accuracy by category . . . . .	58
5.1	CYLICON-EOI event count model . . . . .	69
5.2	Predicted CYLICON counts given EOI rebellion status . . . . .	70
5.3	CYLICON-UCDP/PRIO event count model . . . . .	73
5.4	Predicted CYLICON counts given UCDP/PRIO status . . . . .	74
5.5	UNODC and CYLICON marginal correlations . . . . .	75
5.6	Crime and CYLICON events . . . . .	77
6.1	Clustering of actors by country after multidimensional scaling . . . . .	87
6.2	Clustering of countries after multidimensional scaling . . . . .	88
6.3	True and estimated counts of actor’s countries . . . . .	88

6.4	Multiclass ROC plot for top ten countries . . . . .	91
6.5	Percent correct classification, supervised approach . . . . .	93
B.1	Softpedia corpus action accuracy by category . . . . .	103
B.2	Technical corpus action accuracy by category . . . . .	104

# List of Abbreviations and Symbols

## Abbreviations

APT	Advanced Persistent Threat
CAMEO	Conflict and Mediation Event Observations
COPDAB	Conflict and Peace Data Bank
CYLICON	Cyber Lexicon Event Dataset
DDOS	Distributed Denial-of-Service
GDELT	Global Database of Events, Location, and Tone
ICB	International Crisis Behavior
ICEWS	Integrated Crisis Early Warning System
LDA	Latent Dirichlet Allocation
LSA	Latent Semantic Analysis
MID	Militarized Interstate Dispute
OEDA	Open Event Data Alliance
PETRARCH	Python Engine for Text Resolution And Related Coding Hierarchy
TABARI	Textual Analysis By Augmented Replacement Instructions
UNODC	United Nations Office on Drugs and Crime
w2v	Word2Vec
WEIS	World Event/Interaction Survey

# Acknowledgements

I would like to thank my adviser and chair, Professor Michael D. Ward, for his help and encouragement. I am grateful for your advocacy and enthusiastic support of my research interests. You have been a tremendous inspiration to me and it is an honor working with you.

In addition to Professor Ward, I would also like to thank all the members of my committee: Professor Kyle Beardsley, Professor Scott de Marchi, and Professor Mark Crescenzi, for their advice and support. I am honored to follow in your footsteps.

My sincerest thanks to John Beielser, Casey Hilland, Andy Halterman, and all of the Caerus Associates team for providing me with the tools, resources, and motivation upon which I built this research project.

My thanks also to Kathryn Alexander, Cassy Dorff, Tobias Konitzer, Sophie Lee, Shahryar Minhas, Peter Vining, Fonda Anthony, and the whole Duke Political Science family. And to Professors Joseph Grieco and Christopher Gelpi. It has been a pleasure working with you and I hope we have many more opportunities to collaborate in the future.

Thank you also to my UNC Asheville family: Professors Surain Subramaniam, Mark Gibney, Linda Cornett, Dorothy Sulock, Sam Kaplan, and Patrick Bahls.

Finally, I would like to thank my family, Heidi, James, Becky, Jacob, Yaoyao, and my grandparents, John, Jocelyn, James, and June. You have taught me everything I know about compassion, hard work, kindness, and generosity.

# 1

## Introduction

Recent years have witnessed an explosion of political action in cyberspace. Some of these events, like the exploitation of Syrian radar systems by Israeli operatives, coincide with real-world conflicts while others, like the anti-SOPA blackouts of 2011, are executed almost entirely on the internet. The range of actors in cyberspace is immense. Governments, private companies, nonprofits, academic institutions, and individuals all operate in the same global network. The democratizing force that allows instantaneous and costless global communication has also democratized a monumentally powerful instrument of espionage, sabotage, theft, and political persuasion. Policymakers are only now, more than two decades after the invention of the World Wide Web, considering the ramifications of its widespread and rapid adoption.

The recent breach of Sony networks, allegedly by agents of the North Korean government, highlights several of the complications that cyber conflict poses. When suspects for the attack initially ranged (and, in fact, continue to range) from individual hackers and disgruntled employees to nation states, the problems of attribution and democratized technologies of attack are clear. The sophistication or impact of a cyberattack is not necessarily an indicator of the responsible actor and, many

observers argue, the gap between various actor capabilities will continue to diminish (Schneier, 2015b,a). Furthermore, policymakers need to better understand how liability for cyberattacks should be handled when a single cyberattack can expose the personal and financial information of millions and the responsible party may be discovered only months or years later, if at all.

Political scientists have ironically little data on the events of the information superhighway. This work remedies this problem by introducing a framework for processing raw news stories and transforming them into structured data on the political and economic events of cyberspace. While event datasets have been popular in political science for several decades now, many of the common dictionaries have not been updated to incorporate cyber conflict actors and events. Here, I present work towards automating the task of dictionary creation for event coding in the context of cyber conflict. Using natural language processing and machine learning techniques, candidate domain-specific words are identified and a probabilistic model for classifying these words is described.

This paper begins with an overview of cyber conflict in general, existing data on cyber conflict, and the role of event data in political science. Next, datasources are identified for use in coding cyber conflict events. Then, methods for extracting features for event-coding dictionaries are described. Finally, the paper concludes with directions for the future of this project.

## 1.1 Cyberspace and Cyber Conflict

The study of cyber conflict is difficult because, beyond the fact that it all occurs in “cyberspace,” the characteristics of cyber conflict vary substantially from one instance to another. It has variously been compared to espionage, sabotage, con-

ventional warfare, and weapons of mass destruction.<sup>1</sup> The prefix cyber reflects the domain in which cyber conflict occurs. While often used synonymously with “the internet” or “the world-wide-web,” cyberspace is defined in a number of ways and can encompass a much larger realm.<sup>2</sup> In the context of cyber conflict, cyberspace is generally considered to extend beyond the limits of the internet and to include all computer systems, networked or not. This means that airgapped systems, those that are separated from all other networks by a physical barrier, are within the area of operations for cyber conflict.<sup>3</sup> Critically, this definition is inclusive of special purpose computers such as programmable logic controllers (PLCs), task-specific devices that manage mechanical processes.

Political actors engage with one another, and with the public, in myriad ways. Indeed, cyber political events, or cyber events in this paper, overlap with several types of traditional political interaction. Online protests have, at times, taken the place of physical protests and marches. Self-imposed blackouts of websites in opposition of SOPA, the Stop Online Piracy Act, are largely considered to have been instrumental in the bill’s failure in the U.S. House. In the midst of damaging intelligence leaks in 2013, the Office of the Director of National Intelligence established a tumblr blog to publicize its responses to reports of massive cyber surveillance. And over several

---

<sup>1</sup> For comparisons to espionage and sabotage, see Rid (2013) For a comparison with conventional war, see Gartzke (2013). For a comparison to weapons of mass destruction, see Clarke (2009); Nye (2011).

<sup>2</sup> The term “cyberspace” was coined by Gibson (1982) in his short story *Burning Chrome*. The Oxford English Dictionary defines cyberspace as “The notional environment in which communication over computer networks occurs” (OED, 2014). Damir Rajnovic of CISCO provides an interesting analysis of various definitions of cyberspace (Rajnovic, 2012). He writes that, “probably contrary to popular beliefs, networks and Internet are not necessarily part nor are required for cyberspace but they are still ‘desired.’”

<sup>3</sup> Indeed, these airgapped systems are often the most valuable targets for practitioners of cyber-warfare. For an interesting example of efforts to these ends, see Schneier (2014a) and Sanger and Shanker (2014). Documents described by *Spiegel* outline the US National Security Agency’s ability to covertly implant radio frequency hardware on a target’s computer in order to maintain contact with that computer when it is seemingly disconnected from any other network.

years between 2006 and 2010, a malicious computer virus now known as Stuxnet managed to jump air gaps and infect the control systems of uranium enrichment centrifuges in a high-security Iranian nuclear facility. This last example is probably the most well-known case of what is popularly referred to as cyberwar.

“Cyberwar” is generally used in reference to the use of malicious computer code for political objectives. As scholars have pointed out, this definition is a misnomer as it does not necessarily fit conventional notions of what constitutes war (Rid, 2013). Instead, cyber conflict will be the preferred term used in this paper. Cyber conflict refers to the use of computer networks and their associated hardware to damage, impede, or exfiltrate computer-controlled processes or information for military or political objectives. To the extent that these objectives largely mirror the objectives of many traditional military operations, the term cyberwar seems sensible. However, cyber conflict is more inclusive. While physical acts of espionage are not typically acts of war, the distinction in cyberspace may be less clear.<sup>4</sup> Furthermore, while it is tempting to draw parallels between traditional war and cyber conflict events, doing so threatens to limit our understanding of this new phenomenon. For these reasons, and in keeping with the emerging standard in political science literature, “cyber conflict” is selected over “cyberwar.” A cyber weapon, then, is the tool or set of tools utilized in a cyber conflict event. This could take the form of a virus such as Stuxnet, non-virus malware such as sabotaged source code, or a tactic such as Distributed Denial of Service (DDoS).

While Stuxnet has become the eminent example of cyber conflict since its discovery in 2010, it was far from the first. An oft-cited case of cyber conflict reportedly took place as early as 1982. Former Secretary of the Air Force, Thomas Reed, detailed a plot by the CIA that involved the sabotage of software for the control system

---

<sup>4</sup> For a good discussion of the (non)distinction between cyber espionage and cyber attack, see Schneier (2014b).

of a natural gas pipeline. Suspecting that Soviet spies would steal the software for use in a Siberian pipeline, a logic bomb was inserted into the code that ultimately resulted in the explosion of a section of the line (Reed, 2005). Reed's account is dubious as no other sources have verified his story.<sup>5</sup> Since then there have been several instances of both state and non-state cyber conflict. During the 1999 NATO bombing of Yugoslavia, hackers engaged in a DDoS attack that temporarily shut down NATO websites and disrupted NATO email servers. During the Georgian offensive in South Ossetia and subsequent Russian invasion of Georgia in 2008, actors on both sides of the conflict engaged in cyber attacks. Again, these largely took the form of DDoS events and inhibited the flow of information to civilians more than they hampered actual military operations. One year earlier, in 2007, Israel conducted an airstrike in Syria to destroy a nascent nuclear reactor. Eight Israeli military jets penetrated Syrian airspace without appearing on defense radars; the radars had been infiltrated by Israeli hackers and made to show normal conditions rather than the intruding aircraft (Smith, 2010). At the same time, Israel and the United States were secretly deploying the most advanced piece of malware yet discovered: Stuxnet. Some experts estimate Stuxnet set back the Iranian nuclear program by eighteen months to two years through the covert destruction of uranium enrichment centrifuges (Sanger, 2012).<sup>6</sup> And while Stuxnet is largely considered the first confirmed cyberattack to cause physical damage, recent reports point to a second attack perpetrated by Russia against a German steel company. For an expanded discussion of previous cases of cyber conflict, see Lee (2013).

---

<sup>5</sup> Though it is also not impossible; the CIA did in fact sabotage technologies that were then "leaked" to the USSR (Weiss, 2007).

<sup>6</sup> Others disagree. See Barzashka (2013).

## 1.2 Data on Cyberspace

Despite its clear rise in prominence as a venue for political interaction of all types and the perception that it poses substantial security risks for states, comprehensive data on cyber conflict appropriate for use in questions of political science is lacking. In fact, even Stuxnet is excluded from the current iteration of the Militarized Interstate Dispute (MID) dataset despite its discovery during the period of time covered by the dataset. This omission occurs even though several other MIDs related to US and Israeli concerns over Iran's nuclear program that failed to result in any physical damage or an otherwise substantial outcome are included.<sup>7</sup> Failure to include the actions that state actors take against one another in the context of dispute and militarized conflict from data sets such as MIDs simply because these actions occur in cyberspace rather than on traditional battlefields threatens to bias future work with these sources. To the extent that electronic resources, and the physical infrastructures they maintain, represent valuable state assets, cyber conflict is a real and growing threat to state security.

Of course, this concern leads to a bigger question: how does cyber conflict fit into existing paradigms of political interaction? When cyber attacks are carried out by a state military or intelligence body and result in physical harm to a target's infrastructure, omitting this event from general conflict event datasets requires special justification. Exclusion of operations based on the technology utilized could date data sets and will likely bias results derived from the data.

The lack of unclassified data on cyber conflict is a serious challenge to the incorporation of this activity into the study of politics. That cyber conflict and its effects are often unobservable to those without access to sensitive computer systems

---

<sup>7</sup> Dispute number (DispNum3) 4524 involves the violation of Iranian airspace by US warplanes. Dispute number 4548 relates to Israel's deploying of nuclear submarines to the Persian Gulf as a show of force. Dispute number 4535 is somewhat more substantial and involved the seizure of documents and staff by US forces from the Iranian consulate in Iraq.

makes it less visible even than covert military operations. When the United States military used stealth helicopters to assault Osama Bin Laden’s compound in Pakistan, an operation that required extreme secrecy to avoid detection by the Pakistani military, the event was live-tweeted by Sohaib Athar, an annoyed neighbor. Unfortunately, there is not always someone available to live-tweet cyber conflict operations. Nonetheless, political scientists should not limit their study to those things that are perfectly, or even mostly, transparent. In all cases, political scientists (as well as researchers in other disciplines) are at the mercy of available information and risk the possibility of obtaining a non-random sample of the population of interest. However, further limiting a sample by omitting all instances of a given phenomenon risks exacerbating selection effects and restricts the generalizability of findings. In the case of cyber conflict, it is clear that the available sample of data is smaller than the full population.<sup>8</sup> How much smaller is unclear. But, as always, we should seek to use all available data to extrapolate, to the best of our ability, the true nature of this phenomenon.

Currently, there is one publicly-available dataset on cyber conflict that I have identified. These data are contained in the appendix to Valeriano and Maness (2014). Unfortunately, the data consist of only limited information about conflicts between existing rival states. The authors justify the choice to include only rival dyads, in part, by writing “it would therefore make sense that cyber conflict would be added to the arsenal of rival interactions and should be investigated under this context.” Of course, it also makes sense that cyber conflict would be added to the arsenals of non-rival dyads as well. Furthermore, this focus precludes research into the characteristics of cyber conflict that most promise to distinguish it from traditional strategies of

---

<sup>8</sup> The Washington Post reported in 2013 that US intelligence agencies carried out 231 offensive cyber operations in 2011 (Gellman and Nakashima, 2013). This is valuable information in that it may help us to extrapolate estimates of the true population magnitude of cyber conflict given the assumption that discovered instances of cyber conflict are a representative subset.

warfare: it is no more costly to project globally than it is locally and the targets it is capable of affecting are distinct from those that are easily affected via conventional means. Valeriano and Maness further ask “why focus on all possible dyads when we have exhaustive data on those states most likely to engage in crises, escalated conflicts, and wars?”<sup>9</sup> However, they make no effort to justify their claim to have “exhaustive” data on cyber conflict even between rival states.<sup>10</sup>

Existing interstate rivalries are largely geographic; proximate states have traditionally had greater capabilities to target one another militarily and therefore become rivals. In fact, in Valeriano and Maness’s data, the only dyad pairs that are separated by more than several hundred kilometers are those that involve the United States. However, cyber conflict promises states the ability to project power much further than they could using conventional forces. Potential rival pairs that have never been realized due to constraining costs and an inability to move forces across intermediary states or oceans can now resort to cyber conflict to engage one another. Furthermore, existing rivalries are limited to those disputes over incompatibilities that can be contested via traditional military tactics. Cyber weapons offer a new set of targets including intellectual property, economic assets, command and control systems, and information systems in general. While disputes involving these assets may have previously been fought without militaries simply because militaries did not have the appropriate toolset to resolve the disagreement, these disputes may now fall into the realm of military operations. Finally, cyber conflict at a distance may ultimately result in different deterrence dynamics than cyber conflict with rivals. Cyber conflict with neighbors risks the possibility of dispute escalation and subsequent kinetic conflict. Cyber attacks against geographically distant states, on

---

<sup>9</sup> One possible reason is selection bias.

<sup>10</sup> If the NSA documents made public by Edward Snowden have shown us anything, it’s that politically-motivated cyber operations and the surreptitious infiltration of commercial and government networks is actually quite common between strong allies.

the other hand, would likely limit retaliation to further cyber conflict, economic measures, or political statements. For these reasons, reliance on data about cyber conflict between state rivals only is short-sighted and likely misleading. A new and comprehensive dataset is critical to properly assess this emerging technology.

### 1.3 Event Data

In just the past two years, at least three event datasets have been introduced in political science: The Global Database of Events, Language, and Tone (GDELT), the Integrated Conflict Early Warning System (ICEWS) dataset, and the Phoenix dataset (Leetaru and Schrod, 2013; Lustick et al., 2015; Open Event Data Alliance, 2015b). These are the latest generation in a series of political event datasets. Earlier efforts include the Conflict and Peace Data Bank (COPDAB) and the World Event/Interaction Survey (WEIS), (Azar, 1980; McClelland, 1978). Event datasets provide very fine-grained data on individual events, usually at the daily level and with specific details about the actors involved. Modern event datasets often also provide geographic information at a subnational level. These datasets are enormous, typically hundreds of thousands or millions of events.

The event datasets listed above are built from streams of open source news stories. The stories are processed through software that uses pre-defined dictionaries to infer the actors and actions they describe. Common software for this purpose includes KEDS, its successor TABARI, and PETRARCH (Schrod, 1998, 2014; Open Event Data Alliance, 2015a). The Open Event Data Alliance, authors of PETRARCH, provide the graphic from Figure 1.1 to illustrate their event-coding process. Raw stories are first collected, or “scraped,” from online sources. These are uploaded to a MongoDB database and formatted to the specifications required by TABARI (or PETRARCH). The stories are then passed to TABARI which uses the supplied dictionaries to produce structured data. The data are then de-duplicated to remove

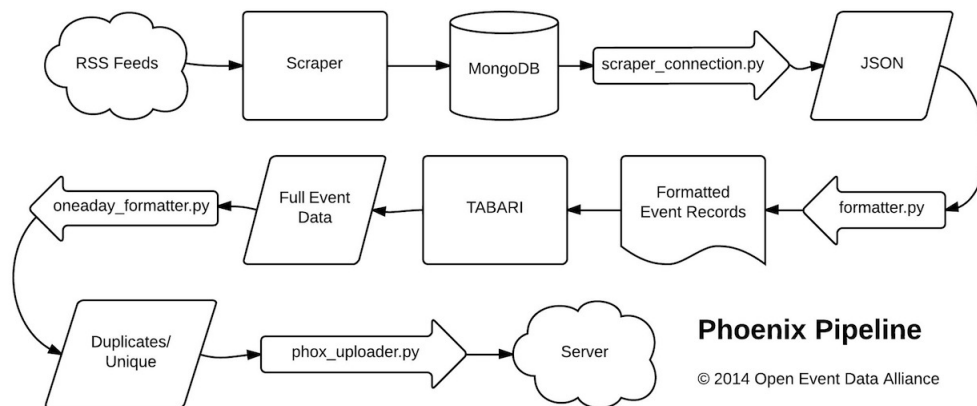


FIGURE 1.1: The Phoenix pipeline (Open Event Data Alliance, 2015c).

multiple stories referencing single events and uploaded to a server that hosts the end product. Under ideal circumstances, human interaction is only required to select appropriate news sources, an ontology for the resulting structured data, and to populate the dictionaries. However, this last step, dictionary creation, requires a substantial level of effort. The standard CAMEO verb dictionary used by the Phoenix dataset is nearly 15,000 lines long and includes very specific phrases that would not necessarily be apparent to researchers a priori. The country actors dictionary, just one of multiple actor dictionaries utilized by Phoenix, is nearly 55,000 lines long. Furthermore, as the relevant actors and language evolve, these dictionaries require regular updates to maintain up-to-date event data.

## 1.4 Conclusion

In the following pages, I will introduce a method for generating event data in new domains and to update existing event data with minimal researcher input. This process allows researchers to rapidly iterate over new ideas and novel ontologies for producing structured data from raw text news stories. By automating the process of dictionary population for event coding applications, not only are researcher-hours minimized, projects are made reproducible from the very first stages of the data

creation process. An example dataset of politically-relevant events in cyberspace is produced using this new method. First, in Chapter 2, the need for a new method of creating event data on cyberspace is elucidated. Examples of common conflict datasets from political science are given and the means by which they are produced are briefly described. The corpus to be coded is then introduced and described. The chapter ends with an attempt to code the events of this corpus with an existing state-of-the-art event coding procedure. The shortcomings of this method motivate the research described in subsequent chapters. Chapter 3 describes a novel approach for producing event data. Drawing on existing techniques and software as well as recent advances in deep learning and natural language processing, this method produces dictionaries appropriate for event data generation with only a small number of researcher-provided seed words. In Chapter 4 the method is applied to the cybersecurity news corpus and a novel event dataset, CYLICON, is produced and described. Particular focus is given to generating and validating entire verb dictionaries (though the actor, agent, issue, and synset dictionaries are also updated using variations of the same automated process). Chapter 5 investigates how the cybersecurity events in CYLICON correlate with events and statistics from other datasets of interest to social scientists. Chapter 6 turns its focus to automatically classifying actors using a very similar process to that described in Chapter 3. This chapter is a validation study in which components of automatically generated actor dictionaries are compared to manually-coded, “ground truth” dictionaries.

## From Cybersecurity Documents to Event Data

The politics of cyberspace, and the impacts that events in cyberspace have on politics, have dominated headlines in recent years. From massive data breaches to ransomware to sophisticated espionage and sabotage, it is hard to overstate the importance of this emerging domain to both everyday life and geopolitics in the modern world. Despite this, little attention has been paid to cyberspace by political scientists.

An event dataset specific to cyberspace would facilitate research on the social aspects of this emerging domain. However, current dictionaries used for coding political events are insufficient for this purpose. Despite already encompassing a vast array of political actors and general verb phrases related to politics, these manually-updated dictionaries are nonetheless ill-suited for cyber conflict. When relevant actors range from nation states to anonymous online collectives to experts in cryptography, maintaining current dictionaries is a challenge. Additionally, important vulnerabilities and computer viruses are often given unique names that trump generic verbiage in reporting; Heartbleed and Stuxnet, for example. Advanced persistent threats (APTs) are also assigned names as they are identified. These names tend to change as the APTs adapt to maintain their ability to operate covertly. In other words,

as soon as an APT is identified and named, the actors involved will abandon their now-compromised techniques and reemerge at a later time to be given a new name upon rediscovery. Keeping up with this rapidly evolving domain in which the relevant vocabulary is emerging in tandem with the innovations of silicon valley and the research of cybersecurity firms poses a challenge to researchers looking for timely data.

To better understand the challenges inherent in developing a cyber events dataset, this chapter explores the raw cybersecurity corpora, briefly discusses existing solutions to the dictionary creation problem, and presents a first cut version of a cyber event dataset produced using existing CAMEO dictionaries provided by the Open Event Data Alliance (OEDA) and PETRARCH. I begin with a summary of some existing datasets utilized by political scientists and the methods used to generate them. Next, the cybersecurity-related texts that will serve as the foundation for a cyber events dataset are examined with a focus on the challenges they pose to existing event coding frameworks. These challenges are demonstrated by attempting to apply the PETRARCH coder and CAMEO dictionaries, unmodified, to the data. The resulting events are then compared side-by-side to their originating stories. Additionally, stories uncoded by PETRARCH are analyzed to determine whether important events are missed by the CAMEO dictionaries. The chapter concludes with the consideration of simple techniques for automated dictionary generation and argues that a novel solution is required.

This chapter serves to emphasize the point that domain-specific dictionaries are critical to producing accurate event data of the domain of interest and to make the case that new techniques are required to produce these dictionaries. Even when researchers are interested in a subset of the existing CAMEO ontology, they should be wary of relying on the turnkey PETRARCH & CAMEO combination. For instance, as will be demonstrated here, many cybersecurity events can be categorized broadly

as “assault” under the CAMEO ontology but will not be coded as such due to their domain-specific vocabulary.

## 2.1 Data Creation in Political Science

Many of the the highly cited and ostensibly current datasets that political scientists rely on are manually-coded. Teams of researchers spend months and years pouring over news articles to identify events of interest, to infer underlying latent features, or to develop rules and dictionaries for subsequent machine-coding schemes. Recent research efforts have applied advances from natural language processing to reduce the man hours required to produce social science data. These efforts have resulted in massive machine-coded event datasets. Even more cutting-edge efforts have focused on developing algorithms to further reduce the man-hours required to produce data of interest to social scientists. One such effort applies support vector machines to a bag-of-words-transformed dataset to classify political governance. Despite these efforts, most of the data that are relied upon by political scientists are produced, in large part, by hand.

### *2.1.1 Manually Coded Datasets*

The vast majority of datasets cited in political science research efforts are the product of manual coding. In this section, I will highlight a few of the most highly-cited of these.

The Militarized Interstate Dispute (MID) dataset has long been a standard for scholars of international relations. The latest incarnation, Palmer et al. (2015), updates this data to include the years 2002-2010. At best, then, the MID dataset is five years delayed from the present. Previous versions of the MID dataset were wholly coded by hand while the latest was supported by automated article classification. In particular, support vector machines (SVM) were utilized to categorize articles

into “relevant” and “irrelevant” bins prior to manual dispute coding. The SVM successfully pruned the relevant news articles down from over 1.7 million to 132,515 or 15,000 per year. Manual coding of the remaining articles resulted in the identification of 262 militarized interstate disputes in the period between 2002-2010. The MID project appears to report neither inter-coder reliability scores nor man-hours of effort.

The International Crisis Behavior (ICB) dataset is similarly coded by hand (Brecher and Wilkenfeld, 2000). The latest version, released in 2010, covers crises between 1918 and 2006. Like MIDs, the ICB is manually-coded by a team of researchers using a corpus of news articles and the human-hours required to code and validate these data limit the timeliness with which they can be produced. The 2010 version features crises no more recent than three years prior. Unlike MIDs, ICB has yet to make any effort at leveraging machine-learning techniques to increase coding efficiency. Again, inter-coder reliability scores are unavailable.

While manual-coding has previously provided some degree of accuracy, subject to the coders’ ability to judge the context and content of the data they analyze, advances in natural language processing coupled with machine-coding techniques promise to match or surpass the reliability of human coders (King and Lowe, 2003). Furthermore, machine-coding is potentially cheaper and more time-efficient than hand-coding, producing more up-to-date data for forecasting applications.

### *2.1.2 Machine Coded Datasets*

Recently, machine-coded event datasets have proven a valuable resource for social scientists. ICEWS, GDELT, and Phoenix provide detailed representations of newsworthy political events on a global scale. These datasets allow for an unprecedented view of world events for researchers interested in a variety of topics, from protests to military conflicts to diplomatic overtures. Given a set of rules and dictionaries, software solutions code these datasets from a vast corpora of input news articles.

However, dictionary creation is still done manually and these dictionaries must be updated periodically to ensure that event coding schema accurately capture current affairs.

ICEWS, GDELT, and Phoenix are the successors to an earlier generation of event datasets that include COPDAB and WEIS (McClelland, 1978; Azar, 1980). In the 1990s KEDS, the Kansas Event Data System, offered the first software solution for event coding (Schrodt et al., 1994). Schrodt motivates KEDS by noting that “Historically, event data have been coded by legions of bored undergraduates and M.A. students flipping through copies of the New York Times and other printed sources” (Schrodt et al., 1994, 562). Given a set of dictionaries specific to a domain of interest, KEDS would parse a corpus of text to produce structured data of the format  $\langle \text{actor}_1 \rangle \langle \text{action} \rangle \langle \text{actor}_2 \rangle$ . KEDS was succeeded by TABARI, then PETRARCH, and soon the upcoming PETRARCH 2 (Schrodt, 2014).

In large part because they do not rely on human coders, event datasets like these are updated regularly and often provide data as recent as one day (Phoenix and GDELT) or one year (ICEWS).<sup>1</sup> They also provide reproducibility in that version control of the coding rules, dictionaries, and software guarantees that previous versions of the data can be perfectly replicated. On the other hand, these datasets often lack contextual information that might be provided by other, more specific data. For example, event datasets generally do not provide information on casualty levels associated with any particular violent event that might be available from human-coded resources.

ICEWS, the Integrated Crisis Early Warning System, was funded by the Defense Advanced Research Project Agency and the Office of Naval Research (Boschee et al., 2015). The program aimed to forecast events of interest including international crisis,

---

<sup>1</sup> A proprietary version of ICEWS offers more timely data but is embargoed from public release for one year.

domestic crisis, insurgency, rebellion, and ethnic and religious violence (Ward et al., 2013). GDELT, the Global Database of Events, Location, and Tone, was introduced in Leetaru and Schrodt (2013). GDELT is currently publicly-available via Google’s BigQuery. Due to legal issues, many of the original contributors to GDELT left the project and began work on a new event dataset, Phoenix (Open Event Data Alliance, 2016). Phoenix is maintained by the Open Event Data Alliance (OEDA) and is available freely online. All three event datasets utilize the standard CAMEO taxonomy of event types (Schrodt, 2012).

### *2.1.3 Machine-learned Datasets*

At the current cutting-edge of data generation in political science are efforts to entirely remove humans from the coding process. Minhas et al. (2015) use support vector machines (SVM) to predict polity and regime type from U.S. State Department Country Reports. This approach relies on a training set of pre-classified countries that must be hand-coded. Once the model is trained, however, regime-type predictions can be made for new observations with no human intervention. The researchers report exceptional out-of-sample predictive performance with class-wise precision and recall exceeding 90% each in all but one category.

## 2.2 Raw Cybersecurity Data: Lots of Text

Currently, two corpora of news stories serve as the foundation of this dataset. These corpora were selected for their subject-matter specificity. The first is a comprehensive collection of articles from Softpedia’s Security News section. The second is a collection of stories gathered from various sources that track cybersecurity and technology-related news and opinion pieces. I will refer to this corpora as the Technical Corpus as its sources tend to include more technical details and are not targeted at a casual audience. Both corpora are described in more detail below.

### *2.2.1 Softpedia Corpus*

Softpedia stories, more so than those in the technical corpus, resemble the newswire stories common to event datasets. They are short, generally only two or three paragraphs long, and written in accessible language. The stories that generate this corpus were scraped using novel software written in Python. The 18,338 stories cover 2005 through most of 2014. An excerpt selected from a sample story of this corpus reads:

The Heartbleed bug, the OpenSSL vulnerability that can be exploited to obtain sensitive information from affected servers, has made a lot of headlines this week. The bug is highly critical because it can be used to steal passwords, financial data, and the contents of communications (Kovacs, 2014b).

Unsurprisingly, the frequency of stories in this corpus increases over time. This is a common problem faced by event datasets as the availability of electronic news sources has increased over the past few decades. Figure 2.1 depicts the distribution of stories over time. In this case, however, the increased volume does not correspond to increasing numbers of news sources transitioning to online distribution. Instead, this likely reflects both an increase in demand for cybersecurity news as well as an actual increase in cybersecurity-related activity over the past decade.

### *2.2.2 Technical Corpus*

The technical corpus is collected from an array of cybersecurity RSS feeds. It contains 56,948 stories from 2014. The heterogeneous sources present a challenge as their intended audiences vary substantially. Some of the stories are written for wider audiences while others are intended for cybersecurity experts and practitioners. They tend to be longer and more detailed than stories from the Softpedia corpus. On the other hand, this heterogeneity better resembles the diverse range of sources often

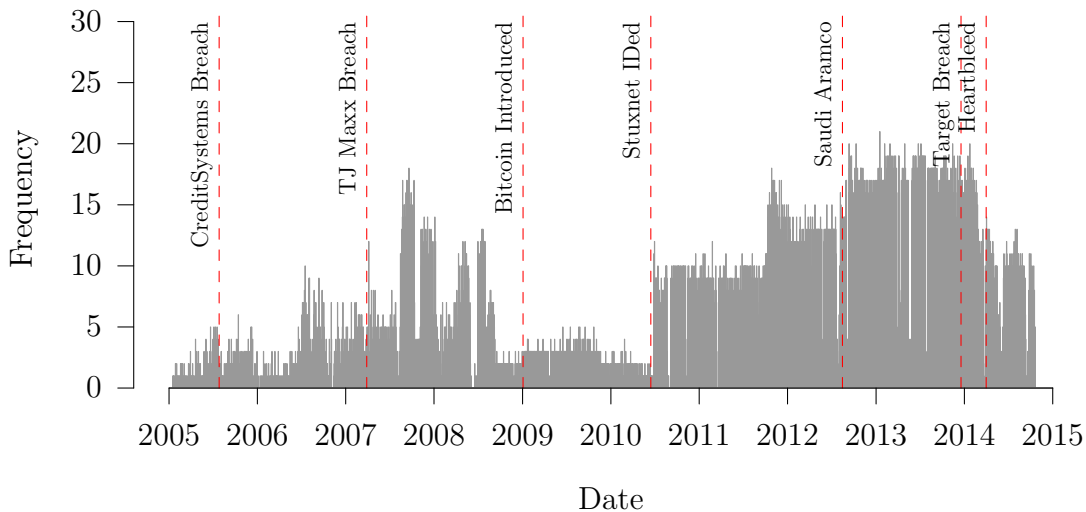


FIGURE 2.1: Softpedia stories over time.

compiled for event coding projects. An excerpt from one of these stories, selected to match the subject matter of the example from the Softpedia corpus, reads:

The maintainers of the OpenSSL library, one of the more widely deployed cryptographic libraries on the Web, have fixed a serious vulnerability that could have resulted in the revelation of 64 KB of memory to any client or server that was connected. The details of the vulnerability, fixed in version 1.0.1g of OpenSSL, are somewhat scarce. The OpenSSL Project site says that the bug doesn’t affect versions prior to 1.0.1 (Fisher, 2014).

This corpus is comprised of stories from several hundred sources, but the majority of stories are from a few particularly prolific publications. The top thirty sources are shown in Table 2.1. The distribution of stories over time for the technical corpus is shown in Figure 2.2. The low level of stories early in the year is an artifact of how different RSS feeds archive their older content.

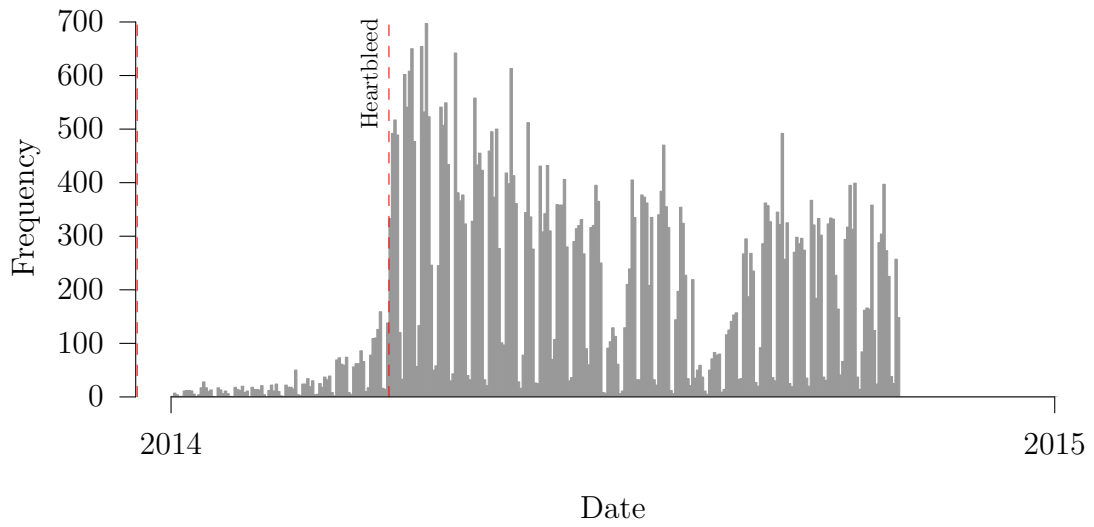


FIGURE 2.2: Technical corpus stories over time.

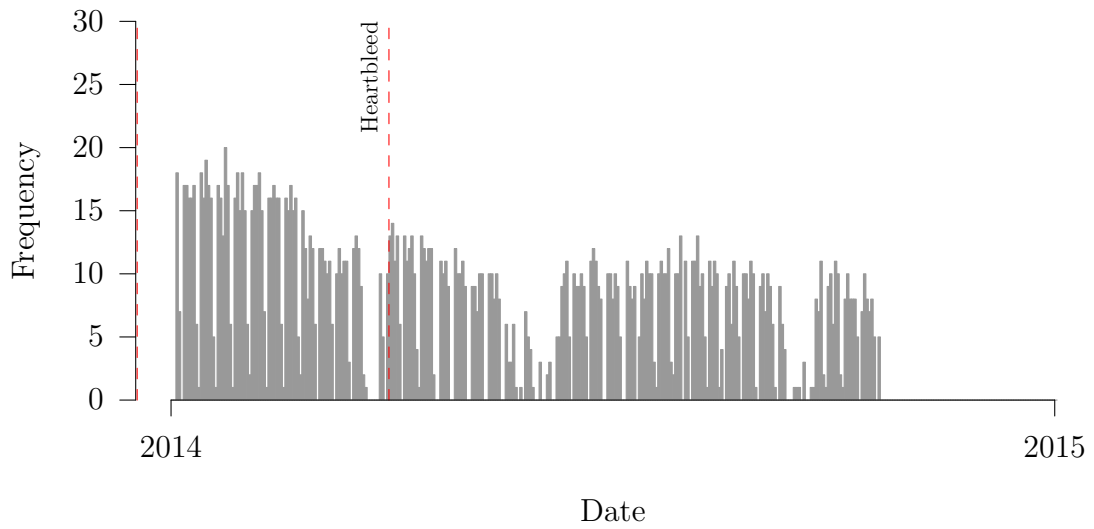


FIGURE 2.3: Softpedia stories in 2014 for comparison with Figure 2.2.

Table 2.1: The 30 most frequent sources of news in the technical corpus.

feedproxy.google.com	bits.blogs.nytimes	feeds.arstechnica.com
go.theregister.com	itnews.com	isc.sans.edu
cnet.com.feedsportal	computerweekly.com	dataprotectioncenter.com
securitynewsportal.com	databreachtoday.com	informationweek.com
csoonline.com	eweek.com	hotforsecurity.com
zdnet.com	bankinfosecurity.com	ciol.com
darkreading.com	inforisktoday.com	news.softpedia.com
rss.computerworld.com	infoworld.com	web.nvd.nist
circleid.com	fiercegovernmentit.com	feeds.trendmicro.com
databreaches.net	esecurityplanet.com	secureworks.com
⋮	⋮	⋮

### 2.3 CAMEO and the Need for Novel Dictionaries

New dictionaries necessary to produce viable event datasets that represent cyber conflict and cybersecurity. To demonstrate this need, a first draft of the CYLICON cyber events dataset is produced using existing event coding dictionaries and software. No claim is made that these dictionaries will accurately reflect cybersecurity events or actors. However, it is possible that some cybersecurity events will map reasonably well onto the CAMEO verb ontology and that many of the actors and agents prominent in cybersecurity news will also exist in the PETRARCH actor and agents dictionaries. Using the corpora described above and the PETRARCH coding software from the Open Event Data Alliance, this prototype dataset is described below.

The CAMEO ontology for event coding began development in 2000 (Gerner et al., 2002). The event taxonomy consists of 20 top-level categories and 329 subcategories. The authors motivated CAMEO by contrasting it with existing event-coding ontologies: “While innovative when first created, these [older] coding systems are not optimal for dealing with contemporary issues such as ethnic conflict, low-intensity violence, organized criminal activity, and multilateral intervention” (Gerner et al.,

2002, 1). This same issues underlies the motivation for automated dictionary generation; CAMEO cannot accurately describe some domains of interest in 2016.

The 20 top-level event categories from CAMEO include: make a public statement, appeal, express intent to cooperate, consult, engage in diplomatic cooperation, engage in material cooperation, provide aid, yield, investigate, demand, disapprove, reject, threaten, protest, exhibit force posture, reduce relations, coerce, assault, fight, and use unconventional mass violence. Roughly 15,000 verb phrases are mapped to these top-level categories or any of the 329 subcategories. None of these, however, are suited for coding cybersecurity events. For instance, the word “hack” appears only in the context of “hack to death” which maps to subcategory 1823, *kill by physical assault*.

The draft dataset consists of 3,980 events. It was created with PETRARCH, the OEDA’s open source event-coding software, and the default set of CAMEO dictionaries. A random sample of 100 of these events is inspected and a representative selection of events are presented here. Only a very small portion of the sampled events that relate to cybersecurity are coded in such a way that they might be considered accurate. In the following lists, the coded sentence is preceded by the CAMEO category it was assigned.

1. **Use conventional military force:** “Indonesian hackers have launched a distributed denial-of-service (DDOS) attack against the official website of the Australian Security Intelligence Organization (ASIO).” (Kovacs, 2013e)
2. **Abduct, hijack, or take hostage:** “At the end of January, a 22-year-old student from Poland was arrested for defacing the site of the country’s prime minister as part of an anti-ACTA protest.” (Kovacs, 2012a)

In item 1, a DDOS attack is identified as use of conventional military force. To the extent that a DDOS attack is a destructive offensive measure, this coding might be

considered accurate. Whether it is a conventional military tool is less clear. In fact, DDOS is most often utilized by individuals or small groups of non-state actors. One particularly notable exception to this is the use of a massive DDOS attack against GitHub by China (Graham, 2015). In apparent response to subversive software hosted on GitHub, the great firewall of China was configured to redirect debilitating amounts of traffic to GitHub that resulted in substantial service disruptions for many days. The event referred to in item 1 does not appear to have been undertaken by the Indonesian military.

Item 2 maps a website defacement incident to a CAMEO category that includes hijacking and hostage-taking. This is a fair portrayal of website defacement: control of property has been seized by a third party and diverted from its intended use.

The dual-meaning of the word “breach” leads to many cybersecurity events falling into the CAMEO category *defy norms, law*. Three instances are reproduced below.

- **Defy norms, law:** “Hackers of the Pakistani group TeaM MaDLeeTs have breached the systems of the organization responsible for the registration of Montenegro (.me) domain names (domain.me).” (Kovacs, 2014a)
- **Defy norms, law:** “According to Snowden, the United States has breached the systems of the Tsinghua University in Beijing, one of the country’s top education and research institutes.” (Kovacs, 2013c)
- **Defy norms, law:** “Hackers backed by the Russian government are believed to have breached the unclassified computer network of the White House recently...” (Ilascu, 2015)

The majority of the events coded by PETRARCH using CAMEO are either accurate but unrelated to cybersecurity or are inaccurately coded cybersecurity events. Despite the small number of events that are coded in reasonable ways according to the

CAMEO ontology, the verb dictionary is not capable of reliably coding cybersecurity-related events. Without relevant event categories that correspond to different types of cybersecurity incidents, most cybersecurity stories that do get coded are given completely irrelevant event codes. This ignores the many cybersecurity events that go completely uncoded due to their relevant verb phrases' absence from the standard CAMEO verb dictionary.

## 2.4 Preparation for Automated Dictionary Generation

Dictionaries are critical to the creation of event datasets using existing coding software such as PETRARCH or TABARI. For instance, the actor dictionary for ICEWS coding contains 164,838 entries. These entries are mapped onto a taxonomy of sectors and actor types. Using these dictionaries, an event coder identifies the actors in raw texts and classifies the text based on the actor's predetermined characteristics and affiliations. Currently, no dictionaries specific to events and actors in cyberspace exist. My preferred approach to dictionary creation is an automated classification scheme that extracts keywords from the text and builds dictionaries (or candidate dictionaries) with minimal researcher intervention. The groundwork for dictionary coders of this sort has already been developed in the Natural Language Processing community (Fleischman and Hovy, 2002; Nadeau and Sekine, 2007). Adapting this work to produce usable vocabularies for event coding is the focus of this dissertation.

In order to implement an automated coder of this sort, a variety of pre-processing and feature extraction steps are undertaken first. These include cleaning the data, sentence parsing, and named entity extraction.

### *2.4.1 Cleaning Data, Sentence Parsing, and Named Entity Recognition*

The raw text used here has either been scraped directly from HTML documents or collected via RSS feeds. In both cases, the text contains some markup and special

characters that must be removed prior to sentence parsing. To this end, extra whitespace, linebreaks, and nonsense characters (those that do not translate correctly into UTF-8) have been removed. Also, semicolons are replaced with periods such that all independent clauses in the text are parsed independently.

Next, Stanford’s CoreNLP software pipeline is used to parse the text (Manning et al., 2014). The annotators used here come from the Apache OpenNLP maximum entropy model set (Apache Software Foundation, 2010; Hornik, 2015). In later chapters, the CoreNLP shift-reduce parser is used instead.<sup>2</sup> These annotators tag sentences, words, and parts-of-speech as well as named entities. These tags are subsequently used for identifying named entities, triplets, and ultimately a cyber-domain-specific vocabulary.

Named entity recognition is a natural language processing technique for identifying proper nouns in parsed texts. The named entity recognition methods demonstrated here are provided by Apache Software Foundation (2010).<sup>3</sup> This process identifies 9,125 and 17,125 named entities in the two corpora. The algorithm is noisy and many of the names that occur only once in each corpus are not, in fact, named entities. URLs, for instance, are sometimes coded as named entities. Also, various spellings of the same entity are often coded independently; for example: “Yahoo” and “Yahoo!”. On the other hand, commonly tagged named entities correspond closely with what would be expected of a cyber conflict dictionary. Figures 2.4(a) and 2.4(b) show the top 200 most frequently-identified named entities in each corpus. Edward Snowden’s prominence in Figure 2.4(b) is representative of that corpus’s later time period.

---

<sup>2</sup> The shift-reduce parser was written by John Bauer and based on Zhu et al. (2013).

<sup>3</sup> Again, in later chapters, the Apache named entity recognition model is replaced with CoreNLP’s.



### 2.4.2 *Triplet Extraction*

A first attempt at dictionary extraction for event data is an implementation of an existing triplet extraction algorithm. This approach was abandoned due to inconsistent performance and a lack of context for extracted terms. However, the results of this preliminary work are briefly discussed here.

Triplet extraction is a method of natural language processing that identifies important features of an independent clause, namely the subject, object, and predicate. These keywords are useful because they could help to identify actors and actions within a text corpus that are missed by named entity extraction methods. The algorithm implemented here is from Rusu et al. (2007). The algorithm works by selecting certain nouns and verbs from the parse trees described in Section 2.4.1 according to a set of rules regarding their “depth” and order within particular phrases.

There are not yet perfect methods for triplet extraction and complicated sentence structures cause the current algorithm in use to break down. However, as Table 2.2 demonstrates, triplet extraction over a large corpus results in a sensible collection of relevant keywords. This suggests that identifying a vocabulary to describe political events requires a large dataset and cannot be accomplished on a sentence-by-sentence basis. Therefore, a two-step approach of dictionary creation followed by bag-of-words-based event coding seems appropriate. Easily parsed sentences will contribute to dictionaries that are then used to code events in sentences independent of their syntactic complexity. Furthermore, this follows the common event-coding scheme in which human coders create dictionaries given prior subject area knowledge and then automate the coding of raw data from these dictionaries.

Both named entity recognition and triplet extraction could be used as naive automated dictionary coding schemes. However, neither are ideal for this purpose. First, they are not domain-specific. Given a corpus, they will extract keywords

Table 2.2: The 25 most common results for subject, object, and predicate from the two corpora. Note that these results are preliminary and were extracted prior to some of the pre-processing described above. Also, blank entries result from character-encoding issues.

Subject		Predicate		Object	
Softpedia	Technical	Softpedia	Technical	Softpedia	Technical
company	NA	is	is	security	security
security	company	are	be	users	NA
hackers	Microsoft	be	are	information	data
Security	security	used	was	time	users
	Google	was	using	fact	information
users	Apple	have	according	number	time
com		using	including	website	number
hacker	report	make	said	data	way
		including	used	company	Google
Users	week	do	have	people	company
Google	Security	made	's	email	people
Microsoft	data	take	do	lot	Security
number	malware	called	make	hackers	news
attack	researchers	protect	use	access	part
information	vulnerability	get	get	websites	malware
vulnerability		stolen	been	way	percent
Experts	year	use	made	malware	vulnerability
malware	com	sent	take	attack	access
experts	percent	affected	protect	customers	years
website	information	targeted	were	part	email
people	Facebook	found	has	attacks	customers
	number	taken	come	computer	thousands
week	people	were	sent	spam	attacks
cybercriminals	companies	compromised	try	cybercriminals	attack
data		been	released	version	year
⋮	⋮	⋮	⋮	⋮	⋮

indiscriminately. Second, they provide no additional information for the classification of extracted elements.

## 2.5 Conclusion

Political scientists lack both an event dataset appropriate for analyzing cybersecurity incidents as well as an easy method for extending current datasets and data collection techniques to accommodate emerging or evolving domains. In the next chapter, I

introduce a technique for quickly updating or generating new dictionaries for event coding software. This method makes use of the preprocessing techniques that were described here as well as cutting-edge research from the fields of natural language processing and artificial intelligence. The resultant dictionaries are appropriate for use in PETRARCH but could be adapted for use with other event-coding software solutions as well.

The new technique makes use of both part-of-speech tagging and named entity recognition but does not utilize triplet extraction. Triplet extraction, while mostly effective for identifying the components of a sentence that map to source actor, target actor, and action, provides no context for structuring the extracted terms into a unified ontology. Furthermore, by foregoing the strict rule-based approach of triplet extraction in favor of neural network models, the solution benefits from estimating its own language model from the provided data. This flexible model facilitates synonym extraction and, more generally, defines a distance metric on the vocabulary space of the corpus.

The new technique is then applied to the cybersecurity corpus to generate an event dataset with high accuracy that substantially outperforms the CAMEO dataset attempt described in this chapter.

## A Semi-Supervised Method for Generating Novel Event-Data

In order to code cybersecurity events with PETRARCH, the existing dictionaries must be either extended or replaced to encompass the new domain. This chapter outlines a new method for producing dictionaries for the event-coding task via a semi-supervised procedure. The problem of dictionary creation is elucidated, then a series of pre-processing steps are outlined and a model for dictionary extraction from text is described. The chapter concludes with a discussion of remaining challenges in automated dictionary creation.

### 3.1 Why automate dictionary creation?

As demonstrated in the previous chapter, the CAMEO coding scheme is not a comprehensive description of public interactions between politically-relevant actors and agents. For researchers interested in types of interaction that do not conform to the existing dictionary structure, creation of new dictionaries is a necessary but costly step. The CAMEO dictionary, for example, contains many thousands of verbs and phrases parsed in a particular format and organized within the predetermined ontol-

ogy. Not only must researchers do this parsing and organization by hand, but they must also begin with a comprehensive list of verbs and phrases that will comprise the dictionary. Historically, the work of identifying verb phrases and classifying them has been done by undergraduate or graduate research assistants. This is time consuming, expensive, and not reproducible. The coding decisions made by research assistants are supposed to follow prescribed rules, but their actual judgments are not auditable.

Streamlining this process offers several benefits. First, automating the dictionary-creation process represents a major step towards fully-automated event-data coding for novel domains. Secondly, because this process can be done largely without human interaction and the content of the dictionaries are a function of the raw data to be event-coded, the dictionaries can be updated in tandem with the event dataset itself; new verb phrases, actors, or agents can be learned by the underlying models as they enter the relevant domain’s vocabulary. Finally, because the process described herein relies on a small amount of initial researcher input data and the raw text data itself, the process of event-data generation is now fully reproducible from start to finish.

### 3.2 A method for creating and extending dictionaries

The process described herein consists of a number of steps. The entire process will be outlined before going into further detail about each step. First, an ontology must be established to describe the domain of interest. In the case of cybersecurity, that ontology consists of sets of relevant event types, actor/agent types, and issue types. Next, the corpus to be coded is parsed using techniques for natural language processing. This is a necessary step for both event coding by PETRARCH as well as the dictionary creation process. The tree parse representations of each news story in the corpus are saved for PETRARCH while the part-of-speech (POS) tags are appended to their respective words for dictionary creation. Additionally, named

entity recognition (NER) is applied to the corpus and entity tags are appended to their relevant words. The entire corpus of POS and NER-tagged words is then phrase-ified using a simple frequency-based model. This results in concatenation of single words into common multi-word phrases. Word2vec is then used to learn a vector-space representation of the entire vocabulary.<sup>1</sup> Seed words and phrases, chosen according to the pre-defined ontology, are then used to extract synonymous and nearly-synonymous words and phrases from the Word2vec model that will populate the dictionaries. Finally, a set of post-processing heuristics are applied to prune and format the dictionaries. While this entire process consists of multiple complicated steps, the researcher is responsible only for supplying an ontology in the form of a small set of seed words and phrases. The process is diagrammed in Figure 3.1. In the last section of this chapter, methods for reducing even this minimal human input are explored. Before detailing each step of the pipeline illustrated in Figure 3.1, Section 3.3 will provide necessary background on a technique that is adopted from natural language processing and artificial intelligence. Sections 3.4 through 3.7 will then detail the steps of the pipeline for automatic dictionary generation.

### 3.3 Word Embeddings

Word embeddings are low dimensional numeric representations of vocabulary that preserve syntactic and semantic relationships between words. Word2vec, a technique developed at Google, is a skip-gram-based model for producing word embeddings, in the form of real-valued vectors, from raw text. Word2vec is based on a single-layer neural network that effectively learns the meaning of words given their contexts in natural language text documents. First, raw text is preprocessed to produce a

---

<sup>1</sup> Training wor2vec on words and phrases that have been tagged with their part-of-speech and named-entity label was a novel development for this application. A similar technique was simultaneously (but independently) discovered by researchers in November, 2015 and named Sense2Vec (Trask et al., 2016).

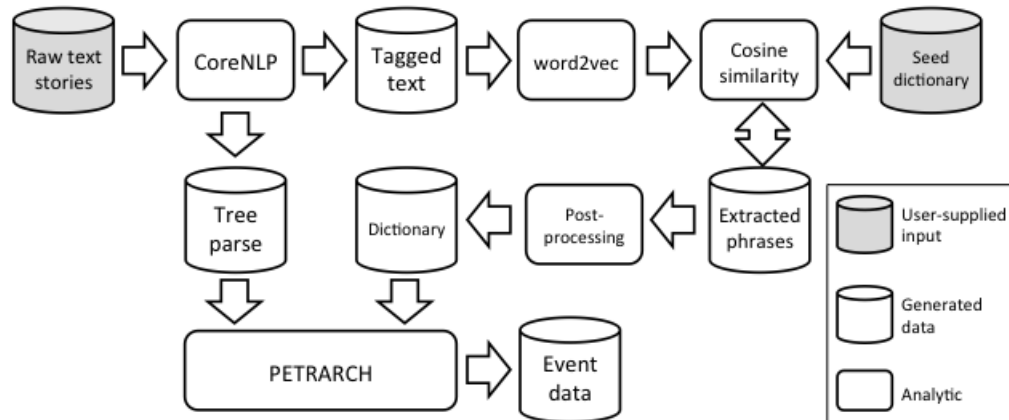


FIGURE 3.1: Diagram of the dictionary learning pipeline

vocabulary of words to learn. For each word in the vocabulary, the model learns the word’s representation based on the context in which that word is found. Word2vec is based on research by Mikolov et al. (2013b). A particularly fast implementation of word2vec in Python was used for this research project (Řehůřek and Sojka, 2010).

While word2vec has quickly become the standard for word embedding applications since its introduction, other techniques also exist. Recent research has used singular value decomposition of a word-adjacency matrix to produce word embeddings (Dhillon et al., 2015). The authors of this research claim comparable or better performance than word2vec on a number of metrics. An extension of word2vec, called doc2vec, estimates vector representations of groups of words in addition to the words and phrases themselves. These “documents” can be full sentences, paragraphs, or larger articles. These skip-gram models build on previous research that has used a variety of machine-learning algorithms to develop language models; these include Latent Semantic Analysis (LSA), Latent Dirichlet Allocation (LDA), and Neural Network Language Models (NNLM). This work is grounded in information retrieval and emerged to solve the problem of variability in word usage in search applications.

LSA was introduced by Dumais et al. (1988) and an overview of subsequent re-

search on the technique is provided by Dumais (2004). LSA assumes that large vocabularies are “noisy” abstractions of an underlying parsimonious vocabulary and that synonyms are actually proxies for a single underlying concept in this latent vocabulary. The latent vocabulary is estimated by applying dimensionality reduction to a sparse matrix representation of a text corpus. LSA proceeds by first producing a document term matrix from a training corpus of text documents. This matrix is often transformed to account for overall term frequency and document frequency (TF-IDF transformed). Singular value decomposition (SVD) is then used to reduce the dimensionality of this matrix and produce a low-dimensional continuous-valued approximation of the original document term matrix. Terms from the original document term matrix correspond to low dimensional vectors on which algebraic operations can be performed to determine “similarity.”

Word2vec, on the other hand, is a particular implementation of an autoencoder. Autoencoders are models that project a high dimensional data point into a lower dimensional representation and then back into the original high dimensional space. Autoencoders are trained such that they minimize a loss function defined as the difference between the original data points and the reconstructed representations of those same datapoints (reconstruction error). In the case of word2vec, the high dimensional space is a count vector of length equal to the number of unique words in the entire text corpus. Every word (i.e. datapoint) in the corpus is represented by a sparse vector where co-located words are indicated by 1. Word co-location is determined by a window of size  $k$ . This representation of a word is called a skipgram. For example, if  $k = 1$ , we can represent the the following sentence in the way shown in Table 3.1: “This is a sentence composed of words.” Therefore, word vectors reflect the context in which a given word appears. Word2vec then produces parsimonious vectors that capture the most informative dimensions of these contexts.

A derivation of the specific flavor of word2vec used here is given by Goldberg and

Table 3.1: Skipgram example

	...	THIS	IS	A	SENTENCE	COMPOSED	OF	WORDS	...
THIS	...	0	1	0	0	0	0	0	...
IS	...	1	0	1	0	0	0	0	...
A	...	0	1	0	1	0	0	0	...
SENTENCE	...	0	0	1	0	1	0	0	...
COMPOSED	...	0	0	0	1	0	1	0	...
OF	...	0	0	0	0	1	0	1	...
WORDS	...	0	0	0	0	0	1	0	...

Levy (2014). There are several methods of estimating word embeddings that all fall under the class of models referred to as word2vec. The most common, and the one recommended by Mikolov et al. (2013b), is called negative sampling. In this model, a feed-forward neural network with a single hidden layer is used to optimize

$$\arg \max_{v_c, v_w} \sum_{(w,c) \in D} \frac{1}{1 + e^{-v_c \cdot v_w}} + \sum_{(w,c) \in D'} \frac{1}{1 + e^{v_c \cdot v_w}} \quad (3.1)$$

where  $v_c$  and  $v_w$  are the word and context vectors;  $D$  and  $D'$  are the given corpus and a random corpus, respectively. Intuitively, the optimization function maximizes the probability of observing the true corpus,  $D$ , given  $v_c$  and  $v_w$ .  $D'$  is a penalty factor introduced to prevent word and context vectors from taking on the same value for all words and contexts, by forcing the optimizer to simultaneously maximize the probability of the observed corpus,  $D$ , while minimizing the probability of the randomly-generated corpus,  $D'$ , given the word and context vectors.

### 3.4 Preprocessing the corpus

The corpus to be event-coded consists of 77,410 news stories related to cybersecurity and technology. These stories are first split by sentence, of which there are 1.14

million.<sup>2</sup> Next, every story is parsed using Stanford CoreNLP.<sup>3</sup> Both PETRARCH and the dictionary creation method described here require that stories be parsed and so the corpus is parsed and the results are saved for later steps. I have elected to use the shift-reduce parser provided by Bauer (2014). The shift-reduce parser is the fastest of those provided by CoreNLP. Additionally, Stanford CoreNLP’s named entity recognizer is used to tag named entities as one of *time*, *location*, *organization*, *person*, *money*, *percent*, and *date* (Finkel et al., 2005).<sup>4</sup>

Once the entire corpus has been parsed and named entities have been identified, two versions of the annotated text are saved. The first version is simply a flat representation of each sentence’s parse tree. These will be saved for input into PETRARCH. The second version of the annotated corpus is formed by appending each word with both its entity-type tag and its part-of-speech tag. For example, the word “hackers” is transformed into “hackers:O:NNS” where “O” indicates that this word is not a named entity and “NNS” refers to a plural noun. “Snowden:PERSON:NNP” indicates that “Snowden” refers to a person and is a singular proper noun. For more on the Penn Treebank POS tags, see Santorini (1990).

The NER and POS-tagged corpus is then processed to produce multi-word phrases. The method for deriving phrases from the corpus is described in Mikolov et al. (2013a). Candidate bigrams (two-word phrases) are scored according to their frequency relative to the frequency of the constituent words being found independently:

$$score(w_1, w_2) = \frac{count(w_1, w_2) - \delta}{count(w_1) \times count(w_2)} \quad (3.2)$$

---

<sup>2</sup> The choice to split articles by sentence gives me more control over how the stories are processed by PETRARCH without having to rewrite the software. PETRARCH’s poor parsing of XML documents often results in large portions of stories going uncoded unless those stories are fed in one sentence at a time.

<sup>3</sup> Stanford CoreNLP is chosen over Apache OpenNLP for compatibility with PETRARCH and also for speed.

<sup>4</sup> *MISC* is also a category for miscellaneous named entities but is not listed in the official documentation from Finkel et al. (2005).

The words  $w_1$  and  $w_2$  are concatenated into a single multi-word term,  $w_1.w_2$ , if  $score(w_1, w_2)$  surpasses a pre-defined threshold.  $\delta$  is a discount factor that prevents spurious phrases from being formed by infrequently-occurring words. In order to produce phrases from more than just two words, this algorithm is run iteratively. After four iterations, phrases of up to five words in length have been formed.<sup>5</sup> An example of this process is given below.

```

These websites could contain specially crafted content that could
  exploit this vulnerability in Internet Explorer.
      ↓
These:0:DT websites:0:NNS could:0:MD contain:0:VB
specially:0:RB_crafted:0:VBN content:0:NN that:0:WDT could:0:MD
  exploit:0:VB this:0:DT vulnerability:0:NN in:0:IN
      Internet:MISC:NN_Explorer:MISC:NNP.

```

POS and NER-tagging each word and phrase in the corpus is necessary to retain sufficient information about each term to post-process the results from word2vec.

### 3.5 Learning the corpus

I use word2vec to learn a vector-space that represents the words and phrases of the corpus after they have been tagged and combined as described above. One benefit of tagging these words before learning them with word2vec is that it affords finer resolution of each word’s meaning given the context in which it is found. For example, “hack” can be used both as a noun and a verb; one can hack into a system or one can exploit a hack. Because the corpus has been POS-tagged previously, word2vec will learn both `hack:0:VB` and `hack:0:NN` as distinct words. According to the model

---

<sup>5</sup> In fact, because multiple multi-word phrases might be combined in subsequent iterations, a small number of phrases longer than 5 words in length may be formed.

itself, the verb form of “hack” is most similar to the phrases “hack into” and “break into” while the noun form is most similar to “attack” and “incident.”

Learning the corpus with word2vec allows us to easily identify synonyms or near-synonyms of our seed words and phrases. First, the element-wise mean vector is computed from the word vectors of each seed word and phrase provided by the researcher. Then, the top  $n$  similar words and phrases are identified by computing the cosine similarity of all word vectors with the mean vector. Cosine similarity, defined as  $(\vec{x} \cdot \vec{y}) / (\|\vec{x}\| \times \|\vec{y}\|)$ , is a measure of the angle between two vectors and is particularly useful when comparing very high-dimensional vectors. Given a seed word or phrase, similar words and phrases can be rank-ordered by their cosine similarity to the seed word in descending order. In order to extract relevant words and phrases from the word2vec model for inclusion in an event-coding dictionary, we must first have a set of seed words and phrases that represent prototypical entries in a known ontology.<sup>6</sup>

### 3.6 Outlining an ontology

Event-coding is the process of applying known labels from an established ontology to relevant news articles. The challenge addressed in this chapter is that of understanding which terms and phrases map to which known labels. The purpose of event-coding dictionaries like those used by TABARI and PETRARCH is to provide an exhaustive list of which terms and phrases map to which labels. In a fully automated event-coding solution, both the ontology and the dictionary could be produced without human intervention. This chapter, however, focuses on the latter challenge: automating the process of synonym and near-synonym extraction and classification given a known ontology. The challenges and possible solutions for automating the former component, ontology generation, are discussed in the conclusion of this chapter.

---

<sup>6</sup> Entirely unsupervised alternatives to this are discussed later.

The top-level of an ontology for event-coding is dictated by the requirements of the coding software, in this case PETRARCH. I have chosen to use PETRARCH I, referred to here simply as PETRARCH, for reasons described in 3.8. PETRARCH’s dictionary structure includes a verb dictionary, three distinct actor dictionaries, an agents dictionary, an issues dictionary, and a discard dictionary. The verb dictionary categorizes verb phrases into the set of predetermined actions described by event data. The three actor dictionaries categorize persons and named organizations by their affiliations (i.e. country, organization type) and their roles related to the domain of interest. These dictionaries also resolve multiple spellings or representations of an entity’s name into a single canonical representation.<sup>7</sup> The default PETRARCH coding scheme provides three actor dictionaries: country-affiliated actors, international actors, and non-state military actors. The agents dictionary describes how to classify unnamed entities. For example, the agents dictionary maps “thief” and “trafficker” to *criminal*. The issues dictionary identifies phrases common to a subdomain of the domain-of-interest to tag news stories with that subdomain’s label. For example, the current PETRARCH issues dictionary tags issues like *foreign aid*, *retaliation*, and *security services*. Finally, the discard dictionary identifies phrases that disqualify sentences or stories from being coded entirely. This helps to remove stories that might be erroneously coded otherwise. For example, sports reporting often uses the language of warfare to describe “victories,” “defeats,” and teams being “destroyed.”

---

<sup>7</sup> An interesting insight related to automated-coding of actors via the process described herein is that canonical actor spellings are no longer important. So long as various alternative spellings of an actor’s name map to the correct higher-level categories, there is no need for the machine to know that those alternative spellings actually refer to the same individual. That said, future research could potentially derive better actor classifications by employing entity resolution techniques to standardize actor name spelling prior to model estimation.

### 3.7 Term extraction and post-processing

For each category described by the ontology, the representative vectors of that category are element-wise averaged.<sup>8</sup> Then, the top  $n_i$  nearest terms to each category <sub>$i$</sub>  are extracted from the word2vec model. Considerations for choosing the correct  $n$  are addressed in Section 3.8. In order to quickly refine the model to better delineate between categories, a sample of the extracted phrases can be coded by the researcher, added to the seeds for the relevant category, and the extraction process can then be repeated. By focusing on hand-coding only those events that are on the boundary between two or more categories, the amount of hand-coding can be minimized while still providing the model with additional information about the characteristics that distinguish the categories.

Extracted terms are then post-processed according to a set of rules associated with the dictionary they are meant to comprise. For example, extracted terms for the verbs dictionary are required to contain at least one verb. Agent and actor dictionaries may be required to contain at least one noun, one proper noun, or one named entity. For this chapter, the following sets of rules have been applied:

- Verb phrases must contain at least one verb (a word tagged VB\_).
- Verb phrases must contain at least one verb above a certain inverse term-frequency value.<sup>9</sup>
- Verb phrases that include either of the words “no” or “not” are omitted.
- Verb phrases that end with “by” have “\$” added to the end and “+” appended

---

<sup>8</sup>  $\|\sum_{\vec{w} \in C} \vec{w}\|_2$  for word-vector  $\vec{w}$  in category  $C$ .

<sup>9</sup> A sample of 10,000 sentences of the corpus is transformed into a document-term matrix. Then, the term frequencies,  $f_{term}$ , are calculated as the percentage of documents that every word appears in.  $1 - f_{term}$  produces the term weights. Verb phrases must contain a verb scored at 0.99 or above to be included. This has the effect of omitting phrases that include only high-frequency verbs such as “was,” “have,” and “been.”

to the beginning in order to to switch the source-target actor. Single-word verb phrases are duplicated and have “by” appended to them in order to catch additional instances of target-verb-source.<sup>10</sup>

- Agents and actors must contain at least one noun (a word tagged NN\_).
- Agent and actor phrases that contain a verb are omitted.
- Duplicate phrases (those assigned to more than one category) are assigned strictly to that category with which they have the greatest cosine similarity.

The remaining phrases are then manipulated to match the format expected by PETRARCH for each dictionary type. This involves, among other things, grouping verb phrases by common verbs and tagging each dictionary entry with a category tag. A sample of each dictionary can be found in Appendix A.

The selected seed dictionaries are those shown in Tables 4.1, 4.2, and 4.3. Due to the necessary pre-processing steps, and especially the POS and NER tagging steps, terms from the seed dictionaries must match the format of the input terms to word2vec. Because the seed terms for each category are conceptually similar to one another, there is often overlap in the extracted phrases. The verb dictionary described in Table 4.1 contains eight categories and 29 words or phrases and extracts 1,615 words or phrases from the model. Removing phrases that do not satisfy the post-processing heuristics further reduces this number to 1,049.

### 3.8 Remaining challenges

While I hope to demonstrate in Chapter 4 that word vectors, with clever pre/post-processing and minimal researcher intervention, can produce functional dictionaries

---

<sup>10</sup> See Section 3.8 for a discussion of alternatives to this simple approach.

for event-coding, several challenges remain. These will be addressed in no particular order in this section.

Choosing  $n_i$ , the number of terms to be extracted from word2vec for category  $i$  is important for controlling the false-positive and false-negative rates of coded events. Extracting too many terms will result in extraneous phrases being coded to the category in question. Extracting too few will result in missed events that should have been coded. Determining the correct cutpoint, however, is difficult. Future research might address this in multiple ways. For example, a large  $n$  may be chosen so that researchers can manually scan the list of extracted terms in descending order of cosine similarity and select a single cutpoint after which the relevance of subsequent terms is deemed low. Alternatively, statistical methods of determining a cutpoint might be applied to minimize researcher effort. Without a “ground truth” dataset, however, this will likely be a difficult task. One approach might be to model the intervals between subsequent ranked terms as a known distribution and then apply a heuristic to find a sufficiently large gap between word vectors that could indicate that subsequent word vectors do not sufficiently capture the original concept. Furthermore, different categories may necessitate different values of  $n$ , hence the subscript on  $n$  when referring to a particular category. For example, there are likely to be more synonyms for “vulnerability” than there are for “zero-day,” as one is a subset of the other.

A related problem is that of pruning the dictionary. Sometimes, phrases extracted by word2vec are ambiguous. For instance, I have found that the phrase “subdomains have been” is often extracted and categorized as “defaced.” While it is actually very likely that the phrase “subdomains have been” should occur in the context of website defacement, this is not guaranteed. One approach to removing these terms from the dictionary would be to require that a more informative verb than “been” or “have” be included in every verb phrase. I have attempted to automatically identify these

“uninformative” verbs by finding the inverse term-frequency of each verb in a large sample of the corpus. Then a threshold is established and a rule implemented that requires every verb phrase to include at least one verb above the threshold. This has the effect of omitting verb phrases that contain only these very common verbs.

For directed-dyad event data, verb phrases should specify source actor and target actor. The method described herein to infer actor order from the presences of the word “by” is not robust. Future research should investigate methods for determining source and target actors from the grammatical structure of a parsed sentence. In fact, PETRARCH II does attempt to infer this information from the parse tree rather than from the verb dictionary (as PETRARCH I does). Unfortunately, other considerations prevented me from using PETRARCH II for this project. In particular, PETRARCH II would require substantial code changes to accept non-Phoenix dictionaries as the Phoenix event codes are hard-coded into the software itself. This hard-coding is such that verbs specific to the Phoenix dictionaries are translated into hex values so algebra can be performed on them. While this system is very clever and reduces the complexity of the required (hand-coded) verb dictionaries, it also prevents PETRARCH II from being extended with novel dictionaries without considerable effort and linguistic expertise.<sup>11</sup> In this paper, the approach I have taken is to simply append a \$ symbol to verb phrases that end with “by.” This has the effect of reversing the usual source and target actor order as the \$ symbol, in TABARI/Phoenix notation, indicates where to look for the source actor.

Further complicating the event-coding process are sentences with multiple verb phrases. Take, for example, the following sentence: “The Head of the IDF cyber defense unit revealed that infiltration had also been attempted on IDF networks, but he verified Israel’s high technological capabilities were elevated in order to ensure

<sup>11</sup> The major revisions to PETRARCH I that became PETRARCH II were undertaken by Clayton Norris, a student of computational linguistics at the University of Chicago. He documents the process of determining source and target actor from text in Norris (2015).

breaches did not occur.” (Tripwire Guest Authors, 2014). This event is coded as the Israeli Military/Government infiltrating Israeli computer networks because PETRARCH cannot disentangle the two verb phrases. Norris (2015) has made progress in solving this problem in PETRARCH II. Unfortunately, the solution is part of the complicated hard-coded hex value system and not easily extended to new dictionaries. It is possible, however, that additional verb phrases could simply be added to the PETRARCH II dictionaries if that existing ontology is appropriate for the research question at hand.

Understanding the resolution of word2vec will help to inform the complexity of ontologies that can be learned in this fashion. The distinction between some types of verb phrases is clearly captured by word2vec while others are less clear. The ability of word2vec to distinguish the conceptual relationships between words is, in part, a function of corpus size and is an active area of research in machine learning and artificial intelligence (Kim et al., 2015; Rong, 2016).

### 3.9 Conclusion

This chapter has outlined a methodology for populating dictionaries for the event-coding task. The method relies on minimal researcher input and can rapidly produce large dictionaries directly from the text corpus for which event-coding is desired. Established techniques from natural language processing as well as cutting-edge methods from machine learning are combined to produce a pipeline that transforms a raw text corpus into a set of dictionaries that can be fed directly into PETRARCH or, with alternative post-processing, other event data coding software. In the next chapter, this process is put to the test and shown to produce accurate event data in a novel domain: cybersecurity.

## Introducing CYLICON, a Cyber Event Dataset

I now apply this novel method for automatic dictionary generation for event coding to a new domain: cybersecurity. As was illustrated in Chapter 2, researchers lack a high-resolution dataset for the study of cybersecurity events. In this chapter, I describe the creation of a new event dataset for this domain, the rationale for the chosen ontology, and investigate the accuracy of resulting data. I will begin by quickly reiterating the process described in Chapter 3 paying close attention to how the methodology was impacted by the particulars of the cyber domain. This will include discussion of decisions made during iteration on the dictionary generation cycle. Once a set of appropriate dictionaries is generated, they are used to produce CYLICON (pronounced “silicon”), the *CYber LexICON* Event Dataset. Next, a thorough side-by-side comparison will interrogate the coded events against the raw text that generated them. Finally, this chapter will conclude with an evaluation of the overall quality of the resulting event data and considerations for next steps in both cybersecurity event data and automated event data generation. The entire data set evaluated here is reproduced in Appendix C. The work in this chapter emphasizes the performance of automatically generated verb dictionaries. While

actors, issues, and agents are also generated and coded, the resulting dictionaries are appended to their corresponding default dictionaries (those supplied with PETRARCH). However, no verb phrases from manually-coded dictionaries are used in this chapter. Therefore, the verb phrases coded herein are wholly representative of the automatically-generated dictionaries while the issues, agents, and actors result from a combination of manual and automatically-generated dictionaries. Actor coding is evaluated in more depth in Chapter 6.

## 4.1 Dictionary Generation Process

First, a cybersecurity ontology is selected and seed phrases are chosen to represent each category of that ontology. Five dictionaries, described in Section 4.2, are generated: verbs, actors, agents, synsets, and issues. After the first round of phrase extraction using a trained word2vec model, a small selection of the most “controversial” phrases are manually-coded into the appropriate categories. That is, 30 extracted words and phrases that are assigned to multiple categories of the ontology are identified and hand-coded to the appropriate categories. The updated seeds are then used to extract a new candidate dictionary. This tuning process is repeated three times with each iteration taking under five minutes. The extracted dictionaries are then passed through a set of filters and post-processing steps to prune poor phrases, replace certain terms with synset labels, distinguish between actors and agents, distinguish between source and target actors when possible, assign duplicated phrases to the most likely category, and to parse the data into the correct format for each dictionary type. No manual changes have been made to the dictionaries after their creation. Simple visual inspection and correction would require minimal researcher effort and could result in better event data. However, it is not reproducible and is therefore not performed here.

## 4.2 A Cybersecurity Ontology

Table 4.1 shows the selected ontology and seed words used to produce the cybersecurity event-coding verb dictionary. Eight categories of events are identified: defacements, DDOS events, infiltrations, leaks, infections, vulnerabilities, arrests, and patches. Extracted phrases will be categorized based on their similarity to the relevant seed phrases. The 29 seed phrases are prototypical examples of verb phrases that represent each category. These are chosen by the researcher. The extracted dictionary contains 1,615 candidate phrases. After post-processing, 1,049 verb phrases remain and comprise the final verb dictionary. These values are a function of the initial number of phrases extracted from the word2vec model per action category. This value is chosen through trial-and-error by the researcher and will likely impact the false positive and false negative rates of the resulting data. Extracting too many phrases from the model will lead to false positive events while extracting too few will cause PETRARCH to overlook events that should not have been. The false positive rate can be mitigated, in part, by careful post-processing steps.

The seed phrases serve as a guide for the type of events that should be represented by each category. For example, the category labeled “arrest” also includes extradition as a seed phrase. The resulting category includes phrases indicative of arrests, extraditions, and other law enforcement actions such as confiscation or asset seizure.

Table 4.2 shows the selected ontology and 14 seed phrases for cybersecurity event-coding actor and agent dictionaries. Actors and agents are combined in the seed dictionary and then parsed in post-processing. This is due, in part, to the fact that the actor and agent dictionaries are, for our purposes, interchangeable. Because each actor and agent will be explicitly tagged with a code from the ontology, the terms can be placed in either the actor or the agent dictionaries. The exception to this is

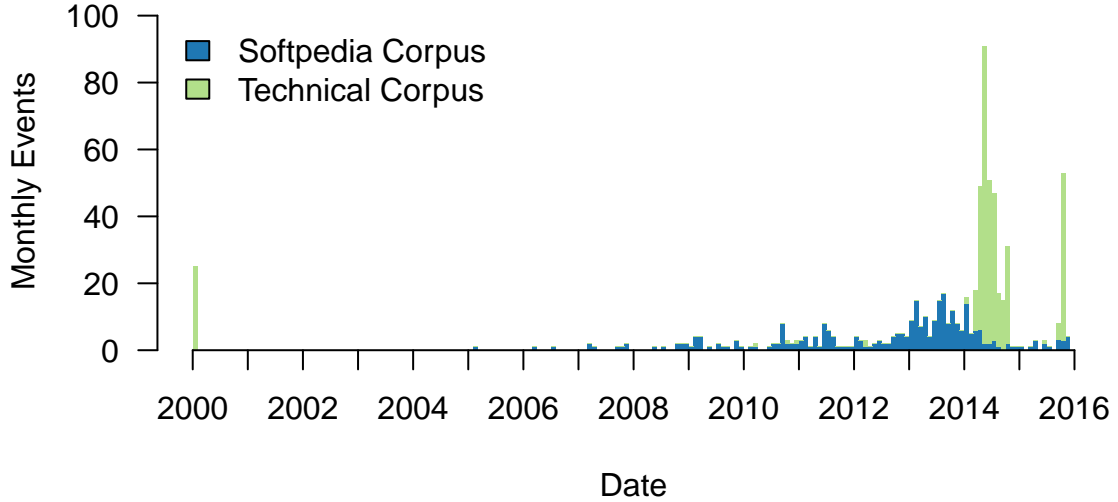


FIGURE 4.1: Cyber events over time

when actors require more than a single tag (i.e. location and role). This is addressed in Chapter 6. The new categories of actors and agents introduced in CYLICON include hackers, researchers, whistleblowers, and antivirus companies/organizations. These categories are appended to the existing actor and agent classifications already found in the default PETRARCH dictionaries.

Table 4.3 shows the selected ontology and 14 seed phrases for issue-coding. Again, these new categories are appended to the issues already supplied with PETRARCH. They include TOR, 0Day, hacktivism, DDOS, social engineering, and state-sponsorship.

### 4.3 CYLICON Event Data Overview

The distribution of events over time in the CYLICON dataset is represented by the stacked bar chart in Figure 4.1. In all, 694 events are represented. 25 events are erroneously coded to January 1<sup>st</sup>, 2000 when actual publication dates are not recorded by the RSS feed. Accurately time-stamped events begin in early 2005 (the date of the earliest available stories from Softpedia). The Softpedia and Technical corpora

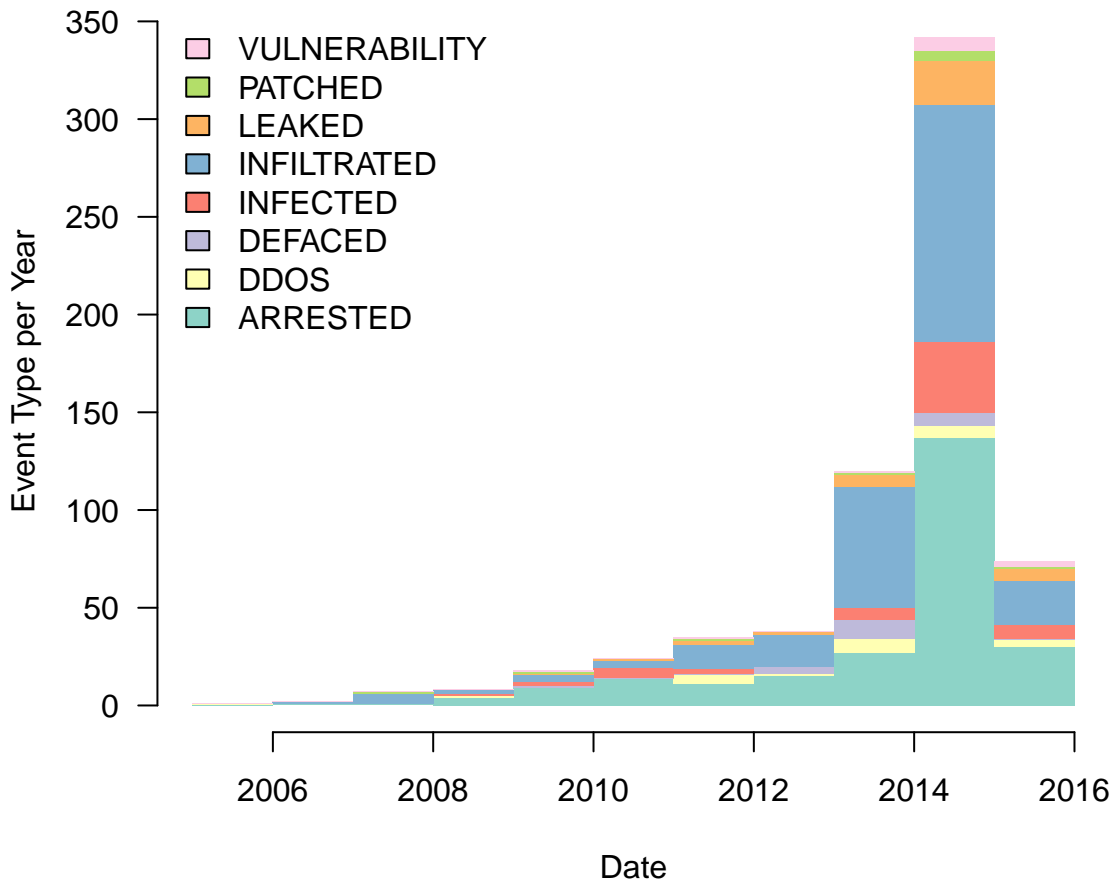


FIGURE 4.2: Cyber events by type

are distinguished since they represent different time periods. By inspecting the Softpedia events over time, an upward trend in cybersecurity-related events is evident from 2005 until 2013, after which events (or reporting) drop off. The Technical corpus represents two distinct news collection periods - one during much of 2014 and another beginning at the end of 2015.

Figure 4.2 shows the distribution of event types in CYLICON from 2005 through 2015. Infiltration makes up the largest category (263 events) followed by arrests (259), infection (61), leaks (45), DDOS (25), defacements (22), vulnerabilities (13), and patches (10). Infiltration is the most common category as many common verb

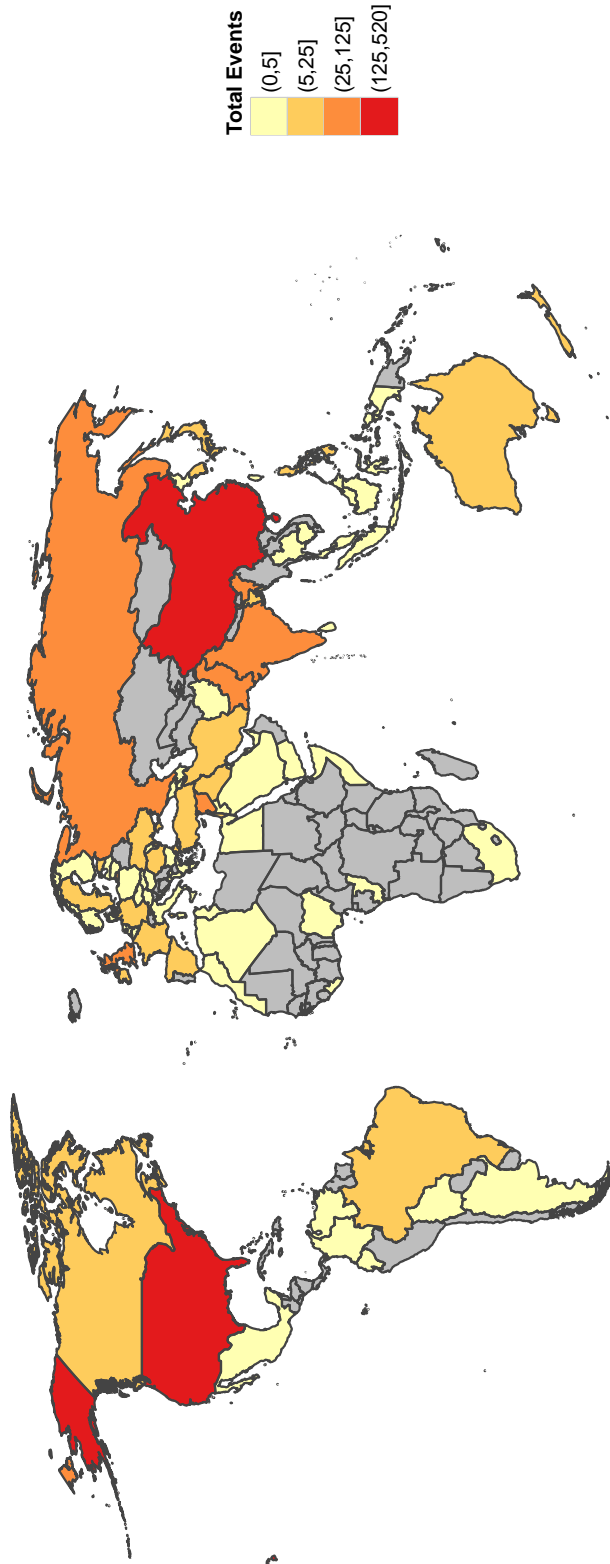


FIGURE 4.3: Spatial distribution of cyber actors

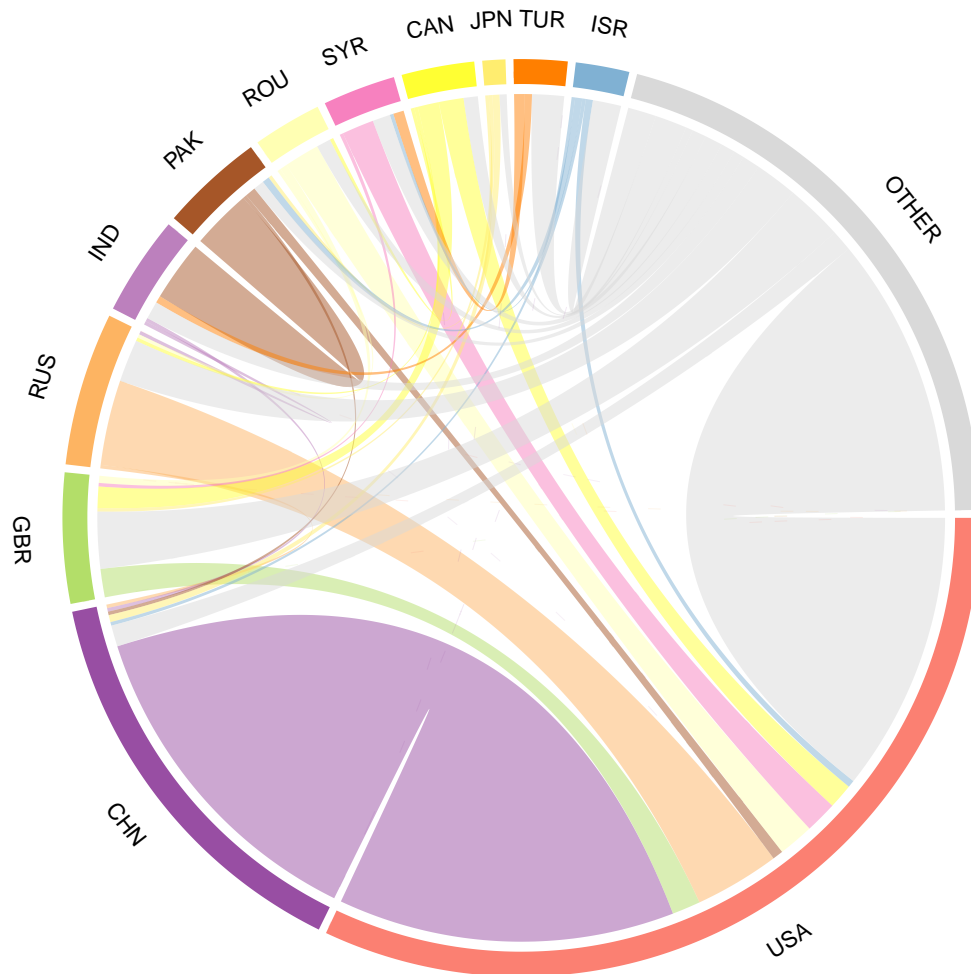


FIGURE 4.4: Top country dyads

phrases from cybersecurity reporting mapped to it; for example, a number of phrases that include the words “breached” and “hacked” are automatically classified as infiltration. Infections and defacements also imply infiltration, and therefore many events could have been accurately coded as either an infiltration or as one of infection or defacement. Generally, reporting on cybersecurity is not precise enough to distinguish the characteristic of a particular “hacking” event in a single sentence. Because PETRARCH relies on all of the relevant information being contained in a

single sentence, a bias towards infiltration coding is induced. In other words, if the coded sentence explains that a target was “hacked” and a second sentence explains that the event resulted in the defacement of the target’s website, PETRARCH will fail to connect the defacement to the hacking event and will therefore code the event as an infiltration rather than a defacement.

The geographic distribution of actors involved in cyberspace is apparent from Figure 4.3. This map corresponds to conventional wisdom about the most active actors in cybersecurity-related activity (The Economist, 2012; Clapper, 2015; Akamai, 2015). However, this map is not representative of the entire CYLICON dataset. Not all relevant actors are geo-coded. Of 1,388 total coded actors, 1,225 are assigned to specific countries. Actors affiliated with international organization or otherwise unaffiliated with specific countries are, of course, not included in the map. For example, as can be seen in Table 4.2, agents and actors specific to cybersecurity are coded as one of hacker, researcher, whistleblower, and antivirus (for antivirus and other cybersecurity-related firms). Chapter 6 investigates a technique for geo-coding these actors and agents but this work is not represented in CYLICON. PETRARCH will attempt to geo-code agents when country-specific keywords are found near the relevant agent phrases but will not do so for actors. Therefore, some of the actors and agents that are found in CYLICON are not geo-coded. The USA is the most prominent actor with 520 events, followed by China (132), Great Britain (66), Russia (47), and India (31). 83 unique countries are represented in all.

Because event data from PETRARCH are dyadic, we can also examine country pair interactions. Figure 4.4 represents the most common dyadic pairs in CYLICON. Chord plots, common in network analysis applications, represent the volume of interaction between nodes or, in this case, countries. This particular chord plot is non-directed and does not include self-connections. The top 12 countries (by volume of events) are plotted and the remaining 71 are grouped into the category “other.”

The larger edges conform to the expectations of Valeriano and Maness (2014); regional pairs and rivals are apparent in the graph. The United States is most active with with China and Russia. India and Pakistan account for the majority of each other’s cyber events. Similarly, Syria interacts primarily with the United States, Turkey, and Israel.

#### 4.4 Example Events from CYLICON

To better illustrate the successes and shortcomings of CYLICON, a selection of events are examined alongside their original text. Event codes are indicated by the triplet `ACTOR1 ACTOR2 ACTION` preceding each sentence. Selected sentences and their corresponding data are enumerated in the list below, beginning with examples of accurate coding and ending with examples of inaccurate coding. Commentary follows.

1. `PAKHAC NGAGOVMED INFILTRATED`: “Pakistani hackers of the Pak Cyber Eaglez group have breached and defaced four Nigerian government websites.” (Kovacs, 2013f)
2. `MYSHAC USAGOVBUS INFILTRATED`: “A Malaysian hacker arrested last year admitted to breaking into a Federal Reserve Bank computer installing malicious code on it.” (Constantin, 2011)
3. `ISR USAELIGOV INFILTRATED`: “According to FBI, in the Year 2000 Israeli Mossad had penetrated secret communications throughout the Clinton administration, even Presidential phone lines.”<sup>1</sup>

---

<sup>1</sup> It is unclear how this particular sentence was found in the Technical corpus since the original article is no longer available. Similar sentences appear in a variety of conspiracy-related opinion pieces online but the original source is elusive. The example is presented here for demonstration purposes as it represents the capabilities of automatic dictionary generation and PETRARCH.

4. GBRCOP IRL ARRESTED: “British police say they have arrested a teenage boy in Northern Ireland in connection with a cyber-attack on British telecoms company Talk Talk.” (Associated Press, 2015).
5. USACOPLEG XXXHAC ARRESTED: “In an interview with The Huffington Post, representatives of the FBI said Anonymous was dismantled after the arrest of the LulzSec hackers.” (Kovacs, 2013a)
6. USACOP EST ARRESTED: “After the Estonian masterminds were apprehended by the FBI, the DNSChanger Working Group was established and the cleaning process began.” (Kovacs, 2012d)
7. PAKHAC ISRMED DEFACED: “On Monday, Pakistani hackers took credit for defacing a number of high-profile Israeli websites, including BBC, Coca Cola, Intel, and several ones managed by third parties on behalf of Microsoft.” (Kovacs, 2012c)
8. MYSHAC PHLHACGOVMED DDOS: “After Anonymous Malaysia launched distributed denial-of-service (DDOS) attacks against several Philippines government websites, Filipino hackers went on the offensive, defacing a large number of commercial websites.” (Kovacs, 2013d)
9. XXXRES XXXRESMED PATCHED: “Methodman has recently disclosed similar XSS vulnerabilities affecting the website of Kaspersky Labs.” (Constantin, 2009)
10. CANMIL RUS LEAKED: “...Canadian naval officer accused of leaking information to Russia faces life in prison.” (Kovacs, 2013g)
11. USAHAC USAMIL INFECTED: “US officials did not provide details on the status of the ”corrupt” software installed on DoD computers, but common sense points us to believe it was removed back in 2013.” (Cimpanu, 2015)

12. XXXANT MNCUSAMED INFECTED: “While the malware was first detected in 2014 and was mainly distributed to users installing Android apps from unverified sources, Bitdefender is now reporting on several instances of the trojan being found distributed via the official Google Play Store.” (Cimpanu, 2012)
13. BGD MED BGD INFILTRATED: “A Bangladeshi publisher of secular books has been hacked to death in the capital Dhaka in the second attack of its kind on Saturday, police say.” (BBC, 2015)
14. IRNGOVGOVMILHAC USA INFILTRATED: “Head of Iran’s Civil Defense Organization Gholam Reza Jalali told the agency that the country never hacked financial institutions from the United States.” (Kovacs, 2012b)

The first three examples are all accurately identified as instances of infiltration by PETRARCH. Item 1 could have been correctly coded as either infiltration or defacement. This is reflected in the phrase “breached and defaced.” Generally speaking (with rare exceptions), defacement implies an infiltration. Actors are coded accurately. One of the defaced sites was that of the National Malaria Control Programme, leading to the erroneous issue coding of DISEASE.<sup>2</sup> Item 2 could have been correctly coded as either infiltration or infection. In this case, the event coder has picked up on “breaking into” and chosen infiltration. Again, the actors are coded correctly. This story was issue-coded with MALWARE, CYBER\_SECURITY, accurately reflecting the “malicious code” hackers planted on the Federal Reserve Bank’s systems. Item 3 is coded correctly as an instance of infiltration. The actors are also accurate, though PETRARCH codes Mossad as ISR rather than ISRSpy. The issue coding for this story, HACKTIVISM, is incorrect. Additionally, the date (April 8, 2014) is incorrect as the event is being related several years after it occurred.

---

<sup>2</sup> The additional text that explains the target website is not shown here. It was appended to the sentence in Item 1 due to imprecise sentence parsing.

Items 4 through 6 are accurately categorized examples of arrests. In Item 5, the source actor is inaccurately labeled LEG, for legislature. PETRARCH confused “representatives of the FBI” with representatives in Congress. The other two labels assigned to the source actor, USA and COP, are accurate. In Item 6, the automated dictionary generation process identified “were apprehended” as indicative of arrest. XXXHAC is from the CYLICON actor dictionary and represents a hacker that has not be geo-coded.

Items 7 through 10 cover defacement, DDOS, patches, and leaks. Items 7, 8, and 10 are unambiguous and are coded correctly. Item 9 is less clear from the text. First, Methodman is a hacker, not a researcher, and so the source actor is incorrect. Additionally, it is unclear from this sentence whether disclosure resulted in a patch or was simply the identification of a vulnerability. These two categories are very similar and it is uncommon for one to be reported in the absence of the other. In this particular case, the event code should have been coded as a vulnerability identification, though Kasperky did quickly patch the vulnerability.

Items 11 and 12 highlight the difficulty associated with coding infection events. Both events are accurately coded as infections since “corrupt software” and “malware” are cited in the text. However, as is often the case with infection events, a source actor is not described. In both cases, the target actors are accurately identified. In item 11 the source actor is coded as hackers from the United States. This may or may not be the case but it is certainly not clear from the relevant text. PETRARCH, at the current time, can only code dyadic events. Without a clear source actor in the text, PETRARCH appears to have interpreted “US officials” as the source of the infection, not the source of the story details. In item 12 an antivirus firm (XXXANT) is identified as the source actor. This is not a wholly inaccurate characterization as the antivirus firm in question, Bitdefender, did report the malware. However, Bitdefender is certainly not the malicious actor responsible for spreading

the malware.

Items 13 and 14 were incorrectly coded. The incorrect coding in item 13 resulted from the dual meaning of the verb “hacked.” It is possible that with a larger ontology, one that includes both computer infiltration and murder, “hacked\_to\_death” would be accurately coded. However, without a method for automatically pruning erroneously-coded phrases from the dictionaries, edge cases like this must be identified and removed by hand. No manual pruning has been performed on these dictionaries and so edge cases remain. Item 14 is incorrectly coded because the sentence itself is a denial of the action that was coded. An Iranian official denies that his country had hacked into financial institutions in the United States but PETRARCH interpreted the sentence to mean that the event had, in fact, occurred.<sup>3</sup>

## 4.5 Evaluating Event Data

The CYLICON event data are evaluated for accuracy by performing an in-depth review of all coded events. The events have been reviewed manually and scored on a rubric to help quantify the efficacy of automatically generated event data dictionaries. Of the 1,049 verb phrases in the CYLICON dictionaries, 174 of them account for all of the coded events. This is a six-fold increase over the size of the seed dictionary.

Actions, actors, and issues are evaluated and categorized as follows. Actions and issues are rated as “inaccurate” (0), “partially accurate” (1), or “accurate” (2). Actors are rated as “both inaccurate” (0), “one accurate” (1), “both accurate” (2), or “both accurate but incorrect order” (3). While a comprehensive set of coding guidelines is provided in Appendix D, the following rules accounted for coding in all but a handful of edge cases.

Actions are considered to be accurate if they describe an event from the original

<sup>3</sup> This is a tricky case since prevailing wisdom is that Iran was, in fact, complicit in the attacks in question (Volz and Finkle, 2016). The point stands, though, that PETRARCH failed to code the event accurately given the relevant context.

ARRESTED	16	6	237
DDOS	5	11	9
DEFACED	2	4	16
INFECTED	12	7	42
INFILTRATED	55	15	193
LEAKED	8	1	32
PATCHED	2	2	6
VULNERABILITY	11	1	1
TOTAL	111	47	536

(0) INCORRECT    (1) PARTIALLY CORRECT    (2) CORRECT

FIGURE 4.5: Event accuracy by category. Numerical values represent the frequency of each category of event by that event’s hand-coded accuracy score. Colors are scaled independently by category. Events from both Softpedia and the Technical corpus are represented.

article. Partial correctness is awarded when the relevant text is ambiguous but an alternative code is strongly implied. When the text is ambiguous but does not strongly imply one code over another, the event is coded correct for either of the potential event codes. Actors are coded correct if at least half of the three-letter codes assigned to them are accurate. If both actors are correct, a code of 2 is assigned. The cases in which the “source” and “target” actors are reversed are assigned a 3. Partial accuracy, when only one actor is coded correctly, results in a score of 1. When neither actor is coded correctly, a score of 0 is assigned. Additional rules apply when verb phrases reference more or fewer than two actors. For brevity, these are not discussed here but will be mentioned on a case-by-case basis in the following sections and can be found in full in Appendix B.

Figure 4.5 presents the results of this review by event category. The values are

counts and the cells are colored by their row-wise percentage. Overall accuracy, the number of correctly-coded events divided by the total number of coded events, is 77%. When partially-correct events are included, this increases to 84%.

Actor accuracy, defined as the total number of events in which the actors are correctly identified as source and target and in which at least one half of all three-letter-codes per actor are correct, is 56%. Relaxing this condition such that the source and target actors can be reversed (in other words, undirected dyads) results in 70% accuracy. At least one actor is coded correctly 98% of the time (including scores of 1, 2, or 3). Again, actors are coded as correct if at least half of their assigned three-letter codes are correct. Accuracy would clearly suffer if a more stringent standard were used (such as 100% accurate per-actor coding).

## 4.6 Remaining Challenges

One clear lesson to come out of CYLICON is the need for more flexible event-coding software. PETRARCH only functions properly when coding dyadic events. In cybersecurity, events of interest are not always dyadic. For example, reports of cyber attacks often do not include attribution. Because of this, many unattributed attacks go uncoded. When attacks are attributed, it is often in sentences adjacent to the one containing a verb phrase. This is because attribution is frequently uncertain and needs to be qualified. A sufficient solution would be to improve the monadic coding capabilities of PETRARCH. A better long-term solution would adapt PETRARCH to code events at the story level rather than the sentence level.

Coding at the story level would also allow for the production of more detailed event data. Cyberattacks, for instance, are often described in simple and ambiguous terms before more details are given later in a news story. For example, the following sentence was coded as a DDOS attack: “Over the past weeks, three major US newspapers have reported being attacked by China, but the countrys officials argue that

the accusations are based solely on some IP addresses” (Kovacs, 2013b). Given the text, DDOS is a reasonable categorization. However, the accompanying text makes it clear this was not a DDOS attack but an infiltration for the purpose of data exfiltration. Event coders that are capable of linking concepts between sentences will help to better identify both actors and details of their actions.

Another issue highlighted by the review of CYLICON data is that of three-actor events. PETRARCH struggled especially with arrests and leaks when more than two distinct actors are named. For example, with regard to leaks, news will often report “*A* leaked documents about *B* to *C*,” where *A*, *B*, and *C* are all actors or agents. Similarly, in the case of extradition (in the arrest category), reports will read “*A* has extradited *B* from *C*.”

In the next chapter, the exploration of CYLICON continues. CYLICON is compared, via a series of analyses, to data from multiple external sources. These cursory examinations seek to reveal how cybersecurity events correlate with events and measures from related domains, those of armed conflict and crime.

Table 4.1: Seed phrases for verb dictionary

Category	Seed Phrase
DEFACED	DEFACED:O:VBD
DEFACED	DEFACE:O:VB
DEFACED	VANDALIZED:O:VBD
DDOS	LAUNCHED:O:VBD_DISTRIIBUTED:O: ...
DDOS	DDOSED:O:VBD
DDOS	DENIAL-OF-SERVICE:O:NN
INFILTRATED	INFILTRATED:O:VBD
INFILTRATED	GAINED:O:VBD_ACCESS:O:NN
INFILTRATED	HACKED:O:VBD_INTO:O:IN
INFILTRATED	BREACHED:O:VBD
INFILTRATED	COMPROMISED:O:VBD
LEAKED	HACKERS:O:NNS_LEAKED:O:VBD
LEAKED	HACKTIVISTS:O:NNS_HAVE:O:VBP ...
LEAKED	DUMPED:O:VBD
LEAKED	LEAKED:O:VBD_ONLINE:O:NN
INFECTED	INFECTED:O:VBD
INFECTED	INFECTED:O:VBN
INFECTED	INFECTING:O:VBG
INFECTED	HAVE:O:VBP_BEEN:O:VBN_INFECT ...
VULNERABILITY	ALLOWING:O:VBG_A:O:DT_POTENT ...
VULNERABILITY	THAT:O:WDT_CAN:O:MD_ALLOW:O: ...
ARRESTED	ARRESTED:O:VBD
ARRESTED	ARREST:O:VB
ARRESTED	EXTRADITED:O:VBD
PATCHED	PATCHED:O:VBD
PATCHED	UPDATED:O:VBD
PATCHED	FIXED:O:VBD
PATCHED	PATCHED:O:VBD_UP:O:RP
PATCHED	FIXES:O:VBZ

Table 4.2: Seed phrases for agent and actor dictionaries

Category	Seed Phrase
HACKER	HACKER:O:NN
HACKER	HACKERS:O:NNS
HACKER	ANONYMOUS:O:NNP
HACKER	LULZSEC:O:NNP
RESEARCHER	RESEARCHER:O:NN
RESEARCHER	RESEARCHERS:O:NNS
RESEARCHER	SECURITY:O:NN_RESEARCHER:O:NN
RESEARCHER	BRUCE:PERSON:NNP_SCHNEIER:PER ...
RESEARCHER	EUGENE:PERSON:NNP_KASPERSKY:P ...
WHISTLEBLOWER	EDWARD:PERSON:NNP_SNOWDEN:PER ...
WHISTLEBLOWER	JULIAN:PERSON:NNP_ASSANGE:PER ...
ANTIVIRUS	AVAST:ORGANIZATION:NNP
ANTIVIRUS	F-SECURE:ORGANIZATION:NNP
ANTIVIRUS	KASPERSKY:ORGANIZATION:NNP

Table 4.3: Seed phrases for issue dictionary

Category	Seed Phrase
TOR	TOR:O:NN
TOR	ONION:O:NNP_ROUTER:O:NNP
0DAY	0DAY:O:NN
0DAY	ZERO-DAY:O:NN
HACKTIVISM	HACKTIVISTS:O:NNP
HACKTIVISM	HACKTIVISM:O:NN
DDOS	DDOS:O:NN
DDOS	DDOS:O:NN_ATTACK:O:NN
DDOS	DENIAL-OF-SERVICE:O:NN
SOCIALENGINEERING	SOCIAL:O:JJ_ENGINEERING:O:NN
SOCIALENGINEERING	PHISHING:O:NN
STATE-SPONSORED	APT:O:NN
STATE-SPONSORED	ADVANCED:O:JJ_PERSISTENT:O:JJ ...
STATE-SPONSORED	STATE:O:NN_SPONSORED:O:VBD

Table 4.4: Seed phrases for synsets

Category	Seed Phrase
VIRUS	TROJAN:O:JJ_HORSE:O:NN
VIRUS	VIRUS:O:NN
VIRUS	MALWARE:O:NN
COMPUTERS	COMPUTER:O:NN
COMPUTERS	SERVERS:O:NNS
COMPUTERS	HARDWARE:O:NN
WEBASSET	WEBSITE:O:NN
WEBASSET	SUBDOMAIN:O:NN
WEBASSET	INTERNET:O:NN
SOFTWARE	SOFTWARE:O:NN
SOFTWARE	WINDOWS:MISC:NNP
SOFTWARE	PROGRAM:MISC:NN
SOFTWARE	APP:O:NN

## Exploring CYLICON

In this chapter, a broader exploration of CYLICON in the context of other conflict and crime data is undertaken. Characteristics of CYLICON are compared and contrasted with similar event, conflict, and crime datasets. Cyberconflict is often described as a subset of each of these domains (among others) and a natural extension to the creation of CYLICON is exploratory work to identify the overlap between these domains and cyberconflict. This chapter is not intended to provide a comprehensive analysis of CYLICON or the other datasets examined but is instead meant to elucidate promising avenues for future research efforts. As such, no theories about the nature of cyberconflict are formalized herein.

### 5.1 ICEWS and CYLICON

A natural starting-point for comparison to external data sources is ICEWS, the Integrated Crisis Early Warning System, a political event dataset that covers (in large part) the same period of time as CYLICON. ICEWS can be obtained, with a one year moratorium, from the Harvard Dataverse (Boschee et al., 2015). ICEWS actually provides two distinct datasets, both of which will be examined here. In addition

to the standard CAMEO-coded daily event data (referred to as events), ICEWS also contains monthly events of interest (EOI). These are considered “ground truth” events representative of political conflict for a given country-month (Lustick et al., 2015). The EOIs include Domestic Political Crisis (DPC), Insurgency, International Crisis, Rebellion, and Ethnic/Religious Violence (ERV).

The difference in volume of CYLICON versus ICEWS events over the period from 2005 until mid 2015 is substantial with 9.1 million ICEWS events versus 602 CYLICON events.<sup>1</sup> This roughly corresponds to 72,000 events per month versus 5 events per month. A number of factors likely contribute to this discrepancy including, but not limited to, the greater number of news sources for ICEWS, the greater breadth of event types in ICEWS, the low probability of dyadic cybersecurity events being reported,<sup>2</sup> and the overall difference in frequency between common events (i.e. statements, verbal cooperation, etc...) in ICEWS and CYLICON. In other words, there are a number of reasons that CYLICON and ICEWS should not be comparable based on volume of events alone. However, despite this, we should expect some correlation between the datasets. For example, those countries that interact most overall might likewise interact most frequently in cyberspace.

### *5.1.1 Geographic distribution of events*

To explore the spatial dimension of cyberconflict, I compare the distribution of countries in ICEWS with the distribution of countries in CYLICON over the same period of time.<sup>3</sup> First, I compare the overall distribution of events across countries in each

---

<sup>1</sup> We lose nearly 100 observations from CYLICON that were either undated or occurred after the last available ICEWS event.

<sup>2</sup> Especially in an event-coder-friendly news format.

<sup>3</sup> Unfortunately, due to an oversight, the three-character abbreviation used to indicate an antivirus firm, ANT, is also used by the International Standards Organization (ISO) to denote Netherlands Antilles. This problem was corrected prior to the analysis presented here. This issue will be addressed in the next iteration of CYLICON.

dataset. Because the count values are on different orders of magnitude and do not necessarily follow a linear relationship, I calculate a Spearman’s rank correlation coefficient<sup>4</sup> on the rank-ordering of the countries by count.<sup>5</sup> The Spearman’s coefficient is a test for monotonicity and, like Pearson’s coefficient, takes on values between -1 (strictly monotonically decreasing) and +1 (strictly monotonically increasing). The data are zero-padded when countries in ICEWS are entirely absent from CYLICON. The distribution of actors in ICEWS and CYLICON achieves a rank correlation of  $\rho = 0.67$  with an estimated  $t$ -score of 14.22 ( $H_0 : \rho = 0$ ). In other words, the most frequently represented countries in ICEWS are, by and large, also the most frequently represented in CYLICON. The usual suspects top both ICEWS and CYLICON: USA, Russia, China, and India. The set of countries that appear very frequently in ICEWS and not in CYLICON includes Kenya, Lebanon, Mexico, Sudan, Uganda, and Vietnam.

Next, I turn to country-dyads. Dyads capture the interaction between countries and, as such, represent a joint distribution over the country-space as opposed to the marginal distributions of countries described above. Because country-dyads are likely to be sparse (some country-dyads will never appear in ICEWS or CYLICON), a master list of possible undirected-country-dyads is constructed. The country-dyads from ICEWS and CYLICON are then merged with this master list and missing country-dyads are zero-padded. As before there is a positive, but weaker, rank correlation between CYLICON and ICEWS with  $\rho = 0.36$ . There are notable differences between common dyads in ICEWS and CYLICON. The top dyads in ICEWS are primarily “domestic” events in which the source and target countries are the same. In fact, 62% of events in ICEWS are domestic; only 29% of events in CYLICON are.

---

<sup>4</sup> Spearman’s rank correlation coefficient for data that includes tied ranks is  $\rho = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$ . In fact, Spearman’s  $\rho$  is Pearson’s coefficient applied to ranked data.

<sup>5</sup> For the curious, I also calculated a Pearson’s coefficient on the original data and found  $\rho_{XY} = 0.61$  with a 95% confidence interval of (0.53 – 0.69).

The top ten dyads in each dataset can be seen in Tables 5.1 and 5.2

Table 5.1: ICEWS top dyads

Dyad	Count
India - India	518671
Australia - Australia	196133
Russian Federation - Russian Federation	150920
China - China	104875
Israel - Palestine, State of	102725
Philippines - Philippines	101833
United Kingdom - United Kingdom	81357
Japan - Japan	76045
Russian Federation - United States	71479
China - United States	71043

Table 5.2: CYLICON top dyads

Dyad	Count
United States - United States	85
China - United States	83
Russian Federation - United States	24
India - Pakistan	17
Romania - United States	10
China - China	9
Syrian Arab Republic - United States	8
Canada - United States	7
Iran, Islamic Republic of - United States	7
United Kingdom - United States	7

That the countries (and dyads) associated with actors in ICEWS and CYLICON are positively correlated indicates that the underlying processes that drive political events may also drive cybersecurity events. However, reporting bias in event data may also play a role. Weidmann (2015) cautions researchers against relying on event data, especially in conjunction with data on information and communication technology, without also considering the media generation process. CYLICON and ICEWS rely on English language news feeds which may over-report events from certain countries or dyads.

### 5.1.2 *Events of Interest*

ICEWS EOIs represent “ground truth” about the state of political conflict on the country-month level. These are less likely to suffer from the reporting bias that afflicts automated event coders since the data are manually-curated by subject matter experts. By combining these data with a monthly aggregate of CYLICON data, we can better understand the relationship between cyberconflict and “real world” conflict.

Unfortunately, ICEWS EOIs do not include the United States and, therefore, we must omit a large portion of CYLICON events. I begin by testing the simple hypothesis that countries suffering from more EOIs are also likely to experience more cybersecurity events. Because EOIs are defined on monads, not dyads, CYLICON events are aggregated to the country-month level.<sup>6</sup> A Poisson regression model of cybersecurity event counts on EOIs with country and year random effects allows us to identify correlations between EOIs and cybersecurity events while controlling for country-level confounds. A model summary is shown in Figure 5.1.<sup>7</sup>

Rebellion is the only EOI with a distinguishably non-zero correlation with CYLICON events. This holds true at  $\alpha = 0.05$  after Bonferroni correction. The coefficient value of rebellion, 1.69, corresponds to an expected rate ratio of 5.4. A country-month during which there is a rebellion should witness 5.4 times more cybersecurity events than a country-month without rebellion. In practice, however, these would often be very low values due to the extremely low baseline rate for country-month incidence in CYLICON. Figure 5.2 visualizes the predicted counts for Pakistan in

---

<sup>6</sup> In the following analyses, the entire CYLICON dataset from 2005-2014 is used. The analyses are also replicated using only those events scored “accurate” or “partially accurate” in Chapter 4. The results are, for all intents and purposes, identical and so not presented here.

<sup>7</sup> Collapsing the dependent variable to a binary indicator, at least one cybersecurity event per country-month (1) or none (0), and using a random effects logistic regression model yields substantially similar results.

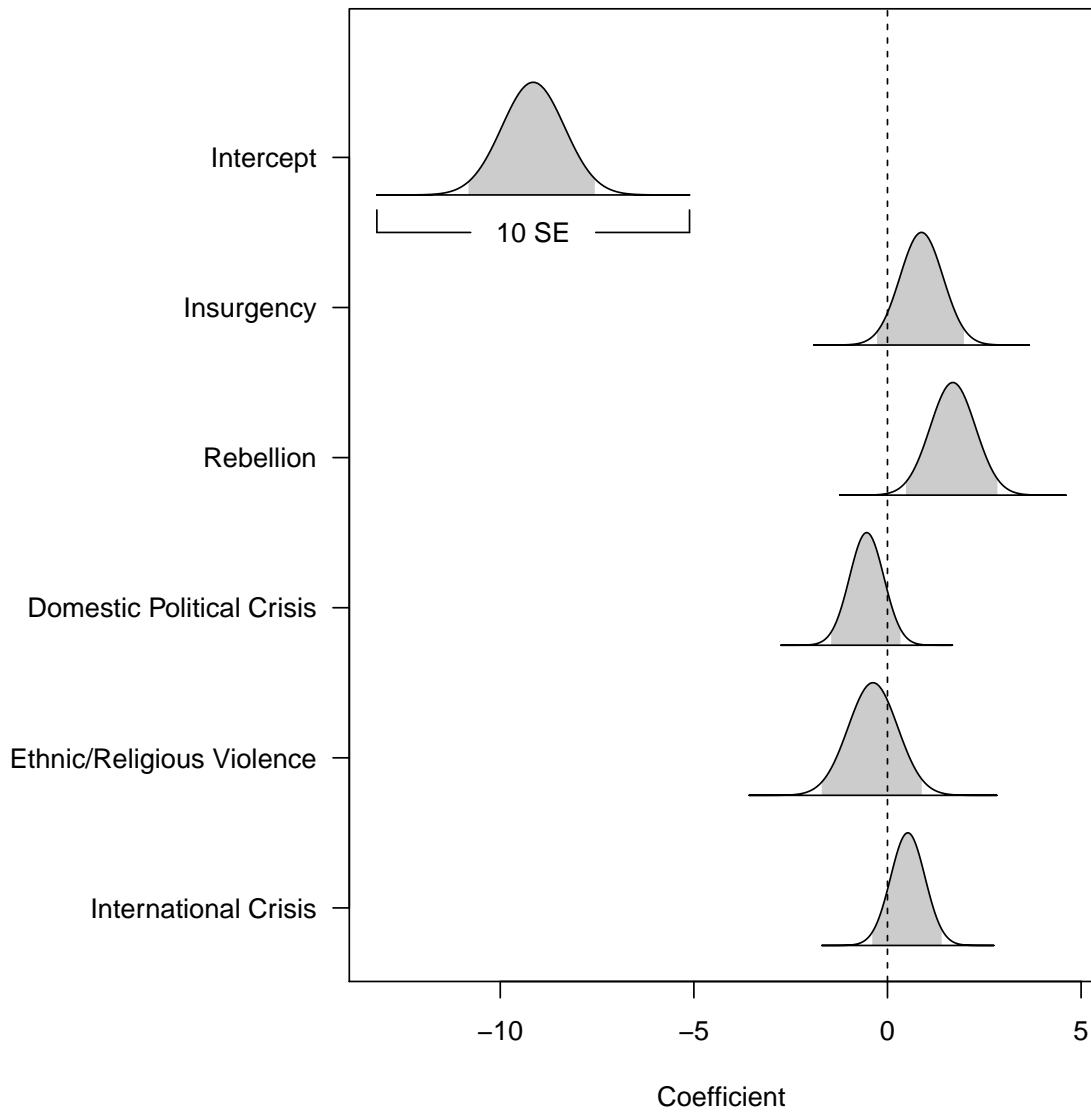


FIGURE 5.1: Coefficient plot for CYLICON event counts regressed on EOI indicators. Shaded regions indicate 95% confidence intervals. Only the intercept and rebellion EOI are significant predictors of CYLICON event counts at the 5%  $\alpha$ -level. Both remain significant at this level after Bonferroni correction. Random effects not shown.  $N = 26082$ ,  $AIC = 775.5$ ,  $BIC = 840.8$ ,  $\log L = -379.7$ .

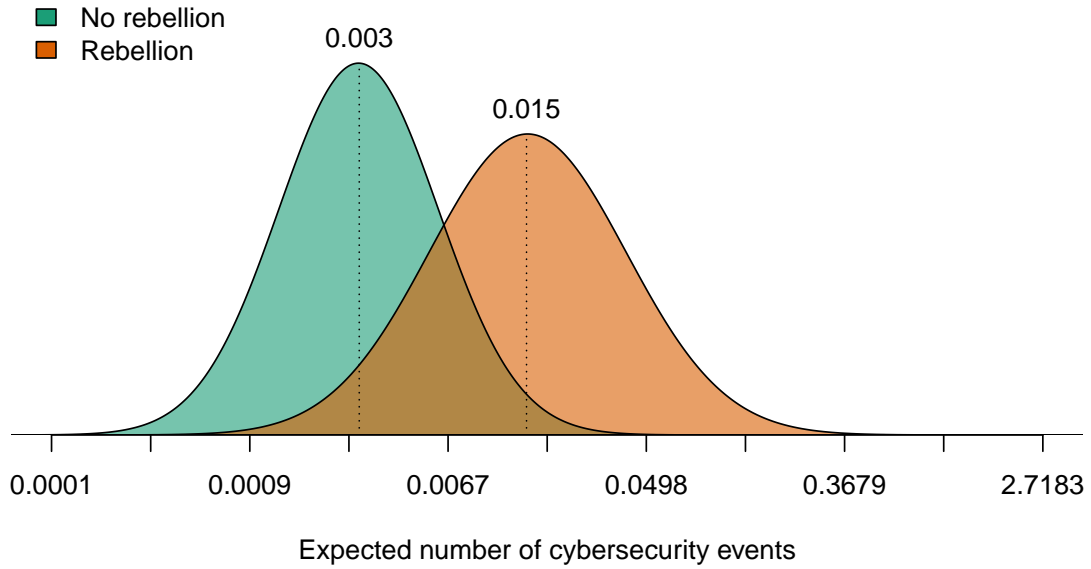


FIGURE 5.2: Predicted number of CYLICON cybersecurity events for Pakistan in (1) a non-rebellion month and (2) a rebellion month. The predicted values have been log-transformed and the x-axis has been modified accordingly. Distributions are based on simulations with 5000 sets of coefficients drawn from a multivariate Gaussian distribution to incorporate model uncertainty.

months with and without an rebellion EOI.

To better understand this finding, those country-months with non-zero CYLICON counts and ongoing rebellion are considered. This is to rule out the possibility that a single outlier is driving the results. For instance, the Syrian Electronic Army received a large amount of press during the Syrian civil war and may fully explain this result. However, the full set of countries experiencing both insurgency and cybersecurity events simultaneously includes India, Indonesia, Pakistan, Philippines, Russian Federation, and Turkey. A more thorough examination of this finding should account for, at a minimum, national economic status, since the relationship between domestic conflict and wealth is well-documented as is the relationship between wealth and technology (Fearon and Laitin, 2003; Collier et al., 2005; Lee and Becker, 2014; Erumban and Das, 2016).

It is possible that converting dyads in CYLICON to monads has overweighted (effectively double-counted) events that involved pairs of these countries (like Pakistan and India). Because PETRARCH does not code event location, just actor location, a better coercion of the CYLICON dyads is unavailable.<sup>8</sup> To get a better sense of how CYLICON corresponds with ground truth conflict data at the dyad level, I now turn my attention to an analysis of cyber conflict among state-dyads experiencing armed conflict.

## 5.2 UCDP/PRIO and CYLICON

The UCDP/PRIO Armed Conflict Dataset covers all armed conflicts with at least 25 battle-related deaths and at least one involved government from 1946 through 2014 (Gleditsch et al., 2002; Pettersson and Wallensteen, 2015). Both the UCDP/PRIO and CYLICON data are subset to 2005-2014 and aggregated to the country-dyad-year. These are merged with a comprehensive set of all country-dyad-years generated from the *cshapes* package in R (Weidmann et al., 2010).

In coercing UCDP/PRIO to dyads, all states involved on Side A are crossed with states involved on Side B. If the conflict type is domestic, the dyad is constructed such that it is Side A - Side A.<sup>9</sup> Because I do not distinguish, in this analysis, between government and non-government actors in CYLICON, it is appropriate to construct domestic dyads in this way. This results in over 18,000 dyads with roughly ten annual observations each. Three types of armed conflict occur during this time period: international, internal, and internationalized internal (Themner, 2015). Indicators for each are included in the model. The response is the monthly count of CYLICON

---

<sup>8</sup> Not to mention the fact that even the manual geo-coding of cybersecurity events would be a substantial challenge in and of itself.

<sup>9</sup> In domestic conflict situations, only the primary state on Side A is used for Side B. Supporting states on Side A are not assigned to Side B.

events per dyad.<sup>10</sup>

The model is depicted in Figure 5.3. To isolate the relationship between armed conflict events and the frequency of cybersecurity events, a Poisson regression with year and dyad random effects is utilized.<sup>11</sup> These random effects should help to account for dyad or year-specific confounds that are not explicitly modeled. As with ICEWS, a positive relationship exists between certain dyads involved in armed conflicts and the number of cybersecurity events between the states in those dyads. The coefficient on internationalized internal conflict corresponds to an expected five-fold increase in the number of cybersecurity events within dyads involved with those types of conflict. After Bonferroni correction, this effect remains distinguishable from zero. The coefficient on international conflict is also distinguishable from zero and indicates an expected 70% decrease in the number of cybersecurity events within a given dyad-year. Caution should be taken when interpreting this result as only four dyad-years were involved with armed international conflicts during the time period in question: India - Pakistan (2014), Djibouti - Eritrea (2008), South Sudan - Sudan (2012), and Cambodia - Thailand (2011). Internationalized internal conflict, on the other hand, occurs in 790 dyad-years during this period. The expected number of cybersecurity events in a dyad-year for varying conflict statuses is illustrated in Figure 5.4.

Between ICEWS EOIs and UCDP/PRIO, the results presented in this and the previous section point to a positive relationship between cybersecurity events and violent conflict (notwithstanding the negative correlation between international conflict and CYLICON counts). Whether this is a causal link, violent conflict promotes cyberconflict, or a correlation due to an unidentified confound, is an open question.

---

<sup>10</sup> Once again, a logistic model is also estimated with a binary response indicating the existence of at least one cybersecurity event.

<sup>11</sup> Alternatively, tensor methods like those employed by Minhas et al. (2015) may be used to model the dyadic state system as a network that evolves over time.

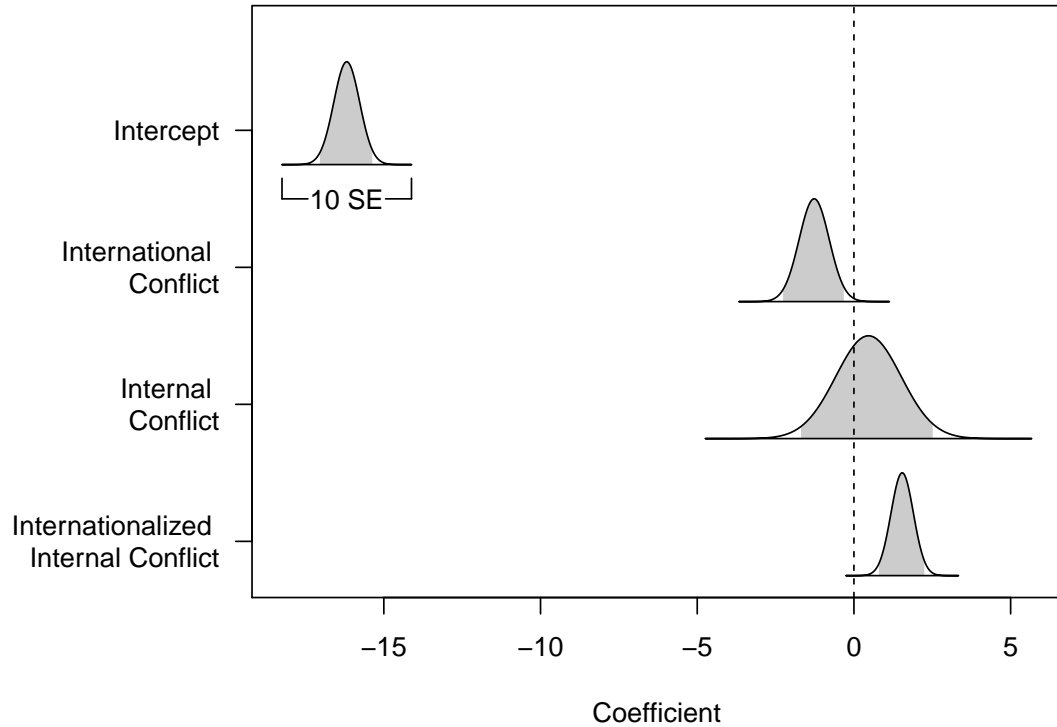


FIGURE 5.3: Coefficient plot for CYLICON event counts regressed on UCDP/PRIO armed conflict indicators. Shaded regions indicate 95% confidence intervals. Random effects not shown.  $N = 181263$ ,  $AIC = 2173.5$ ,  $BIC = 2234.1$ ,  $\log L = -1080.7$ .

### 5.3 Crime and CYLICON

At the national-level there is evidence that cybersecurity events and armed conflict are correlated. Next, I explore the possible relationship between cybersecurity events and crime. Do countries with greater crime problems also find themselves experiencing more cybersecurity events?

Crime data are provided by the United Nations Office on Drugs and Crime (2016) (UNODC). Data on homicides, theft, motor vehicle theft, and assault for the years 2005 through 2014 are investigated here. Figure 5.5 illustrates the pair-wise correlations between CYLICON event types (including all types combined) and the four selected crime statistics. CYLICON events are summed over time per country. UNODC rates (per 100,000 population) are averaged over time per country. As before,

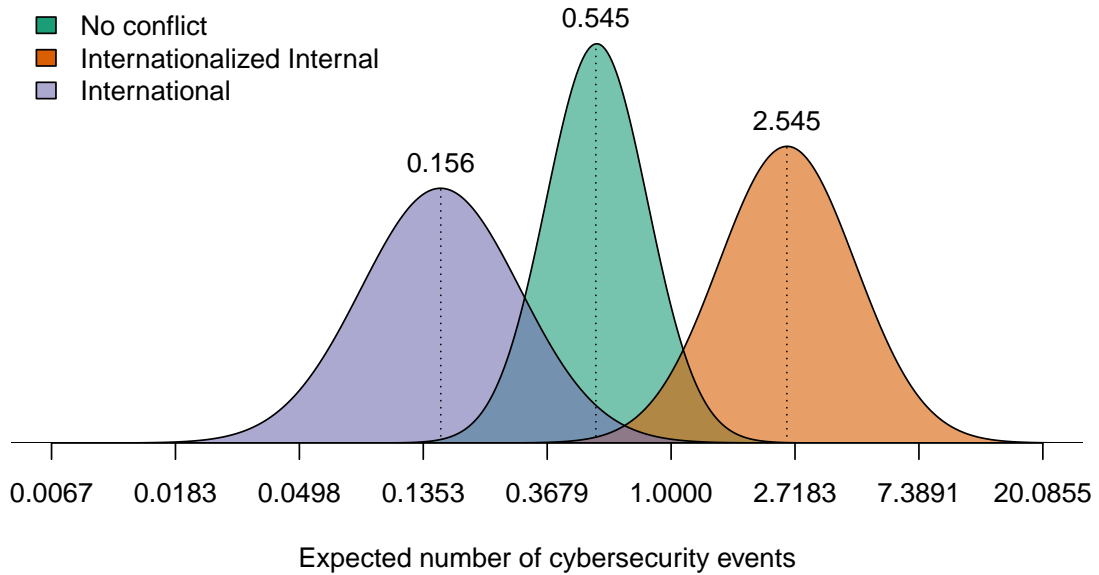


FIGURE 5.4: Predicted number of CYLICON cybersecurity events for India-Pakistan dyad in (1) a non-conflict year, (2) during an ongoing internationalized internal conflict, and (3) during an international conflict. The predicted values have been log-transformed and the x-axis has been modified accordingly. Distributions are based on simulations with 5000 sets of coefficients drawn from a multivariate Gaussian distribution to incorporate model uncertainty.

Spearman’s rank correlation coefficient is selected. In no pair does the correlation exceed 0.31 in magnitude. This indicates, at best, a relatively weak correlation between national crime levels of various types and aggregate levels of cybersecurity events.

While the aggregate statistics do not reveal a strong relationship between national crime rates (of the selected types) and cybersecurity events, perhaps a more nuanced time-series cross-sectional approach will. At the country-year level, there is a substantial amount of missingness in the crime data. Amelia II is used to impute missing values. Because values are unlikely to be missing at random, a variety of additional covariates are obtained from the World Bank’s World Development Indicators (WDI) prior to imputation (Honaker et al., 2011). These indicators include

All	-0.308	0.053	0.014	0.218
Vulnerability	-0.054	0.152	0.046	0.056
Patch	-0.069	0.128	0.000	0.140
Leak	-0.033	0.092	0.089	0.121
Infiltration	-0.258	-0.007	-0.034	0.059
Infection	-0.089	0.033	-0.061	0.071
Defacement	-0.157	-0.057	-0.156	-0.013
DDOS	-0.174	0.045	-0.062	0.148
Arrest	-0.284	0.127	0.103	0.308
	Homicide	Assault	Theft	M.V. Theft

FIGURE 5.5: Country-level Spearman’s rank correlation between mean crime-rate level (per 100,000 population) and CYLICON event rates. Possible values include  $[-1,1]$  with 0 indicating no correlation.

measures of wealth (GDP per capita), unemployment (total % unemployed), education (spending as % of GDP), and adult literacy, among others (The World Bank, 2016). The following analyses are performed across fifty imputed datasets. Results have been aggregated according to Rubin (2004).

Two crime measures have been selected: *homicide rate* from UNODC and *ICFRM-CRM-CRIME8* (Percent of firms identifying crime, theft and disorder as a major constraint) from WDI. These two were selected both for their pre-imputation coverage as well as to account for both violent crime and non-violent crimes that affect business. A third variable is constructed from all five available crime measures. The five measures are projected onto their dimension of greatest variance via

principal component analysis (PCA) and their values along this dimension, the first principal component, are substituted for the crime variable. This new latent variable accounts for just under 40% of the variance in all five original crime measures.<sup>12</sup> Each of the eight CYLICON event types is regressed on all three crime measures individually. Additionally, the overall count of CYLICON events is regressed on each crime measure. This results in 27 total models (9 CYLICON types  $\times$  3 crime measures) estimated 50 times each. All models described below are generalized linear mixed effects models with Poisson response, country, and year random effects as described in 5.1-5.7.

$$y = t \times \exp(X\beta + Z\gamma) \tag{5.1}$$

$$t := \text{Offset value (expected count)} \tag{5.2}$$

$$y := \text{Response (e.g. Infiltration)} \tag{5.3}$$

$$X := \text{Fixed effects matrix (e.g. homicide rate or IC-FRM-CRM-CRIME8)} \tag{5.4}$$

$$\beta := \text{Fixed effects coefficient vector} \tag{5.5}$$

$$Z := \text{Random effects matrix (i.e. year and country indicators)} \tag{5.6}$$

$$\gamma := \text{Random effects coefficient vector} \tag{5.7}$$

The models are summarized in Figure 5.6. Very little evidence is found of any correlation between cybersecurity incident counts in CYLICON and aggregate measures of crime on the country-year level. In fact, the only coefficient to be significantly distinguishable from zero at traditional  $\alpha$ -levels is that of the relationship between

---

<sup>12</sup> For completeness, a maximum likelihood factor analysis (FA) model is also used to estimate a latent crime variable. This variable differs from the PCA variable in interpretation: the five measures are projected on to their dimension of greatest common, or shared, variance. This dimension accounts for approximately 25% of the covariance among the original measures. The disparity in explained variance between PCA and FA is due to the fact that FA incorporates an error term that represents variance unique to each of the original variables. Nonetheless, the results from both latent variable approaches are substantially the same. The FA results are omitted in favor of the PCA analysis.

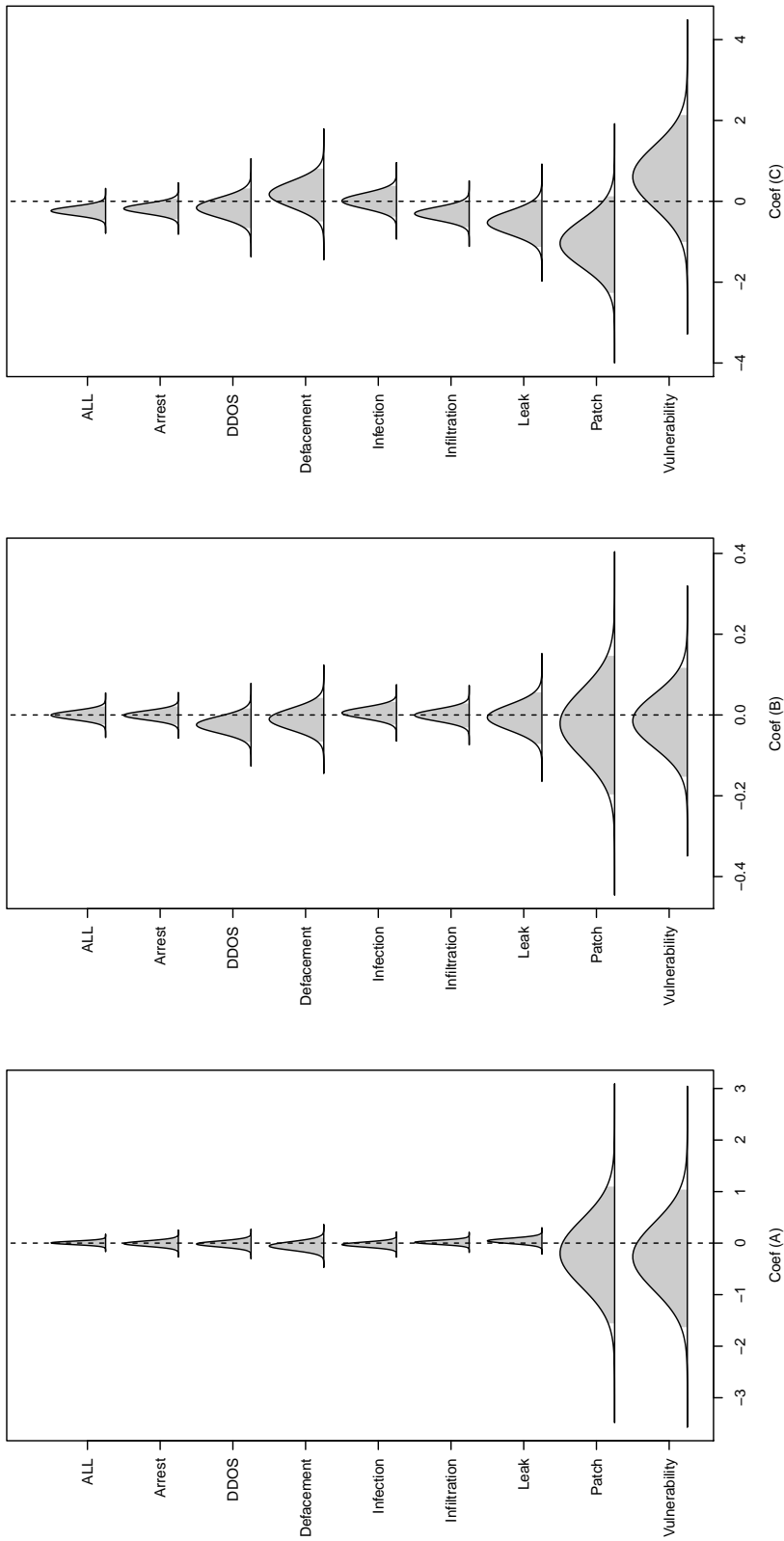


FIGURE 5.6: CYLICON event counts regressed on (A) homicide rate per 100,000, (B) IC-FRM-CRM-CRIMES, and (C) latent crime rate. Coefficients are  $\log(\text{rate ratio})$ . Shaded regions indicate 95% confidence intervals. Intercepts and random effects not shown.  $N = 2450$ .

latent crime and all CYLICON events aggregated. However, this effect does not withstand a Bonferroni correction for multiple hypothesis testing. In practice the effect is relatively large. For a given country-year, a shift in the latent crime dimension from the minimum observed value to the maximum observed value results in a 95% reduction in expected cybersecurity incidents.<sup>13</sup> A shift of this magnitude seems unlikely, though. A shift of one standard deviation along the latent dimension corresponds to a 28% decrease in expected cybersecurity events.

Between both the ranked correlation tests and the regression models, little evidence of a relationship between crime rate and cybersecurity events is identified. Perhaps the non-cyber crimes reported by managers that make up IC-FRM-CRM-CRIME8 are substantially more common than cybercrime, which gets lost in the noise. Or, only a few very significant cybercrime events surpass the threshold necessary to be reported in the news sources from which CYLICON is currently drawing. Alternatively, the data generating processes that produce crime for which data are readily available are mostly distinct from the processes that lead to cybersecurity events.

## 5.4 Conclusion

In this chapter, several popular data sources have been compared with CYLICON. In ICEWS and UCDP/PRIO, evidence is found of a relationship between armed conflict and cybersecurity events. The nature of this relationship requires further investigation, but substantial and positive correlations exist between cybersecurity event counts and countries/dyads experiencing rebellion or other armed conflict. An exception to this is dyads involved in international conflicts, which experience fewer cybersecurity incidents in expectation. This result is based on an extremely small

---

<sup>13</sup> The coefficient value is -0.236. The minimum and maximum latent crime values are -8 and 4.4, respectively.  $1 - e^{(-0.236 \times 4.4)} / e^{(-0.236 \times -8)} \approx 0.95$

sample and should be interpreted with caution.

On the other hand, a thorough data mining effort failed to turn up any substantial evidence that cybersecurity incidents are correlated with measures of crime at the national level. Taken together, these results could indicate that newsworthy cybersecurity events tend to be those that are generated by similar process to international conflict, not crime. However, it is also possible that these findings are spurious or are due to biases in the generation of CYLICON.

With finer-grained data on crime and conflict and an updated version of CYLICON that draws from a wider range of news sources, one should be able to verify the findings here. By better understanding whether common underlying data generating processes produce both cybersecurity events and conflict (or crime), we may begin to produce predictive models of these incidents. For example, those same predictors that are useful in forecasting EOIs may help to forecast cyberattacks or cybersecurity incidents may portend armed conflict.

Future work should also carefully consider the types of crime that could be related to cybersecurity incidents. In particular, understanding the relationship between organized crime and cybersecurity is important for combating both large-scale cybercrime as well as nation-state actions in cyberspace. Are certain types of organized crime groups more or less likely to engage in cybercrime? Can our understanding of organized crime groups help us to understand when states tolerate or punish hacker groups?

## Automated Classification of Actors for Event Coding

Dictionary creation for event data coding is not a single problem but a series of related problems. These include, but are not limited to: term and phrase extraction, ontology learning, term and phrase classification, synonym and alternate nomenclature identification, and term and phrase post-processing. These steps are complicated by the fact that event-coding dictionaries must distinguish between subjects, objects, and predicates, all of which may require a unique set of considerations.

This chapter investigates the use of word embeddings, skipgram-based numerical representation of words as concepts, for the automated classification of terms and phrases. More specifically, given a set of domain-specific terms and multi-word phrases, word vectors are used to place those words within an existing dictionary structure. The example used here builds on the PHOENIX country-actor dictionary. This dictionary is used by event data coding software to identify country-level actors and their associated countries in raw text. In this paper, I assume that a list of actors and a list of countries is available to the researcher, possibly as the result of

an earlier named entity recognition task, but that the associations between actors and countries is unknown. I attempt to then classify actors by associated country using both an unsupervised and a supervised approach.

The results are promising and I demonstrate that unsupervised learning techniques can be used to associate actors with the correct country between 30% and 64% of the time depending on the size of the text corpus in question. The supervised learning approach performs somewhat, though not dramatically, better. These results, while not yet accurate enough to produce event data coding dictionaries without even minimal human involvement, demonstrate that automatic dictionary creation for data of interest to political scientists is a tractable problem.

This paper proceeds as follows. First, a brief background of event data and data-coding methods in political science is given. Then, machine learning methods central to this paper’s technique are described. Next, the approach to actor classification for the dictionary creation problem is outlined. The data is subsequently summarized and model results are discussed. The paper concludes with directions for future efforts in unsupervised data collection.

## 6.1 Approach

Machine-coded classifications are tested against a human-coded dictionary. First, a word2vec model is trained on a corpus of news stories. Then, an existing political event-coding dictionary is parsed and those actors that are present in both the word2vec model and the dictionary are extracted. Then, the actors’ learned representations from the word2vec model are compared to the learned representations of the potential categories (e.g. location names) using cosine similarity. The top matches for each actor are selected and compared to the “ground truth” classification from the original dictionary. In this way, the accuracy of the unsupervised classification method can be judged. This technique is then compared to a supervised learning

Table 6.1: ICEWS data summary

	Documents	Learned words and phrases
<i>ICEWS90</i>	570,488	682,324
<i>ICEWS183</i>	1,210,483	1,136,519
<i>ICEWS365</i>	2,441,345	1,793,776
<i>ICEWS730</i>	5,058,635	2,605,372

extension. Instead of comparing actor vectors with known location vectors, k-nearest neighbors is used to cluster uncategorized actors with known actors.

While this is an unsupervised classification technique, there are some caveats to address. First, the classification problem itself is unsupervised but both the categories and the actors must be known a priori. As mentioned earlier, both the learning of categories (or ontologies) and the extraction of relevant terms and phrases are related but distinct problems. The classification problem can be conceptualized as a special case of k-nearest neighbors clustering. The distance metric employed is cosine similarity and the cluster centroids are known. The algorithm partitions the vector space into regions whose borders are the exact midpoints between each of the categories’ vectors. Actors are then assigned the label of the nearest category’s vector rather than that of their nearest neighbor.

## 6.2 Data

The PETRARCH country-actor dictionary, provided by the Open Event Data Alliance, is used for validation. The dictionary is pruned and all dictionary entries that do not appear in each model are removed. This is done so that the test vocabulary represents named entities that could conceivably be extracted from the corpus.

Word embedding models are very sensitive to both corpus size and corpus pre-processing techniques. For this reason, multiple corpora are selected to train word2vec models and the results are subsequently compared. First, a model provided by Google and trained on the entirety of Google News content is used to prove the viability of

this method. Then, a series of models trained on news data from ICEWS are examined to test the feasibility of this method for researchers unable to obtain a corpus as thorough as Google’s.

Google provides a pre-trained model that was built with the entirety of the Google News corpus. This model was trained on 100 billion words and resulted in a dataset describing 3 million unique words and phrases as vectors of length 300 (Google, 2015). Google also provides a word2vec model specific to named entities from the Freebase dataset. While this would be an ideal starting point for the entity categorization project described here, Freebase is no longer updated and alternatives are not yet available. Since using the entity-based word2vec model would not be sustainable as the dataset becomes more and more outdated, the choice is made not to use this data here.

ICEWS provides a corpus of news documents from which novel word2vec models can be trained. The entire ICEWS corpus for 2013 and 2014 is downloaded and subsets are used to train four unique models. These models cover 90 days of news, 183 days of news, 365 days of news, and 730 days of news. I will denote these models as *GoogleNews*, *ICEWS90*, *ICEWS183*, *ICEWS365*, and *ICEWS730* respectively. Table 6.1 summarizes the ICEWS data.

To prepare the ICEWS data for modeling, the articles are broken up by sentence. The context window of word2vec resets with each sentence. Therefore, regardless of selected window size, words are only trained on their context within a single sentence.<sup>1</sup> All words are converted to uppercase, sentences with multiple independent phrases are split at the semicolon, punctuation is removed, and letters not found in the standard English alphabet are converted (where possible) to their English

---

<sup>1</sup> In fact, the 2013 corpus is divided by sentence and the 2014 corpus is divided by article. This was done for model evaluation purposes and made no noticeable difference to the results. Due to time and resource constraints, the 2013 and 2014 corpora were not rebuilt in a standard format before training the final models presented here.

alphabet equivalent. A simple phrase parser was used to preprocess the data and concatenate words into multi-word phrases of up to four words in length. For training the selected window size is 10 and the dimensionality of projections (in effect, the size of the hidden layer in the neural network) is set at 300 to match that of the *GoogleNews* model. Words and phrases that do not appear at least 3 times in the entire corpus are omitted.

Table 6.2: Most similar words

Term	Most similar terms (decreasing order)
BANANA	TOMATO, COCONUT, PAPAYA, ...
FOOTBALL	SOCCER, BASKETBALL, RUGBY, ...
BARACK OBAMA	OBAMA, PRESIDENT BARACK OBAMA, WHITE HOUSE, ...
XI JINPING	XI, HU JINTAO, COMRADE XI JINPING, ...
WASHINGTON DC	CHICAGO, NEW YORK, WASHINGTON, ...
BELJING	CHINA, GUANGZHOU, HONG KONG, ...

Table 6.3: Algebra on word vectors

Equation	Most similar terms (descending order)
OBAMA + CHINA - USA <sup>1</sup>	XI, PREMIER LI, BEIJING, ...
OBAMA + NEW YORK - USA	MR DE BLASIO, MAYOR BILL DE BLASIO, MR CUOMO, ...
CHINESE + RIVER	YANGTZE RIVER, YANGTZE, YELLOW RIVER, ...
AMERICAN + RIVER	RIVERTHE <sup>2</sup> , ROCKY MOUNTAINS, POTOMAC RIVER, ...
DEMOCRACY + CHINA - USA	CONSTITUTIONALISM, HARMONIOUS MODERNIZED SOCIALIST, ONE PARTY RULE, ...
F22 + CHINA - USA	J20 FIGHTER, TURBOSHIFT ENGINE, J16, ...

1. "USA" is written here for space considerations while "UNITED STATES" was used in practice.
2. There were occasional parsing errors.

## 6.3 Results

The results are presented in three parts. First, properties of the word2vec models are explored. Next, the unsupervised approach to actor classification is described in more detail and the results are described. Finally, two supervised approaches are described and the results compared to those of the unsupervised approach.

### 6.3.1 Word2Vec Models

*Word2vec* exhibits interesting properties that result from its skipgram structure. By learning from context, word2vec models tend to place synonyms close to one another in the resulting vector space. By calculating the cosine similarity between a given word and the rest of the words in the learned model, one generally finds that those words near to the chosen word in space are also near to the chosen word in meaning. The examples in Table 6.2 come from *ICEWS730*. Additionally, algebra can be performed on the vectors to produce the equivalent of analogies. Examples of this are shown in Table 6.3. The canonical example in the word2vec literature is king:man::queen:woman.<sup>2</sup> *ICEWS730* learned, for instance, that F22 is to the USA as the J20 is to China.<sup>3</sup> Democracy is to the USA as constitutionalism,<sup>4</sup> harmonious modernized socialist, and one party rule are to China (in that order).

In order to categorize individuals by country given their word vectors, it is first instructive to visualize how well the model separates groups by location. Multi-dimensional scaling is used to project the 300 dimension word vectors into a two dimensional space while best preserving between-unit distance. First, the actors and

---

<sup>2</sup> Given the problem king:man::\_\_\_\_\_:woman, *ICEWS730* fills in the blank with “princess,” “queen,” and “majesty,” in that order.

<sup>3</sup> Interestingly, F35 is to USA as \_\_\_\_\_ is to China returns J31, China’s next generation fighter aircraft that has been rumored to be based in part on stolen plans for the F35.

<sup>4</sup> “Constitutionalism” appeared frequently in news reports with respect to Chinese governance during the Fourth Plenum of the 18th Party Congress in 2014 during which President Xi expounded the importance of “rule of law.”

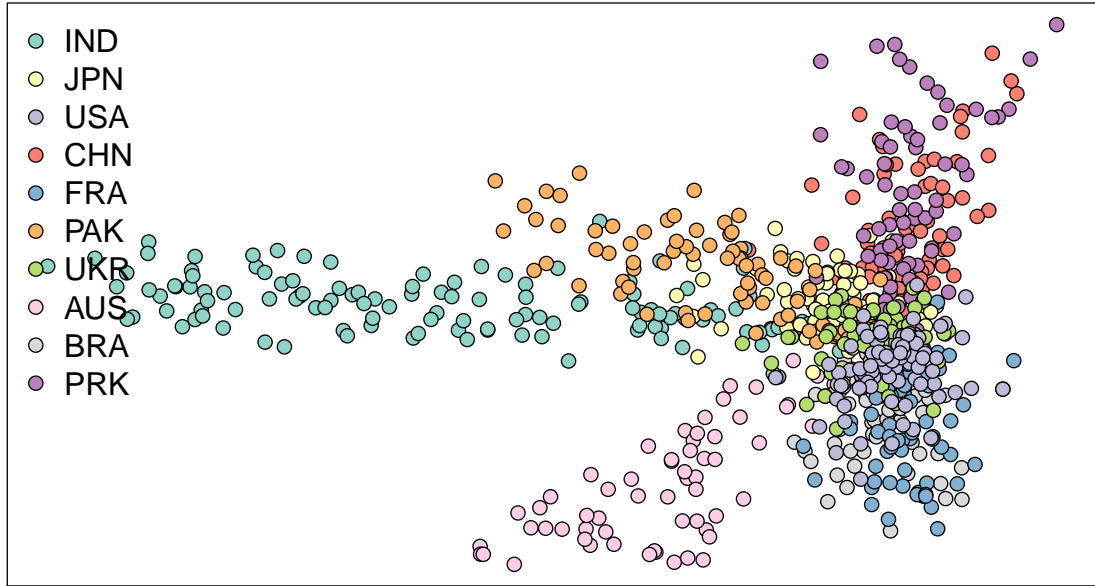


FIGURE 6.1: Clustering of actors by country after multidimensional scaling

locations from the PHOENIX dictionary are matched with actors and locations in *ICEWS730*. Then, the ten countries with the most actors are taken to be plotted. Figure 6.1 plots all unique actors from the top ten countries colored by their associated country. The figure shows clear clustering by country but only marginal separation between countries. Figure 6.2 plots unique locations (alternate country spellings, city names, state names, etc...) and colors by country. Again, there is clear clustering of locations by country, but also overlap between the country clusters.

Table 6.4: Model performance in unsupervised task

Model	Actors	Locations	Countries	Random 1	Random 2	k=1	k=3	k=5	k=7
<i>ICEWS90</i>	2884	2271	182	0.01	0.01	0.25	0.27	0.29	0.30
<i>ICEWS183</i>	3890	2447	182	0.00	0.01	0.29	0.31	0.34	0.35
<i>ICEWS365</i>	4764	2630	182	0.01	0.01	0.32	0.34	0.37	0.38
<i>ICEWS730</i>	5096	2756	183	0.01	0.01	0.37	0.39	0.42	0.43
<i>GoogleNews</i>	4932	2616	183	0.01	0.01	0.58	0.62	0.64	0.64

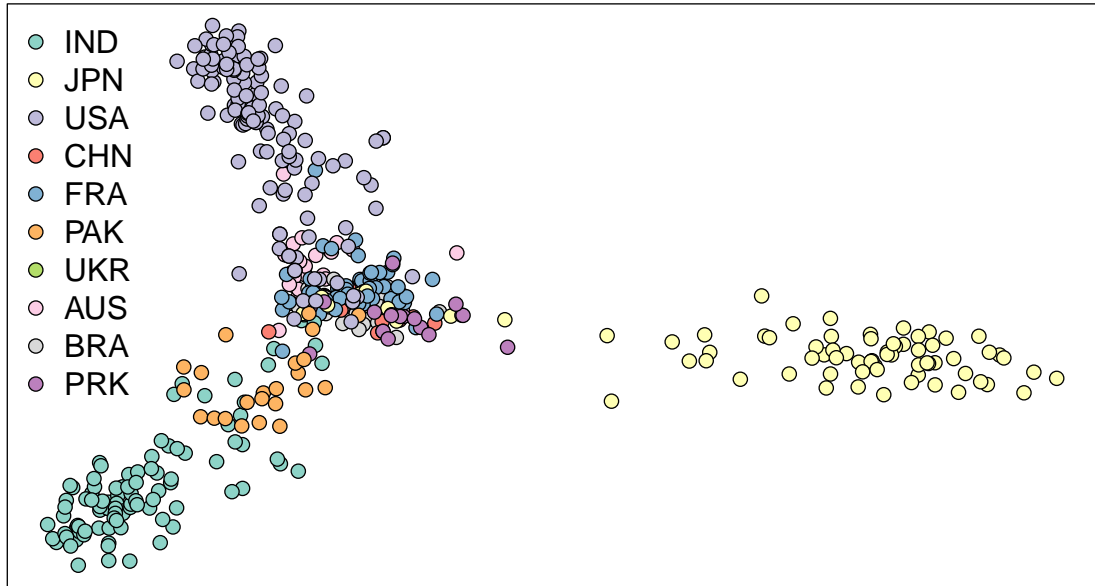


FIGURE 6.2: Clustering of countries after multidimensional scaling

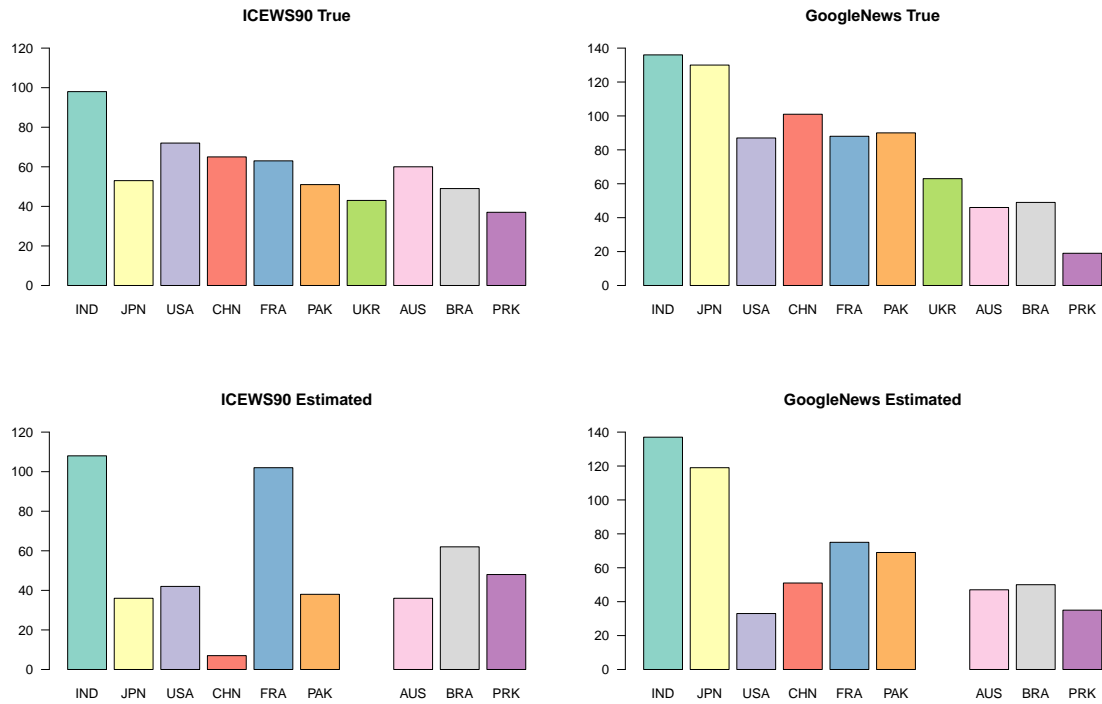


FIGURE 6.3: True and estimated counts of actor's countries

### 6.3.2 Unsupervised classification

In the first classification task, we are given a set of location names and a set of actors to classify by country, but no training data (actors already labeled by country affiliation). The objective is to simulate the process of ingesting news and producing actor-country dictionaries for event coding without human intervention. Actor and location names are assumed to have been extracted using existing named entity recognition solutions though, in reality, actor and location names are chosen in accordance with the PHOENIX country-actor dictionary for validation purposes.

In the unsupervised approach, a cosine similarity matrix is constructed between the vectors associated with the actors' names and all of the vectors associated with location names that are identified in both the PHOENIX dictionary and in the trained models. Cosine similarity,  $(\vec{X} \cdot \vec{Y})/(\|\vec{X}\|\|\vec{Y}\|)$ , is a measure of the angle between high-dimensional vectors. Each actor is then assigned to the country associated with the location to which he or she is most similar. The process is tested for all five available word2vec models and the results are shown in Table 6.4. Additionally, k-nearest neighbors voting for odd values of k between 3 and 7 is also tested.<sup>5</sup>

Table 6.4 begins with a summary of each model including the number of actors from PHOENIX that are identified in the model's vocabulary, the number of locations from PHOENIX that are identified in the model's vocabulary, and the number of countries that are represented by those locations. The columns labeled *Random 1* and *Random 2* indicate the proportion of correctly-classified actors under uniform random assignment and random assignment from the true distribution of actor countries, respectively. Under both cases of random assignment, we expect to correctly classify about 1% of actors. Finally, the last four columns use the k-nearest neighbors classifier to assign country labels to actors based on cosine similarity.

---

<sup>5</sup> K-nearest neighbors where "nearness" is determined by cosine similarity.

All five models substantially outperform random assignment in classifying actors. The worst model correctly classifies 25% of the data. Models trained on larger corpora achieve better performance than those trained on smaller corpora and k-nearest neighbors voting improves scores as k increases. *GoogleNews* correctly classifies between 58% and 64% of all actors in the PHOENIX country-actor dictionary. The next highest scores are achieved by *ICEWS730*, correctly classifying between 37% and 43% of actors. Interestingly, more PHOENIX actors and locations are identified in the *ICEWS730* model’s vocabulary than in *GoogleNews*’s. This is likely due in large part to the ability of the researcher to match data preprocessing procedures between the dictionary and the *ICEWS* models. The discrepancy in performance between *GoogleNews* and *ICEWS730* could be the result of corpus size or model training parameters and will require more investigation.

Figure 6.3 should alleviate concerns about a single country dominating the results. On the top row of the figure, the true distribution of actors among the top ten most common countries is plotted for *ICEWS90* and *GoogleNews*. On the bottom row is the distribution of estimated labels. No single country dominates either the true distributions or the estimates.

K-nearest neighbors does not inherently provide measures of uncertainty in classifications. In order to better understand model fit, another model is estimated using a weighted k-nearest neighbors where  $k=10$  and voting is weighted by cosine similarity. Therefore, every actor will be assigned at least one and at most ten country labels. Each label is accompanied by a weighted score. These scores are l1-normalized to produce “probabilities.” All other countries, those that are not in the top ten votes according to k-nearest neighbors, are assigned a probability of 0. The results are then treated as a series of one-versus-all binary classifiers and a ROC curve is estimated for each. Figure 6.4 plots the ROC curves for the top ten countries. The model used was *ICEWS730*. The average AUC for the top ten is 0.91 while the average AUC

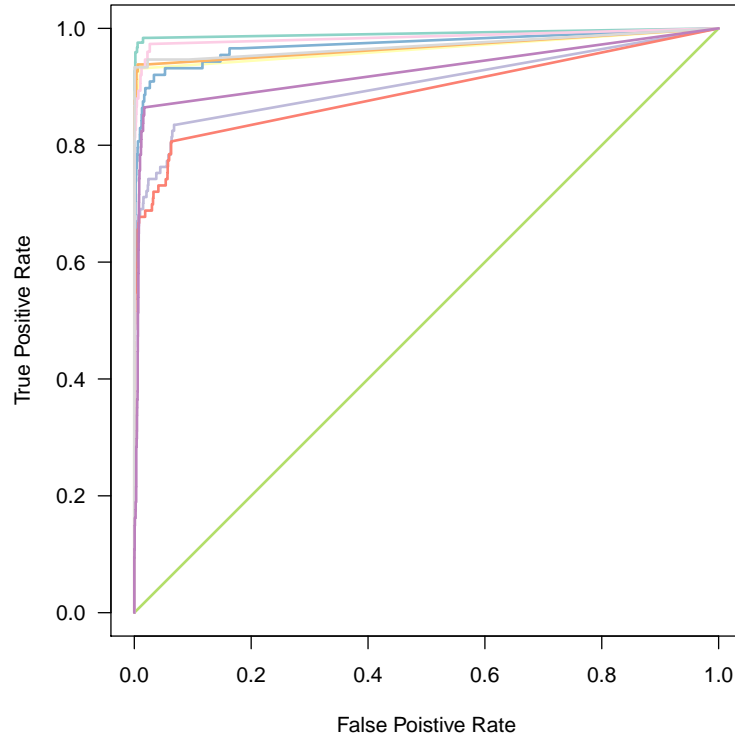


FIGURE 6.4: Multiclass ROC plot for top ten countries

for all 183 countries is 0.77. The weighted k-nearest neighbors model itself performs comparably to the unweighted k-nearest neighbors models and correctly classifies 45% of actors.

### 6.3.3 Supervised classification

An alternate approach to actor classification would compare the similarity of actors to one another rather than to the location vectors. This requires that some actors be labeled with the appropriate countries a priori and is therefore a supervised classification problem. For updating event coding dictionaries, this is not an unreasonable assumption. For example, maintaining the PHOENIX dictionaries could be made more manageable if information contained in previous versions could be used to train the nearest neighbors model for near-real time updating.

Once again, a series of k-nearest neighbors models is estimated using the five

word2vec models. However, unlike before, location vectors are not included in the model and instead labeled training data consisting of pre-classified actors is included. The amount of training data is varied such that each model is trained using a randomly-selected training set of 10% through 90% of the data in steps of 10%. Furthermore, uncertainty is estimated by running each classifier five times and calculating the standard deviation of the classification scores. The results for the *ICEWS* models can be seen in Figure 6.5. The black line indicates the average percentage of the test set that is correctly classified for a given training set size. The two gray shaded regions indicate 95% and 99% confidence intervals estimated by repeatedly sampling training and test sets and calculating the standard deviation of percent correct classification. Results for  $k = 1, 3,$  and  $5$  were estimated;  $k = 5$  in the results shown here.

Once again, model performance improves with corpus size. Additionally, performance improves as the size of the training set increases. In fact, as the size of the training set approaches nine times the size of the test set, the supervised models outperform their unsupervised counterparts by about ten percentage points. *ICEWS90* correctly classifies just over 20% of the test cases when given a small training set. On the other end of the spectrum, *ICEWS730* correctly classifies over 60% of the test cases when given a large training set.

The choice of supervised or unsupervised classification largely depends on the problem faced by the researcher. In the case of producing dictionaries for event data coding tasks, the availability of existing dictionaries in need of updating may provide a cheap and effective training set for the supervised classification task. On the other hand, I have shown that the unsupervised classification task for producing dictionaries (nearly) from scratch may be a viable alternative to the supervised approach.<sup>6</sup>

---

<sup>6</sup> I also attempted a supervised classification task that included both known location vectors

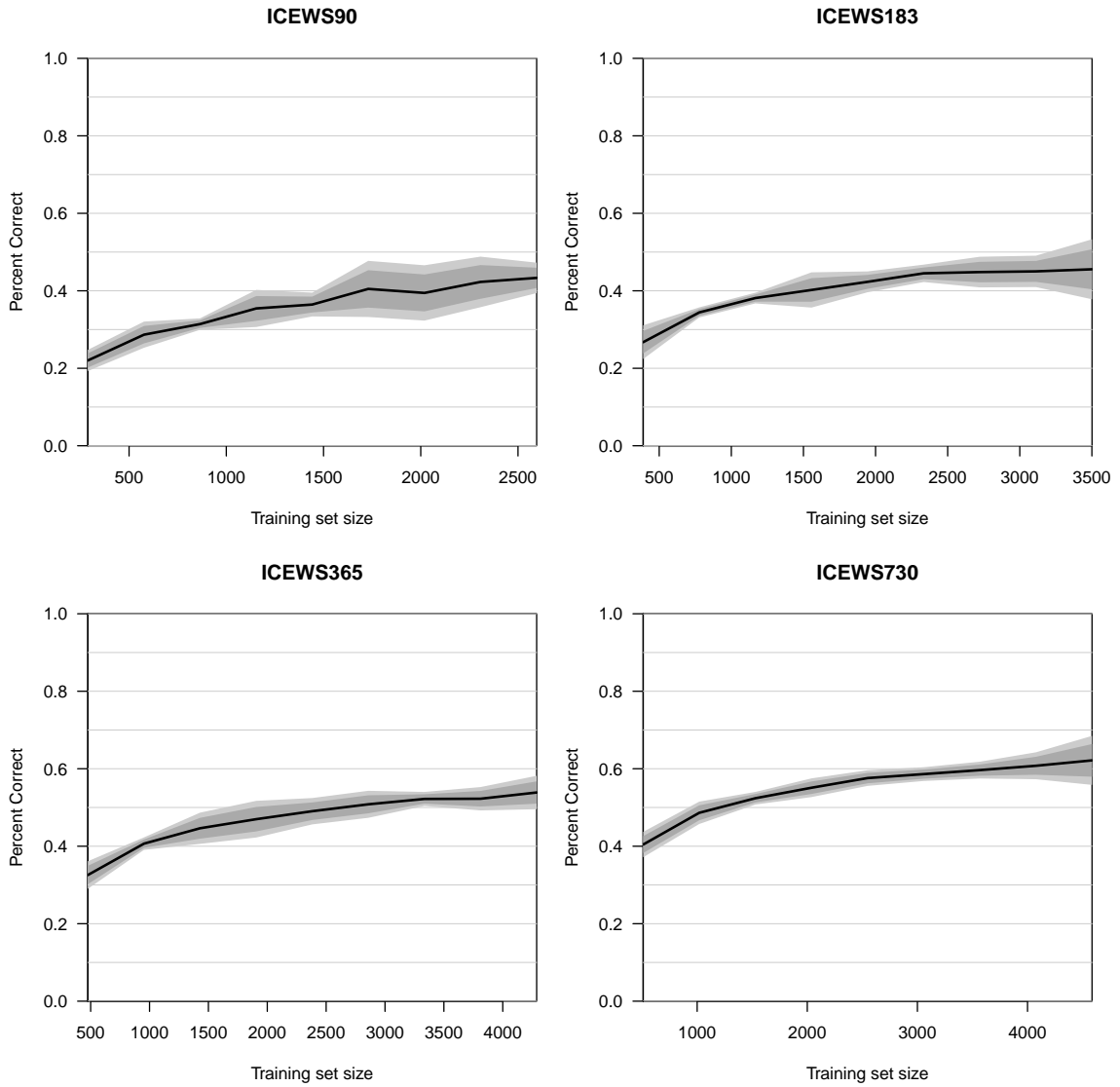


FIGURE 6.5: Percent correct classification, supervised approach

## 6.4 Conclusion

Event data promises political scientists a rapidly updated and detailed view of world events. While event data is machine-coded, the dictionaries required for event data are still produced manually and require frequent updating. This paper presents a potential technique for the automating of the dictionary creation process. In particular, and labeled actor vectors in the training set. This resulted in nearly identical performance to the actor-only supervised task and so is not reported here.

the same text data that would eventually be coded as event data is used to learn the country associations of the political actors relevant to that data. A future iteration of this research could find its way into a dictionary creation pipeline that begins with named entity extraction and results in entirely machine-learned and machine-coded event data.

While this paper focused on one particular PHOENIX dictionary, the country-actor dictionary, there are also dictionaries for international actors and non-state actors. Future research will need to extend this work for other dictionary types and, ultimately, for all dictionary types. This will require multiclass labeling of actors that may fall into more than one category.

Additionally, more refinement will be needed before the models presented here are ready for deployment in a real-world event data coding framework. While these models perform extraordinarily well when compared to random assignment, they still likely fall short of the accuracy of dedicated human coders. Improvements to data preprocessing and to the post-word2vec label assignment stage will likely lead to more accurate machine-learned actor classifications.

## Conclusion

In this chapter, I briefly review the research described thus far and identify relevant work that remains to be done. First, the novel contributions of automated dictionary generation and CYLICON are summarized. Then, open questions in event coding techniques are identified. An alternative and promising approach to cybersecurity data for social scientists is also described.

### 7.1 CYLICON and Event Coding

The work here outlines a process by which researchers can quickly and easily produce novel event datasets in new and interesting domains. With minimal input from a researcher about the topic of interest, the method produces dictionaries of pre-categorized words and phrases for use with existing event-coding software. While the focus of this work has been integration with PETRARCH I, adapting the techniques to work with additional event coders should be, in large part, a matter of adjusting the heuristics and formatting used for phrase post-processing.

CYLICON provides the first dataset for social scientists on cybersecurity events consistent with existing event datasets. The work presented here offers not only new

data, but a novel set of dictionaries that describe the domain of cybersecurity. It is my hope that CYLICON will allow researchers to both better understand interaction in cyberspace as well as incorporate cooperative and conflictual events that occur in this domain into broader analyses on the interaction of politically-motivated or relevant actors.

Collection of data for CYLICON continues. While the data presented here end in late 2015, a webscraper has continued to collect data and updated event data will be made. Because dictionary updates are now inexpensive, the question of version control becomes pertinent. New event data can be produced daily and so can new dictionaries. However, this will result in the dictionaries themselves evolving over time and therefore the events themselves will be coded inconsistently from day-to-day. One option is to accept this. In fact, vocabulary does evolve with time and ambiguous phrases like “hackers attacked” may likely refer to DDOS during one time period and infiltration during another. On the other hand, researchers may prefer that their verb phrases remain consistently coded throughout the entirety of a dataset. For this reason, I propose that CYLICON be updated on two schedules: a daily event data cycle and an annual dictionary schedule. In this way, CYLICON is versioned by year with each year representing a new set of dictionaries derived from a new word2vec model. Within each version, the dictionaries will remain consistent.

## 7.2 The Future of Automated Event Coding

While the automated dictionary generation process takes a substantial step in the direction of a fully-automated event data generation solution, work remains to be done in this area. Event coding software itself, like PETRARCH, remains primarily heuristic-based and imperfect. The stacking of multiple analytic techniques for sentence parsing, phrase-extraction, and named entity recognition, among others, compounds errors that lead to sub-optimal event coding. Future efforts should leverage

advances in machine learning to minimize the application of heuristics and stacking of text pre- and post-processing steps.

Smaller yet important improvements also remain to be made. Techniques for better identifying the direction of action and for determining the number of relevant actors would greatly extend the usability of event data. Allowing monadic or triadic interactions in event data is a promising area of research. Finally, solutions that would allow both supervised and unsupervised event data generation should be explored. Is it possible to generate event data given domain-relevant texts but no a priori knowledge of the ontology, words, or phrases that constitute that domain? Can the performance of a system built in this way be improved by supplying a training set of hand-coded event data relevant to the domain? Alternatively, can event data in domain A be used to train an algorithm that produces event data in domain B? Work that progresses in these directions will ultimately help us to produce better and fully replicable event datasets that are tailored to specific domains and research questions.

### 7.3 Alternative Approaches to Cyber Data in Political Science

As cyberspace has become a prominent venue for major crime and international conflict, incorporating events that occur in cyberspace into existing event datasets will become ever more important. However, there are other sources for data on cyberconflict that could prove fruitful for social scientists that do not lend themselves to the event format. For instance, cybersecurity firms regularly put out reports on APT activity that include the countries and industries targeted by those campaigns.<sup>1</sup> Lacking identification of specific victims, these reports rarely make the news feeds that supply event data and tend to be too technical for use with existing event coding

---

<sup>1</sup> For an example of these reports, see the Kaspersky Targeted Cyberattack Logbook at <https://apt.securelist.com>. Also FireEye Cyber Threat Intelligence Reports at <https://www.fireeye.com/current-threats/threat-intelligence-reports.html>.

solutions. But, a manual effort to code these APT campaigns from investigative reports could produce a valuable dataset similar to the ICB dataset: *campaign name, suspected source, target country, target industry, 0days employed, discovery date, earliest known instance, latest known instance, suspected objective*. This dataset would capture much of the state-sponsored activity that is overlooked by newswire-style reports.

Coding these data would come with challenges. Reports often catch APTs *in media res* but do not provide subsequent follow-ups. Therefore, determining the termination date of a campaign may sometimes be impossible. APTs are given nicknames by each firm that discovers them and so deduplication of APTs across multiple reports from multiple security firms would provide an additional challenge. Finally, these reports come in a variety of shapes and sizes with varying degrees of specificity and so finding a common subset of indicators and covariates like those suggested above will prove challenging.

While some large-scale political phenomenon like Militarized Interstate Disputes are, arguably, an aggregate of many smaller events that would appear in traditional event datasets, advanced cyber conflict may not appear as an aggregate of smaller cybersecurity events.<sup>2</sup> APTs tend to practice extreme secrecy and adapt rapidly after discovery. The first “event” associated with many APTs is discovery, at which point they have likely already infected many targets, most of whom will forever remain anonymous. Researchers should carefully consider both the underlying data generating process and the reporting process of cyber conflict events and campaigns as, for the reasons described above, it is unlikely that a single dataset will capture the full spectrum of political activity online.

---

<sup>2</sup> The goal of the MADCOW project, funded by the National Science Foundation, is to produce political indicators from open information sources. Philip Schrodt writes of the program “We will use event data, textual frames, and crowd-sourced alerts to provide rapid updating on militarized disputes as they occur” (Schrodt et al., 2016).

## 7.4 Final Thoughts

Cybersecurity is a huge concern for governments and industry alike. One estimate puts the value of the cybersecurity industry at \$120 billion in 2017 (Ross, 2016). Morgan (2015), a contributor to Forbes, cites an industry valuation of \$170 billion by 2020. Intel Security (2014) estimated the annual global cost of cybercrime to be over \$400 billion. Meanwhile, nation states continue to escalate conflict in cyberspace, from the theft of massive amounts of data from the U.S. Office of Personnel Management by China to allegations of Russian interference in the U.S. presidential election (Nakashima, 2016).

This domain promises exciting new opportunities for social scientists seeking to expand our understanding of how people, organizations, and governments interact with one another. CYLICON represents an effort to quantify interesting events in cyberspace; to record a new political space as it has evolved from inception to its current state. As the cybersecurity landscape changes in the coming years, CYLICON will remain up-to-date due to its ability to adapt to a changing and growing vocabulary and set of relevant actors. It is my hope that CYLICON, and the techniques developed in its creation, will help researchers to advance our understanding of cyber conflict and to generate better event data across domains.

# Appendix A

Dictionary samples

Table A.1: Sample of the verb dictionary

```

--- FIXES [---] ---
FIXES [PATCHED0022]
- ADOBE ** [PATCHED0755]
- + ** BY $ [PATCHED0001]
- ** INCLUDED [PATCHED1917]
- ** CRITICAL [PATCHED0410]

--- FIXED [---] ---
FIXED [PATCHED0024]
- VULNERABILITIES ** IN [PATCHED0331]
- HAVE BEEN ** [PATCHED1119]
- HAD BEEN ** [PATCHED1853]
- FLAWS ** [PATCHED0659]
- BUGS ** IN [PATCHED1260]
- ALSO BEEN ** [VULNERABILITY1375]
- + ** BY $ [PATCHED0002]

--- UPDATED [---] ---
UPDATED [PATCHED0025]
- + ** BY $ [PATCHED0003]
- ** FLASH PLAYER [PATCHED0857]
- ** AT ALL TIMES [WARNED2825]

--- PATCHED [---] ---
PATCHED [PATCHED0026]
- WILL BE ** [PATCHED1462]
- WAS ** [PATCHED0715]
- VULNERABILITY ** [PATCHED0833]
- RECENTLY ** VULNERABILITY [VULNERABILITY1417]
- RECENTLY ** [PATCHED0518]
- ORACLE ** [PATCHED0588]
- HAS BEEN ** [PATCHED0876]
- HAS ALREADY BEEN ** [PATCHED0701]
- HAS ** [PATCHED0402]
- HAD BEEN ** [PATCHED1441]
- HAD ALREADY ** [PATCHED1830]
- FULLY ** &SOFTWARE [PATCHED1001]
- CRITICAL VULNERABILITIES ** IN [PATCHED1831]
- BEEN ** [PATCHED0735]
- APPLE ** [PATCHED0651]
- ALSO BEEN ** [VULNERABILITY0985]
- + ** BY $ [PATCHED0004]
- &SOFTWARE ** [PATCHED0418]
- ** UP [PATCHED0023]
- ** &SOFTWARE [PATCHED1651]

--- CONFISCATED [---] ---
CONFISCATED [ARRESTED0027]
- HAVE BEEN ** [ARRESTED0617]

```

Table A.2: Sample of the agents dictionary

KASPERSKY [~ANT]  
 F-SECURE [~ANT]  
 AVAST [~ANT]  
 SECURITY\_RESEARCHER [~RES]  
 RESEARCHERS [~RES]  
 RESEARCHER [~RES]  
 LULZSEC [~HAC]  
 ANONYMOUS [~HAC]  
 HACKERS [~HAC]  
 HACKER [~HAC]  
 HACKTIVISTS [~HAC]  
 TEAMPOISON [~HAC]  
 GROUP [~HAC]  
 UGNAZI [~HAC]  
 SECURITY\_EXPERT [~RES]  
 ANONYMOUS\_HACKERS [~HAC]  
 SUNBELT [~RES]  
 FORMER\_CONTRACTOR [~WHI]  
 EXPERT [~RES]  
 SENIOR\_ANTIVIRUS [~RES]  
 MEMBERS [~HAC]  
 VULNERABILITY\_LAB [~RES]  
 EXPERTS [~RES]  
 S3RVEREXE [~HAC]  
 SECURITY\_RESEARCHERS [~RES]  
 SECURITY\_FIRM [~RES]  
 MCAFEE\_AVERT\_LABS [~RES]  
 VULNERABILITY\_LAB [~RES]  
 SECURITY\_SOLUTIONS\_PROVIDER [~ANT]  
 FORMER\_NSA\_CONTRACTOR\_WHO [~WHI]  
 OPKILLINGBAY [~HAC]  
 GCSB [~WHI]  
 SECURITY\_RESEARCHER [~RES]  
 ASSANGE [~WHI]  
 NSA\_DOCUMENTS [~WHI]  
 PANDA\_SECURITY [~ANT]  
 OPERATION\_FREE\_PALESTINE [~HAC]  
 REVELATIONS [~WHI]  
 AUTHORITIES [~HAC]  
 HACK [~HAC]  
 FAWKES\_SECURITY [~HAC]  
 COMMTOUCH [~ANT]  
 KASPERSKY [~RES]  
 WEBSITE [~HAC]  
 FIRM\_F-SECURE [~ANT]  
 GREY\_HAT [~HAC]  
 WBC [~HAC]  
 MANNING [~WHI]  
 FORMER\_NSA\_CONTRACTOR [~WHI]  
 SECURITY\_FIRM [~ANT]

# Appendix B

## Verb dictionary performance

ARRESTED	3	4	89
DDOS	2	8	8
DEFACED	2	4	15
INFECTED	4	3	15
INFILTRATED	18	2	111
LEAKED	3	0	10
PATCHED	1	1	3
VULNERABILITY	2	0	1
TOTAL	35	22	252

(0) INCORRECT    (1) PARTIALLY CORRECT    (2) CORRECT

FIGURE B.1: Softpedia corpus action accuracy by category. Numerical values represent the frequency of each category of event by that event's hand-coded accuracy score. Colors are scaled independently by category. Only events coded from Softpedia stories are represented.

ARRESTED	13	2	148
DDOS	3	3	1
DEFACED	0	0	1
INFECTED	8	4	27
INFILTRATED	37	13	82
LEAKED	5	1	22
PATCHED	1	1	3
VULNERABILITY	9	1	0
TOTAL	76	25	284

(0) INCORRECT    (1) PARTIALLY CORRECT    (2) CORRECT

FIGURE B.2: Technical corpus action accuracy by category. Numerical values represent the frequency of each category of event by that event's hand-coded accuracy score. Colors are scaled independently by category. Only events coded from the Technical corpus stories are represented.

# Appendix C

## CYLICON data

Grey highlighted items are from the Technical Corpus. Other items are from Softpedia *Security News*. Event codes are followed by four-digit numerical codes to facilitate evaluation of the dictionaries.

Date	Action	Source Actor	Target Actor
20000101	LEAKED0030	UKRHAC	UKR
20000101	INFILTRATED0044	XXXHAC	USAEDU
20000101	ARRESTED0695	FRA	ROU
20000101	LEAKED0578	COLHAC	COLCOPHAC
20000101	DEFACED0106	SYRMIL	MNCUSAMED
20000101	INFILTRATED0735	USAOPP	CHN
20000101	INFILTRATED0392	USASPY	CHNHACBUS
20000101	DEFACED0505	MNCUSAMED	PAKHAC
20000101	DEFACED0586	XXXHAC	TURGOV
20000101	INFILTRATED0190	INDHAC	XXXRESANT
20000101	INFILTRATED0195	RUSGOV	USAGOV
20000101	INFILTRATED0043	USACOP	USAHACBUS
20000101	ARRESTED0021	BGRRES	USA
20000101	DDOS0431	USAMED	CHN
20000101	DEFACED0295	LKAGOV	XXXHAC
20000101	ARRESTED0218	POLEDU	IRQGOVHAC
20000101	DDOS0430	CHNHAC	RUSGOV
20000101	INFILTRATED0043	XXXHAC	USAHAC

20000101	ARRESTED0402	JORELI	USAGOV
20000101	INFECTED1020	GBRRESMED	GBRHACLEG
20000101	INFILTRATED0753	XXXHAC	USA
20000101	ARRESTED0278	PRKGOV	MNCJPNHAC
20000101	ARRESTED0175	SGP	IRQGOVMED
20000101	INFILTRATED0195	JORHAC	EGYGOVMED
20000101	ARRESTED0481	MNCGBRMED	AUS
20050217	INFILTRATED0037	IRNHAC	ITA
20060303	INFILTRATED0034	ARGHAC	VENGOVMED
20060706	INFILTRATED0195	SYRMIL	USAHAC
20070312	ARRESTED0006	CHNGOV	USA
20070319	ARRESTED0396	CHNCOP	CHNGOV
20070405	INFILTRATED0736	ROURES	FRAHACRES
20070906	ARRESTED0218	LVA	ROU
20071008	INFILTRATED0044	USA	USAMED
20071126	ARRESTED0652	IRLEDU	XXXHACLEG
20071129	INFILTRATED0044	USA	USABUS
20080514	INFILTRATED0226	SYRGOVHAC	TURHAC
20080715	ARRESTED0177	XXXHAC	GBR
20081004	ARRESTED0218	CANCVL	USA
20081031	ARRESTED0337	USA	NGA
20081110	ARRESTED0534	ARMHAC	USAJUD
20081122	INFILTRATED0564	MNCUSAMED	USASPY
20081201	INFILTRATED0564	XXXRESHAC	USALEG
20081223	INFILTRATED0044	SYRHAC	SYR
20090123	ARRESTED0162	ARGCOP	ARG
20090210	ARRESTED0006	USACOP	USA
20090214	ARRESTED0022	XXXRES	UKR
20090216	ARRESTED0727	SYRMIL	SYR
20090225	LEAKED0030	XXXHAC	USAGOV
20090320	ARRESTED0789	EST	USA
20090323	DDOS0350	SCGSRB	ALBMED
20090325	INFILTRATED0736	MEAREB	USA
20090331	INFILTRATED0011	AUTHAC	AUTLEG
20090525	DDOS0281	PSEREBHMSUAF	USAHAC
20090717	ARRESTED0373	USACOP	MNCUSALAB
20090728	ARRESTED0171	USAGOV	LCAGOV
20090824	INFILTRATED0043	XXXHAC	TURMEDGOV
20090925	ARRESTED0006	USACOP	BOLHAC
20091103	INFILTRATED0225	KMHAC	KMHACGOV
20091119	DEFACED0480	ROUHAC	XXXRES

20091126	INFILTRATED0010	PAKHAC	INDHAC
20091210	INFECTED1020	INDHAC	PAKMED
20100216	INFILTRATED0421	USAEDU	USA
20100304	INFILTRATED0286	PAKHAC	INDGOVMED
20100323	INFECTED1036	USAGOV	NGOUSAHRI
20100628	INFILTRATED0010	BGDLEGHAC	USA
20100701	INFILTRATED0043	TURHAC	INDMED
20100720	INFILTRATED0415	FRA	FRACVL
20100806	INFILTRATED0286	INDHAC	PAKHACMIL
20100817	ARRESTED0144	EST	USA
20100906	LEAKED0883	USA	BHS
20100909	INFILTRATED0226	SYRMIL	QAT
20100915	INFILTRATED0043	XXXHAC	YEMHACGOV
20100920	ARRESTED0170	USAGOV	ROUCVL
20100920	INFILTRATED0226	IRN	MNCUSA
20100927	INFILTRATED0043	XXXHAC	ISRMEDBUS
20100930	INFECTED0903	FINANT	LTU
20100930	ARRESTED0006	AUTCOP	COG
20101002	DDOS0430	CHNHAC	USA
20101019	ARRESTED0156	GBRCOP	JPN
20101025	INFECTED0945	ISR	IRNHAC
20101118	INFILTRATED0226	IND	CHN
20101129	ARRESTED0302	CHNGOV	CHN
20101210	ARRESTED0021	USAGOV	USA
20101214	LEAKED0879	MEAREB	SYRHAC
20101223	ARRESTED0784	USA	USAJUD
20110103	INFILTRATED0043	AFGHACMIL	SYRHAC
20110117	ARRESTED0373	USACOP	USA
20110124	INFECTED0891	CHNHAC	XXXRES
20110208	INFILTRATED0747	XXXHAC	VENMIL
20110216	ARRESTED0021	NGA	USA
20110219	INFILTRATED0011	INDGOV	PAKHACMIL
20110224	ARRESTED0140	POL	GBR
20110308	INFILTRATED0286	IDNHAC	USAGOVAGRMED
20110405	DDOS0431	GBRSPYHACRES	IMGMUSALQMED
20110415	ARRESTED0159	ROUCOPGOV	ROU
20110426	DEFACED0287	INDHAC	MNCINDMEDHAC
20110427	DEFACED0505	PAKHACRES	MNCUSAMED
20110525	ARRESTED0251	USAHAC	USACOP
20110603	INFILTRATED0043	AFGMIL	PAKGOVMED
20110604	INFILTRATED0043	AZEHAC	ARMED

20110613	INFILTRATED0035	IGOEUREEC	ISRMILGOVHAC
20110620	ARRESTED0006	XXXHAC	XXXHACLEG
20110623	INFECTED0903	USACOP	USAHACRES
20110624	INFILTRATED0736	XXXRES	CHNHAC
20110629	INFILTRATED0368	PAKMIL	IGOEUREECMEDBUS
20110630	INFILTRATED0044	XXXHAC	TURHAC
20110709	INFILTRATED0365	XXXRES	RUSCOP
20110713	ARRESTED0156	GBRCOP	GBRHACLEG
20110719	ARRESTED0171	USA	ISRHAC
20110719	LEAKED0101	XXXHAC	USALABGOV
20110720	LEAKED0101	XXXHAC	TURRESHACMIL
20110726	VULNERABILITY0730	XXXANTHACMED	XXXRES
20110808	INFILTRATED0706	PHLHAC	CHNGOVHAC
20110820	INFILTRATED0011	ISRMILHACRES	IRNGOV
20110826	ARRESTED0170	USA	XXXRES
20110826	ARRESTED0200	MED	AUS
20110923	INFILTRATED0736	XXXRES	SYRMILHAC
20111005	INFILTRATED0046	USASPY	PRK
20111105	INFILTRATED0011	USAHACCOP	PAKHAC
20111205	DEFACED0505	IDNHAC	ISRHAC
20120106	DEFACED0243	SRBHAC	IGOUNOHAC
20120107	ARRESTED0288	MDA	USA
20120130	INFILTRATED0043	IND	PHLMEDEDU
20120130	INFILTRATED0736	USABUS	USA
20120208	ARRESTED0633	USA	FRAGOV
20120221	ARRESTED0396	XXXRESCOP	BEL
20120228	INFILTRATED0735	RUSSPY	DEURESANT
20120304	PATCHED0252	XXXRES	USAGOVRESHAC
20120328	INFILTRATED0556	INDHACGOV	INDCOP
20120329	INFILTRATED0736	USAHACMED	ROUHACMED
20120424	INFILTRATED0736	AUSGOV	MNCGBRMEDRES
20120514	ARRESTED0140	CZERES	GRC
20120516	INFECTED0990	MED	TWN
20120607	DDOS0253	NZLGOV	CHN
20120611	PATCHED0854	XXXRESANT	DEUHACRES
20120627	LEAKED0744	USAGOVWHI	USASPY
20120706	ARRESTED0140	XXXWHIRES	NZL
20120719	LEAKED0101	USAOPP	MED
20120806	ARRESTED0798	GBRCOPWHI	GBRSPY
20120829	INFILTRATED0736	XXXWHI	XXXWHIHAC
20120912	ARRESTED0695	ROUCOPGOV	ROU

20120916	INFECTED0892	USASPY	USASPYRES
20120917	INFECTED0892	USAOPP	USASPYRES
20120924	VULNERABILITY0628	USASPY	USA
20121002	ARRESTED0022	NLD	USABUS
20121012	INFILTRATED0045	CHNHAC	MEAREBMEDHAC
20121017	INFILTRATED0041	XXXRESHAC	XXXRES
20121027	ARRESTED0396	USACOP	MNCKORMED
20121031	INFILTRATED0260	MNCKORANT	MNCKOR
20121107	ARRESTED0022	CAN	CANHAC
20121109	ARRESTED0200	NZLGOV	NZL
20121113	ARRESTED0200	NZLGOV	NZL
20121120	INFILTRATED0044	CHNMIL	USA
20121128	ARRESTED0335	CAN	GBR
20121206	ARRESTED0335	CAN	GBR
20121210	ARRESTED0335	CAN	GBR
20121219	ARRESTED0428	CAN	GBR
20121229	ARRESTED0412	USA	USAJUD
20130103	ARRESTED0428	CAN	GBR
20130107	LEAKED0342	USAOPP	USAGOV
20130107	ARRESTED0162	RUS	LBR
20130107	ARRESTED0297	RUSGOV	RUS
20130111	INFILTRATED0044	USASPY	CHNBUS
20130118	INFILTRATED0195	INDHAC	PAKGOVMED
20130124	INFILTRATED0195	INDHAC	PAKGOVMED
20130124	DEFACED0505	PAKHAC	INDHACCOP
20130131	INFECTED0906	MNCUSA	MNCUSAHAC
20130201	INFILTRATED0046	RUSHAC	USA
20130202	INFILTRATED0046	RUSHAC	USA
20130202	INFECTED0970	MNCUSAMED	MNCUSACVL
20130204	ARRESTED0792	MNCUSAMED	IGOEUREEC
20130204	INFECTED0989	SYRGOV	XXXRES
20130204	INFECTED0989	SYRGOV	XXXRES
20130207	ARRESTED0412	XXXRES	USA
20130213	LEAKED0342	USAMILRES	IRN
20130215	PATCHED0699	MNCUSA	USAREB
20130215	ARRESTED0781	IGOCOPITP	PHL
20130223	ARRESTED0798	USAGOV	USACHR
20130223	LEAKED0101	MNCFINMED	MNCUSA
20130225	ARRESTED0146	USAGOV MIL	USAHAC
20130225	INFILTRATED0226	GBRSPY	DEUBUS
20130227	ARRESTED0784	MNCUSALAB	MNCUSA

20130303	LEAKED0864	MNCUSA	MNCUSAMED
20130304	LEAKED0612	MNCUSAHLH	MNCUSAHLHMED
20130306	ARRESTED0795	USACVLEDU	USA
20130315	ARRESTED0795	USACVLEDU	USA
20130318	INFILTRATED0226	USASPY	DEUGOV
20130321	ARRESTED0795	USACVLEDU	USAJUD
20130325	ARRESTED0334	MDV	USA
20130405	INFILTRATED0226	MNCUSAMEDHAC	MNCUSAMED
20130406	ARRESTED0218	RUSHAC	USAGOV
20130413	INFILTRATED0226	PAKHAC	INDMED
20130415	INFILTRATED0226	PAKHAC	INDMED
20130418	ARRESTED0795	USACVLEDU	USAJUD
20130422	ARRESTED0302	USA	CHN
20130423	INFILTRATED0564	CHNHAC	USA
20130424	INFILTRATED0226	MNCUSAMEDHAC	MNCUSAMED
20130424	ARRESTED0334	XXXRES	USA
20130429	INFECTED0892	XXXANT	DEU
20130502	LEAKED0605	USAOPP	XXXWHI
20130506	LEAKED0605	USAOPP	XXXWHI
20130507	ARRESTED0302	USA	CHN
20130513	ARRESTED0302	USA	CHN
20130605	ARRESTED0795	USACVLEDU	USA
20130610	LEAKED0100	USAOPP	USASPYWHI
20130615	LEAKED0100	USAOPP	USASPYWHI
20130620	INFILTRATED0037	USASPY	CHNBUS
20130624	VULNERABILITY0730	XXXRESBUS	USA
20130624	INFILTRATED0736	IRNGOVLEG	IGOEUREECGOVHRI
20130625	INFILTRATED0421	USASPY	CHNHAC
20130626	INFILTRATED0421	USASPY	CHNHAC
20130628	INFILTRATED0010	XXXWHI	USASPY
20130701	ARRESTED0170	USABUS	USAGOV
20130703	INFECTED1006	CHNGOV	USAHAC
20130707	ARRESTED0729	USAGOV	CHNLEGHACREB
20130708	INFILTRATED0044	XXXRES	USA
20130708	ARRESTED0729	USAGOV	CHNLEGHACREB
20130709	INFILTRATED0044	XXXRES	USA
20130709	ARRESTED0171	USA	CHNMILHAC
20130709	INFILTRATED0226	CHN	USABUS
20130712	ARRESTED0171	USA	CHNMILHAC
20130715	INFILTRATED0044	XXXRESHAC	USA
20130717	ARRESTED0729	USAGOV	CHNLEGHACREB

20130719	INFILTRATED0044	XXXRES	USA
20130720	ARRESTED0171	USA	CHNHAC
20130724	INFILTRATED0723	CHNHAC	USA
20130730	INFILTRATED0044	USASPY	BRARES
20130807	ARRESTED0596	CHN	USAHAC
20130807	ARRESTED0451	USAJUD	MEDWHIBUS
20130809	ARRESTED0451	CHN	CHNGOV
20130809	ARRESTED0171	USAGOVAGR	CHNMILHAC
20130812	ARRESTED0729	USAGOV	CHNLEGHACREB
20130812	INFILTRATED0044	XXXRES	USA
20130814	ARRESTED0171	USA	CHNMILHAC
20130814	ARRESTED0414	CHNMILGOVAGRHAC	USA
20130817	INFILTRATED0153	XXXRES	USA
20130818	ARRESTED0171	USA	CHNMILHAC
20130822	ARRESTED0171	USA	CHNMILHAC
20130824	ARRESTED0459	MDA	USA
20130825	INFILTRATED0723	CHNHAC	USA
20130827	INFILTRATED0044	USASPY	BRARES
20130827	ARRESTED0451	USAJUD	MEDWHIBUS
20130827	ARRESTED0451	CHN	CHNGOV
20130829	ARRESTED0171	USA	CHNMILHAC
20130903	INFILTRATED0037	USASPY	BRAGOVBUS
20130914	INFILTRATED0735	USA	USAOPP
20130915	ARRESTED0171	USA	CHNMIL
20130921	INFILTRATED0723	CHNHAC	USA
20130926	INFILTRATED0044	USASPY	BRARES
20130927	ARRESTED0170	USA	CHNMILHAC
20130927	ARRESTED0729	USA	CHNMILHAC
20130928	ARRESTED0021	ZAFJUD	USAMED
20131004	ARRESTED0146	USA	CHNGOVHAC
20131009	ARRESTED0021	ZAFJUD	USAMED
20131011	ARRESTED0021	ZAFJUD	USAMED
20131013	ARRESTED0171	USAGOVAGR	CHNMIL
20131017	ARRESTED0021	ZAFJUD	USAMED
20131019	ARRESTED0171	MEAREBCRM	CHNHACMIL
20131020	INFILTRATED0225	CHNHAC	USABUS
20131021	INFILTRATED0225	CHNHAC	USABUS
20131022	INFILTRATED0225	CHNHAC	USABUS
20131023	INFILTRATED0225	CHNHAC	USABUS
20131025	INFILTRATED0225	CHNHAC	USABUS
20131025	INFILTRATED0225	CHNHAC	USABUS

20131104	INFILTRATED0225	CHNHAC	USABUS
20131104	ARRESTED0729	USA	CHNGOVHAC
20131107	INFILTRATED0706	CHNMIL	CHNCVLGOVBUS
20131111	INFECTED0903	GBRSPYMED	GBR
20131111	ARRESTED0450	MNCUSA	USACOP
20131112	LEAKED0030	USAOPP	USAWHI
20131115	ARRESTED0170	USA	CHNLEGMIL
20131122	ARRESTED0171	USAGOVAGR	CHNMIL
20131202	INFECTED0945	USA	CHNGOV
20131206	ARRESTED0170	USA	CHNLEGMIL
20131207	ARRESTED0146	USA	CHNMIL
20131209	INFILTRATED0206	USASPYHAC	CHN
20131210	INFILTRATED0041	XXXHAC	MNCJPN
20131228	INFILTRATED0153	MNCJPN	MNCJPNMED
20140109	INFILTRATED0041	XXXHAC	MNCJPN
20140110	INFILTRATED0153	MNCJPN	MNCJPNMED
20140111	INFILTRATED0041	XXXHAC	MNCJPN
20140112	INFILTRATED0153	MNCJPN	MNCJPNMED
20140113	LEAKED0886	MNCCAN	USA
20140113	INFECTED0990	UKR	RUS
20140115	INFILTRATED0226	IRNHAC	USAMIL
20140116	LEAKED0101	USAOPP	USASPY
20140116	ARRESTED0144	MDA	USA
20140116	INFILTRATED0736	USASPYHAC	USASPY
20140119	ARRESTED0633	IGOCOPITP	XXXRES
20140119	PATCHED0821	USA	XXXWHI
20140122	INFECTED0907	USACOP	RUS
20140123	INFILTRATED0557	USASPY	CHNGOV
20140123	ARRESTED0416	USAGOV	RUS
20140130	DDOS0431	USA	DEU
20140202	INFILTRATED0736	XXXWHI	USASPYRES
20140211	INFILTRATED0736	XXXWHI	USASPYRES
20140212	ARRESTED0146	USAGOVAGR	CHNMIL
20140212	ARRESTED0633	USACOP	RUS
20140228	INFILTRATED0041	SYRMIL	USAMED
20140303	ARRESTED0171	USA	CHNMILHAC
20140304	INFILTRATED0153	CHNMIL	CHNRES
20140307	ARRESTED0729	USA	CHNSPY
20140308	LEAKED0101	USAOPP	USAGVMIL
20140311	INFECTED0924	MED	GBR
20140312	DDOS0430	USAOPP	CHNMED

20140314	INFILTRATED0735	USASPYWHI	MED
20140315	INFILTRATED0735	USASPYWHI	MED
20140317	INFILTRATED0736	USA	USAOPPHAC
20140320	ARRESTED0772	CANUAF	FRA
20140323	INFILTRATED0736	MED	USASPYWHI
20140325	ARRESTED0140	MARHAC	MAR
20140326	ARRESTED0146	USA	CHNMILLEG
20140327	ARRESTED0146	USA	CHNMILLEG
20140328	PATCHED0329	MNCUSA	MNCUSARES
20140328	VULNERABILITY0730	TURGOV	TURCVL
20140331	INFILTRATED0226	CHNHAC	CHNRESHAC
20140331	INFILTRATED0736	USASPY	USA
20140401	INFILTRATED0226	USASPY	BRAGOV
20140401	INFECTED1020	RUS	USA
20140402	INFECTED1020	RUS	USA
20140402	INFECTED1020	RUS	USA
20140402	INFECTED1020	RUS	USA
20140402	INFECTED1020	RUS	USA
20140403	INFECTED1020	RUS	USA
20140403	INFECTED1020	RUS	USA
20140404	INFECTED1020	RUS	USA
20140404	INFECTED1020	RUS	USA
20140405	ARRESTED0146	USAGOVAGR	CHNMIL
20140408	INFECTED1020	RUS	USA
20140408	LEAKED0224	MNCUSAMED	MNCUSAMEDCVLHAC
20140408	ARRESTED0302	USA	RUSHAC
20140408	INFILTRATED0037	USASPY	MNCUSAMED
20140409	INFECTED0903	USA	XXXWHICRM
20140410	INFECTED0970	USA	USACOP
20140411	INFILTRATED0564	IGOEUREECLEG	TWNGOV
20140412	DEFACED0505	SYRMIL	MED
20140412	INFECTED0970	MNCUSAMED	USASPY
20140413	INFILTRATED0420	CHNSPY	USAMIL
20140413	INFILTRATED0420	CHNSPY	USAMIL
20140414	INFILTRATED0420	CHNSPY	USAMIL
20140414	INFILTRATED0420	CHNSPY	USAMIL
20140414	ARRESTED0022	USALEG	USACOPLEG
20140414	ARRESTED0373	USACOPLEG	USALEG
20140414	ARRESTED0171	USAJUD	CHNMIL
20140415	INFECTED0892	MNCUSAMED	MNCUSA
20140415	INFILTRATED0736	MNCUSAMEDRES	MNCUSAMED

20140416	INFILTRATED0736	CHN	ISR
20140416	ARRESTED0633	USAGOV	USACOP
20140416	INFILTRATED0044	RUSHAC	USA
20140416	INFECTED0904	CHNBUS	CHNBUSMED
20140417	ARRESTED0218	DEUWHIGOV	USAWHI
20140417	ARRESTED0144	SWE	USA
20140417	ARRESTED0144	SWE	USA
20140422	VULNERABILITY0736	MNCUSA	MNCUSAHAC
20140423	INFECTED0970	MNCUSA	MNCUSAMED
20140424	ARRESTED0144	SWE	USA
20140425	ARRESTED0218	MNCUSA	SWE
20140425	ARRESTED0177	TWN	PHL
20140425	ARRESTED0146	RUSHAC	USAHAC
20140428	ARRESTED0146	USAGOVAGR	CHNBUS
20140428	ARRESTED0218	USA	CAN
20140428	INFILTRATED0513	USAGOVMED	USA
20140429	ARRESTED0218	USA	CAN
20140429	INFILTRATED0046	CHNHAC	USAGOVHAC
20140429	ARRESTED0218	CANMEDRES	USA
20140430	INFILTRATED0044	PSEREBHMS	ISRMED
20140501	ARRESTED0218	CANMEDRES	USA
20140502	INFILTRATED0564	MNCUSAWHI	MNCUSA
20140503	ARRESTED0171	USAGOV	CHN
20140507	INFILTRATED0735	GBRSPYWHI	XXXWHI
20140507	INFILTRATED0368	XXXWHI	USASPY
20140508	INFECTED0949	ROURES	GBR
20140508	ARRESTED0022	USAGOVAGR	CHN
20140512	INFILTRATED0347	CHNHAC	USA
20140512	ARRESTED0333	XXXWHI	SWE
20140512	ARRESTED0297	XXXWHI	ECUGOV
20140512	ARRESTED0218	DEUSPYLAB	USASPY
20140512	INFILTRATED0564	USASPY	USA
20140513	ARRESTED0171	USA	CHNLEGMIL
20140513	ARRESTED0218	DEULABSPYWHI	USAWHI
20140513	ARRESTED0715	GBR	USA
20140513	INFILTRATED0736	USA	USAOPP
20140513	LEAKED0101	JPN COP	JPN
20140513	ARRESTED0156	CAN COP	GBRCOP
20140513	ARRESTED0144	MNCUSAMED	USA
20140514	ARRESTED0170	USA	CHNMIL
20140514	ARRESTED0633	USAGOV	GBRHAC

20140515	LEAKED0864	USAMILOPP	MNCUSA
20140516	ARRESTED0691	USA	CHNLEGMIL
20140517	LEAKED0101	USASPY	USAGOV
20140518	INFILTRATED0044	USASPY	USAGOVLEGSPY
20140518	ARRESTED0200	MNCUSAJUD	MNCUSA
20140519	INFILTRATED0368	XXXRESHAC	USAGOV
20140519	INFILTRATED0043	MNCFINMED	MNCUSA
20140519	INFILTRATED0564	MNCUSAMEDBUS	MNCUSAMED
20140519	INFILTRATED0317	USASPY	USAJUD
20140519	INFILTRATED0564	USASPY	USAGOVLEG
20140519	ARRESTED0218	RUSHAC	ARE
20140519	ARRESTED0781	ESPCOP	RUS
20140519	ARRESTED0146	USA	CHNGOVHAC
20140519	ARRESTED0596	CHN	USACOPMED
20140519	ARRESTED0218	RUSHAC	ARE
20140519	ARRESTED0781	ESPCOP	RUS
20140519	ARRESTED0412	GBRCOPMIL	USAJUD
20140519	ARRESTED0488	XXXRES	USA
20140519	ARRESTED0218	RUSHAC	ARE
20140519	ARRESTED0781	ESPCOP	RUS
20140519	INFILTRATED0676	XXXRES	USA
20140519	LEAKED0612	USAEDU	CHN
20140519	INFILTRATED0153	USASPYWHI	USASPY
20140519	LEAKED0886	MNCCAN	USA
20140519	INFILTRATED0226	IRN	ISR
20140519	INFILTRATED0497	IMGMUSISI	MNCJPN
20140520	INFILTRATED0045	MNCHKG	CHN
20140520	INFECTED0970	LKA	LKAGOV
20140520	ARRESTED0245	ITA	USA
20140520	INFILTRATED0421	USASPY	DEU
20140520	INFILTRATED0564	USASPY	USAGOVLEG
20140520	INFECTED0904	CHNBUS	CHNBUSMED
20140520	INFILTRATED0226	IRNHAC	USABUSHAC
20140520	ARRESTED0156	AUSCOP	AUS
20140520	INFILTRATED0046	CHNGOV	USAHAC
20140520	INFILTRATED0152	USAMIL	CHNMIL
20140520	INFILTRATED0043	CHNGOV	USAHAC
20140520	INFILTRATED0736	USA	CHN
20140520	ARRESTED0727	USAMIL	SYR
20140520	INFECTED0945	IGOUNO	FRA
20140520	INFILTRATED0011	SGP	NGO

20140520	ARRESTED0218	NGO	USAGUM
20140521	ARRESTED0175	AUSCVL	AUS
20140521	ARRESTED0175	AUSCVL	AUS
20140521	VULNERABILITY0730	USA	MEAREB
20140521	INFECTED0996	MNCUSAMEDHAC	IDNCVL
20140521	DDOS0431	GBRGOV	USABUS
20140521	INFILTRATED0497	XXXRES	USAGOVBUS
20140521	ARRESTED0729	USA	CHNMIL
20140521	DDOS0430	MNCUSAMED	GBRWHI
20140521	ARRESTED0384	RUSHAC	USA
20140521	ARRESTED0384	RUSHAC	USAHAC
20140521	INFILTRATED0800	USASPY	BELMED
20140521	INFILTRATED0226	IGOWSTNAT	POLRES
20140521	ARRESTED0412	USACVL	USAJUD
20140521	INFECTED0892	RUS	RUSHAC
20140522	INFILTRATED0226	IGOWSTNAT	POLRES
20140522	INFILTRATED0736	FRAHAC	MNCUSAMED
20140523	INFILTRATED0800	USASPY	BELMED
20140525	ARRESTED0662	USABUS	CHNGOV
20140525	ARRESTED0412	USAMEDGOV	USACOP
20140527	VULNERABILITY0730	MNCUSAMEDRES	MNCUSAMEDHAC
20140527	ARRESTED0333	USACVL	USACOP
20140527	ARRESTED0021	ROUGOV	USA
20140528	VULNERABILITY0730	MNCKOR	MNCKORGOV
20140529	LEAKED0342	XXXWHI	USASPY
20140529	ARRESTED0463	SOMCVL	USA
20140529	ARRESTED0021	ROUGOV	USA
20140529	INFILTRATED0044	USAHAC	XXXRES
20140530	INFILTRATED0394	USASPY	USAGOVLEG
20140602	PATCHED0020	MNCUSA	USAREB
20140602	ARRESTED0218	USACVL	USA
20140602	PATCHED0840	MNCUSARES	MNCUSA
20140602	ARRESTED0158	CHN	CHNGOV
20140603	INFILTRATED0044	GBRWHISPY	DEUHAC
20140603	ARRESTED0022	USAGOV	KSVGGOVHAC
20140603	INFILTRATED0564	CHN	USAHACRES
20140603	INFILTRATED0564	USAELIPTY	USAGOV
20140604	INFILTRATED0736	USA	MNCUSAMED
20140604	INFILTRATED0800	USASPYHAC	USASPY
20140604	INFILTRATED0736	MED	XXXANT
20140605	INFECTED0990	XXXRES	MEAREB

20140605	INFILTRATED0736	MED	USA
20140605	INFILTRATED0044	USA	USABUS
20140605	INFILTRATED0736	XXXWHI	SAU
20140605	INFILTRATED0530	USA	USASPY
20140605	LEAKED0612	XXXWHI	USASPY
20140605	ARRESTED0159	MYSGOV	KSVCVL
20140605	ARRESTED0022	USA	CHN
20140606	ARRESTED0297	CHNHAC	USA
20140606	INFECTED0970	USAGOV	USAHAC
20140610	INFILTRATED0225	CHNHACRES	USABUS
20140610	LEAKED0101	USAOPP	USASPY
20140610	LEAKED0342	USA	USAGOVWHI
20140610	INFILTRATED0630	USABUS	USA
20140611	INFILTRATED0564	USAMILHAC	USA
20140611	INFILTRATED0736	XXXWHI	SAU
20140612	DDOS0431	CHNHAC	USA
20140612	INFILTRATED0226	CHNHAC	JPNHAC
20140612	INFECTED0945	MED	UKR
20140612	INFECTED0892	XXXRES	GBR
20140612	INFECTED0949	USASPY	DEUGOVAGR
20140612	ARRESTED0396	GBRCOP	IRL
20140612	ARRESTED0335	IRLCOPMIL	GBRHACMED
20140612	INFILTRATED0153	XXXWHI	USASPYHAC
20140612	INFECTED1031	MEXHAC	CAN
20140612	ARRESTED0156	GBRCOP	IRL
20140612	ARRESTED0156	GBRCOP	IRL
20140614	ARRESTED0022	MED	IRL
20140616	ARRESTED0218	IRL	GBRHAC
20140617	ARRESTED0156	GBRCOP	IRL
20140617	DDOS0430	IRN	USA
20140625	ARRESTED0171	CHN	CHNMILHAC
20140625	ARRESTED0463	XXXRES	USA
20140626	ARRESTED0596	CHN	USAHACBUS
20140626	ARRESTED0156	JPNCOP	XXXRES
20140626	LEAKED0101	USAWHIMIL	USASPYWHI
20140626	INFILTRATED0736	USA	USACOP
20140627	ARRESTED0384	USAOPP	USA
20140627	VULNERABILITY0730	USAJUD	USASPY
20140630	LEAKED0101	USAWHIMIL	USASPYWHI
20140705	ARRESTED0384	USAOPP	USA
20140708	VULNERABILITY0730	USAJUD	USASPY

20140708	ARRESTED0156	GBRCOP	GBR
20140709	ARRESTED0335	GBRCOP	GBR
20140709	ARRESTED0156	GBRCOP	GBRWHI
20140710	ARRESTED0334	GBRCOPHAC	GBR
20140710	ARRESTED0156	NLDCOP	NLD
20140710	ARRESTED0384	USAOPP	USA
20140710	ARRESTED0727	GBR	USA
20140710	ARRESTED0156	GBRCOP	GBRWHI
20140710	VULNERABILITY0730	USAGOV	USA
20140711	INFILTRATED0260	BGDMED	BGD
20140711	ARRESTED0112	USACOP	USACVL
20140711	ARRESTED0112	USACOP	USACVL
20140711	INFILTRATED0010	TURHAC	AUSMED
20140711	ARRESTED0384	USA	IRN
20140712	INFILTRATED0010	PAKLEGMIL	INDHAC
20140714	INFILTRATED0286	KWTRESHAC	IRQGOVHAC
20140714	LEAKED0612	IRQ	KWTRESHAC
20140714	INFILTRATED0011	KWTLEGRESHAC	IND
20140714	ARRESTED0695	USACOPLEG	XXXHAC
20140715	DEFACED0505	PAKHAC	INDHAC
20140715	INFILTRATED0043	PAKHAC	NGAGOVMED
20140715	ARRESTED0171	USA	EST
20140715	INFILTRATED0286	MNCUSAMED	PHLHACCOP
20140716	ARRESTED0159	DNKHAC	SWE
20140716	INFECTED0970	USAGOV	GBR
20140716	ARRESTED0159	USAHAC	USA
20140716	ARRESTED0781	USAHAC	USA
20140717	DDOS0355	ROU	ROUHACGOV
20140718	DEFACED0505	SYRMEDMIL	TURHAC
20140718	ARRESTED0158	ROUGOV	RUSRES
20140722	INFILTRATED0736	USA	SYR
20140722	DEFACED0505	MYSHAC	BGDHAC
20140722	INFILTRATED0041	IRNGOVGOVMILHAC	USA
20140723	DDOS0103	XXXHAC	USAHAC
20140723	ARRESTED0414	GBRGOVBUS	BELHAC
20140723	ARRESTED0373	USACOP	UKR
20140724	LEAKED0101	KORMEDHAC	USAHACMIL
20140725	ARRESTED0156	GBRCOP	XXXHAC
20140728	LEAKED0027	PSE	ISRCVL
20140729	DDOS0051	XXXHAC	USAHAC
20140730	DDOS0058	MYSHAC	PHLHACGOVMED

20140731	INFILTRATED0043	TURLEGHAC	ARGMEDBUS
20140731	INFILTRATED0044	XXXHAC	USAGOVAGR
20140731	INFECTED0903	XXXANTRES	NGA
20140731	DEFACED0161	PAKHAC	ISRMED
20140801	INFECTED0891	XXXANTRES	CAN
20140801	INFILTRATED0736	USA	CHNHAC
20140804	ARRESTED0218	POLEDU	IRQGOVHAC
20140818	INFILTRATED0286	PAKHAC	MNCINDMEDHACGOVBUS
20140819	INFILTRATED0421	USASPY	USASPYLAB
20140820	DDOS0313	USACOP	XXXHAC
20140820	ARRESTED0156	ROUCRM COP	BGRCVL
20140820	INFILTRATED0011	TURHAC	IRQGOVHAC
20140820	INFECTED0959	XXXRESCRM	CAN
20140820	ARRESTED0302	XXXANT	CHNANTEDU
20140820	DDOS0431	PSEREBHMS	USA
20140823	ARRESTED0218	SGP	MYS
20140823	INFECTED0992	MNCUSA	USA
20140825	INFILTRATED0736	XXXANT	AUS
20140827	INFILTRATED0736	BRA	BRAHAC
20140828	INFILTRATED0190	NORMED	USA
20140829	INFILTRATED0564	USA	USAGOVAGR
20140903	ARRESTED0021	ROU	USA
20140903	ARRESTED0402	USACOP	EST
20140904	INFILTRATED0368	TURLEGHAC	HUNHAC
20140911	ARRESTED0297	USAOPPHILAB	RUS
20140912	INFILTRATED0043	BRAHAC	BRAHACMIL
20140915	INFILTRATED0564	MYSHAC	USAGOVBUS
20140915	INFILTRATED0735	USAOPPHI	XXXWHI
20140918	INFILTRATED0043	INDHAC	BGDHACGOV
20140918	ARRESTED0315	ROUJUD	USA
20140918	PATCHED0354	XXXANTRES	USA
20140919	ARRESTED0006	XXXHAC	USACOP
20140919	ARRESTED0144	CZE	USA
20140922	INFILTRATED0368	CANHAC	PHL
20140925	INFILTRATED0564	MNCJPNHAC	XXXHACLEG
20140926	DEFACED0505	PAKHACMIL	CHNGOVHAC
20141001	INFECTED0998	XXXANT	MNCUSAMED
20141001	INFILTRATED0195	IRNGOV	USAMIL
20141001	INFILTRATED0226	TURHAC	ITAMED
20141003	INFILTRATED0043	SYRMIL	GBRMED
20141006	DEFACED0106	SYRMIL	USAGOVHACLAB

20141008	DEFACED0106	SYRMIL	USAGOVHACLAB
20141008	ARRESTED0144	EST	USA
20141009	ARRESTED0158	JPNGOV	CHNAGR
20141009	VULNERABILITY0584	USAJUD	USACOP
20141010	ARRESTED0218	FRAREB	GBR
20141010	ARRESTED0177	CZELABHACRES	GRC
20141010	INFILTRATED0286	SYRHAC	IGOUNOCVL
20141011	INFECTED1006	ROUHAC	USA
20141014	INFILTRATED0011	XXXHAC	CHNMEDHAC
20141014	INFILTRATED0010	INDHAC	PAKHAC
20141014	INFILTRATED0736	XXXHAC	USAGOVAGGRESSPY
20141014	INFILTRATED0226	RUS	UKRGOVHAC
20141017	ARRESTED0288	LVA	USA
20141018	ARRESTED0117	GBRGOV	GBR
20141018	DDOS0430	CHNHAC	USA
20141021	INFILTRATED0753	SYRHACMIL	MEDHAC
20141021	INFILTRATED0226	BRA	MNCESPCVL
20141021	ARRESTED0006	GBRCOP	XXXWHI
20141021	INFILTRATED0010	PHLHAC	PHLCOP
20141022	ARRESTED0006	THACOP	DZA
20141022	PATCHED0808	XXXRES	XXXRESMED
20141022	LEAKED0342	CANMIL	RUS
20141022	DDOS0430	CHNHAC	USAGOV
20141024	INFILTRATED0542	ISR	USAELIGOV
20141025	INFILTRATED0043	XXXHAC	TURHACGOV
20141027	LEAKED0030	ITAHAC	ITAGOVHAC
20141103	DEFACED0505	PAKHAC	JPNANTMED
20141230	INFILTRATED0736	MNCUSAMED	MNCUSAMEDRES
20150119	INFILTRATED0368	EGYHAC	ARE
20150325	INFILTRATED0753	TURHAC	SVK
20150408	DDOS0431	XXXHAC	GBRMED
20150410	INFILTRATED0010	TURHAC	XXXRESHAC
20150411	ARRESTED0171	USA	EST
20150602	VULNERABILITY0730	USALEG	USASPY
20150617	INFECTED0892	USAGOVHACRESHLH	USAGOVHAC
20150623	INFILTRATED0010	PAKHAC	ISR
20150716	INFILTRATED0557	PAKSPYRES	INDHAC
20150901	INFILTRATED0286	TURHAC	MNCKORHAC
20150909	INFILTRATED0260	SYRMIL	USALEG
20150910	INFILTRATED0010	TURLEGHAC	AUT
20150916	INFECTED0949	XXXRES	IRN

20150919	INFILTRATED0041	SYRMIL	MNCUSAMEDHAC
20150922	ARRESTED0416	USAGOV	USA
20150924	INFECTED0945	USAHAC	USAMIL
20150928	INFILTRATED0041	SYRLEG	ISRMEDHAC
20151008	INFECTED1010	CHNHAC	USAMEDHAC
20151008	ARRESTED0022	ROUHAC	USA
20151008	INFILTRATED0368	MNCGBRMEDMIL	QAT
20151013	ARRESTED0159	MED	AUSGOVMED
20151013	INFILTRATED0010	XXXHACLEG	USA
20151013	PATCHED0252	USA	XXXRESANT
20151015	INFILTRATED0037	USASPY	CHNBUS
20151016	ARRESTED0200	USACOP	XXXHAC
20151016	ARRESTED0140	MARHAC	MAR
20151017	INFILTRATED0011	INDHAC	BGDMEDGOV
20151017	INFILTRATED0043	SYRHACMIL	SAUGOVMED
20151018	INFILTRATED0041	PSEHAC	ISRMED
20151019	ARRESTED0159	ROU	CANCOP
20151019	INFILTRATED0011	PAKHAC	INDGOVMED
20151020	ARRESTED0789	USAGOVHAC	USA
20151020	INFILTRATED0010	CHN	USAMIL
20151020	INFILTRATED0043	XXXHAC	TURGOVHAC
20151021	ARRESTED0402	ROU	GBRGOV
20151022	ARRESTED0428	XXXHACLEG	XXXHAC
20151022	ARRESTED0559	ROU	USA
20151022	INFILTRATED0286	TURHAC	INDMED
20151027	INFILTRATED0010	TURHAC	KORMED
20151027	DEFACED0505	PAKHAC	IND
20151027	ARRESTED0156	GBRCOP	XXXRES
20151027	INFILTRATED0736	AUSGOVHLHJUD	AUS
20151027	INFILTRATED0226	RUS	GEOGOV
20151027	INFILTRATED0045	FRAHAC	MNCUSAMEDHAC
20151027	ARRESTED0461	RUSGOVBUS	ESTJUD
20151027	DDOS0509	XXXHAC	TURGOVMED
20151027	DDOS0253	ESPGOV	ESPHAC
20151028	INFILTRATED0010	MNCUSAMEDHAC	RUSMUSMED
20151028	INFILTRATED0736	INDBUSMED	MED
20151028	ARRESTED0022	USAGOV	GBR
20151028	INFILTRATED0736	XXXHAC	USA
20151028	ARRESTED0171	USA	MEDCOP
20151029	ARRESTED0334	USAGOVHAC	ROU
20151029	INFILTRATED0753	SYRHAC	NGAGOVMED

20151029	ARRESTED0218	NLDCVL	ESP
20151029	DEFACED0505	TUNHACRES	MEAREBHACBUS
20151030	INFILTRATED0421	IRQ	MNCUSAMED
20151030	INFILTRATED0041	XXXANT	FRAHAC
20151030	ARRESTED0006	GBRHAC	GBRCOP
20151030	INFILTRATED0365	GBRHAC	GBRRES
20151030	LEAKED0101	JPNMED	JPNCOP
20151030	ARRESTED0140	MDV	GBR
20151030	INFILTRATED0260	IND	RUSBUS
20151030	DDOS0164	IRN	XXXHAC
20151030	DEFACED0505	TURHAC	SYRHACMIL
20151030	INFILTRATED0676	PAKHAC	MNEHAC
20151030	INFECTED0903	XXXRESANT	USA
20151030	LEAKED0342	TURCOP	XXXHAC
20151031	INFILTRATED0226	USA	CHNHAC
20151031	INFILTRATED0011	SAUHAC	USAHACGOV
20151101	ARRESTED0416	RUSGOV	RUSCVL
20151102	INFILTRATED0011	IRQHAC	SAUHAC
20151106	INFILTRATED0225	MEAREBGOV	IRQHAC
20151110	INFILTRATED0707	USAGOVLAB	USAGOV

# Appendix D

## Scoring rules

### D.1 Actor scoring

0: Incorrect    1: Partially correct    2: Correct    3: Correct but reversed order

1. Actors are coded as correct if at least half of the three-letter codes assigned to them are accurate.
2. A score of 1 is assigned if only one of the two actors passes this bar.
3. When the actors coded are accurate to the text but not both relevant to the verb phrase in question, a score of 1 is assigned.
4. When the verb phrase in question references only one actor, a code of 2 is assigned if either actor is accurate.
5. A 3 is awarded when actors satisfy the criteria for category 2 but are in reverse order.
6. Syrian Electronic Army is frequently coded as "SYRMIL" which is incorrect. It is assigned a score of 2 due to the 3-letter codes being at least 1/2 accurate.

7. The actor code HAC is appended to many actors erroneously. If at least half of the other related codes are accurate, then the label is determined to be accurate.
8. Generally, when a sentence or story contains insufficient information to make a precise judgment, partial credit (1) is awarded unless it appears more likely than not that the coding is accurate (2).
9. Some event codes are commonly associated with three actors. For example: *A* leaked information about *B* to *C*. These are generally coded correct if the two coded actors are among the three overall.
10. When multiple actors exist on either side of a code (*A* and *B* arrested *C* and *D*) and only one side is coded, a point of 1 is awarded. If one actor from each side is coded correctly, then a score of 2 is awarded.
11. Sometimes the intended target is unknown or unreported. If, on the other hand, the researchers responsible for finding malware are accurately reported, the event is considered accurate. Essentially, if malware is discovered by researchers, that malware clearly made its way onto the researchers' computers somehow. However, this is clearly an interesting case that occurs frequently in the cybersecurity literature but not so frequently in traditional event coding domains. Events that contain phrases similar to "Researchers report..." are often difficult to code accurately.
12. Extraditions are often correctly coded as "arrests" but their actors are only partially accurate as "*A* was arrested by *B* and extradited to *C*" is coded B-C rather than A-B or A-C.
13. "The Israeli site of American band..." is coded correct for either USA and ISR.

## D.2 Event scoring

0: Incorrect

1: Partially correct

2: Correct

1. Events are coded as 2 (accurate) if they accurately describe an event referenced in the original article. This results in the occasional case of “the right thing being coded for the wrong reason.”
2. Events are coded as 2 (accurate) if the actual verb phrase implies its occurrence unless there is reason to believe this implication is false (a website being defaced implies it was breached).
3. Events are coded as 1 (partially accurate) if the actual verb phrase is ambiguous but the article implies an alternative event type (i.e. “attack” is coded as “ddos” but the context implies “defacement”).
4. Events are coded as 1 (partially accurate) if the relevant verb phrase is not the focus of the story. For example “Chinese officials deny that Chinese hackers have infiltrated U.S. firms.” would be coded as a 1 for “infiltrated.” Similarly, when events are speculative, a score of 1 is assigned. For example: “Officials worry that China could hack into US government machines.”

# Bibliography

- Akamai (Q4, 2015), “State of the Internet - Security,” 2.
- Apache Software Foundation (2010), “Apache OpenNLP Developer Documentation v. 1.5.3,” Online, <http://www.opennlp.apache.org>.
- Associated Press (October 26, 2015), “UK police arrest 15-year-old boy after telecoms cyberattack,” <http://news.yahoo.com/uk-police-arrest-15-old-boy-telecoms-cyberattack-190328711.html>.
- Azar, E. E. (1980), “The Conflict and Peace Data Bank (COPDAB) Project,” *The Journal of Conflict Resolution*, 24.
- Barzashka, I. (2013), “Are Cyber-Weapons Effective?” *The RUSI Journal*, 158, 48–56.
- Bauer, J. (2014), “Shift-Reduce Constituency Parser,” Online, <http://nlp.stanford.edu/software/srparser.shtml>.
- BBC (October 31, 2015), “Bangladeshi secular publisher hacked to death,” Online, <http://www.bbc.co.uk/news/world-asia-34688245>.
- Boschee, E., Lautenschlager, J., O’Brien, S., Shellman, S., Starz, J., and Ward, M. (2015), “ICEWS Coded Event Data,” V15, <http://dx.doi.org/10.7910/DVN/28075>.
- Brecher, M. and Wilkenfeld, J. (2000), *A Study of Crisis*, University of Michigan Press.
- Cimpanu, C. (November 6, 2015), “Military Contractors That Used Russian Programmers for DoD Software Get Fined by US Govt,” *Softpedia Security News*, <http://news.softpedia.com/news/military-contractors-that-used-russian-programmers-for-dod-software-get-fined-by-us-govt-495827.shtml>.
- Cimpanu, C. (September 9, 2012), “Android Malware Secretly Subscribes Victims to Premium SMS Services,” *Softpedia Security News*, <http://news.softpedia.com/news/android-malware-secretly-subscribes-victims-to-premium-sms-services-491264.shtml>.

- Clapper, J. R. (2015), “Worldwide Threat Assessment of the US Intelligence Community,” [http://cdn.arstechnica.net/wp-content/uploads/2015/02/Clapper\\_02-26-15.pdf](http://cdn.arstechnica.net/wp-content/uploads/2015/02/Clapper_02-26-15.pdf).
- Clarke, R. (2009), “War from Cyberspace,” *National Interest*, pp. 31–36.
- Collier, P., Elliott, L., Hegre, H., Reynal-Querol, M., and Sambanis, N. (2005), *Breaking the Conflict Trap: Civil War and Development Policy*, Oxford University Press.
- Constantin, L. (April 15, 2011), “Malaysian Man Admits Hacking into Federal Reserve,” *Softpedia Security News*, <http://news.softpedia.com/news/Malaysian-Admits-Hacking-into-Federal-Reserve-195245.shtml>.
- Constantin, L. (February 25, 2009), “Avira Website XSSed,” *Softpedia Security News*, <http://news.softpedia.com/news/Avira-Website-XSSed-105393.shtml>.
- Dhillon, P. S., Foster, D. P., and Ungar, L. H. (2015), “Eigenwords: Spectral Word Embeddings,” *Journal of Machine Learning Research*, 16, <http://www.pdhillon.com/dhillon15a.pdf>.
- Dumais, S. T. (2004), “Latent semantic analysis,” *Annual Review of Information Science and Technology*, 38, 188–230.
- Dumais, S. T., Furnas, G. W., Landauer, T. K., Deerwester, S., and Harshman, R. (1988), “Using latent semantic analysis to improve information retrieval,” *Proceedings of CHI88 Conference on Human Factors in Computing Systems*, pp. 281–285.
- Erumban, A. A. and Das, D. K. (2016), “Information and communication technology and economic growth in India,” *Telecommunications Policy*, 40, 412–431.
- Fearon, J. D. and Laitin, D. D. (2003), “Ethnicity, Insurgency, and Civil War,” *American Political Science Review*, 97, 75–90.
- Finkel, J. R., Grenager, T., and Manning, C. (2005), “Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling,” *Proceedings of the 43rd Annual Meeting of the Association of Computational Linguistics*, pp. 363–370, <http://nlp.stanford.edu/~manning/papers/gibbscrf3.pdf>.
- Fisher, D. (2014), “OpenSSL Fixes TLS Vulnerability,” *Threat Post*, <https://threatpost.com/openssl-fixes-tls-vulnerability/105300>.
- Fleischman, M. and Hovy, E. (2002), “Fine Grained Classification of Named Entities,” *Proceedings of the 19th international conference on computational linguistics*, USC Information Science Institute.

- Gartzke, E. (2013), “The Myth of Cyberwar: Bringing War in Cyberspace Back Down to Earth,” *International Security*, 38, 41–73.
- Gellman, B. and Nakashima, E. (August 30, 2013), “U.S. spy agencies mounted 231 offensive cyberoperations in 2011, documents show,” *The Washington Post*, <http://www.washingtonpost.com/>.
- Gerner, D. J., Schrodtt, P. A., and Yilmaz, O. (2002), “The Creation of CAMEO (Conflict and Mediation Event Observations): An Event Data Framework for a Post Cold War World,” Prepared for the Annual Meeting of the American Political Science Association 2002.
- Gibson, W. (1982), “Burning Chrome,” in *Omni*, General Media, Inc., New York City.
- Gleditsch, N. P., Wallensteen, P., Eriksson, M., Sollenberg, M., and Strand, H. (2002), “Armed Conflict 1946-2001: A New Dataset,” *Journal of Peace Research*, 39, 615–637.
- Goldberg, Y. and Levy, O. (2014), “word2vec Explained: Deriving Mikolov et al.’s Negative-Sampling Word-Embedding Method,” *arXiv*.
- Google (2015), “word2vec,” <https://code.google.com/p/word2vec/>.
- Graham, R. (April 1, 2015), “Pin-pointing China’s attack against GitHub,” *Errata Security*, <http://blog.erratasec.com/2015/04/pin-pointing-chinas-attack-against.html>.
- Honaker, J., King, G., and Blackwell, M. (2011), “Amelia II: A Program for Missing Data,” *Journal of Statistical Software*, 45, 1–47.
- Hornik, K. (2015), “Package ‘openNLP’ v. 0.2-4,” Online, <http://cran.r-project.org/web/packages/openNLP/openNLP.pdf>.
- Ilascu, I. (April 8, 2015), “Russian Hackers Allegedly Behind White House Network Cyber Attack,” *Softpedia Security News*, <http://news.softpedia.com/news/Russian-Hackers-Allegedly-Behind-White-House-Network-Cyber-Attack-477916.shtml>.
- Intel Security (2014), “Net Losses: Estimating the Global Cost of Cybercrime,” Tech. rep., Center for Strategic and International Studies.
- Kim, H. K., Kim, H., and Cho, S. (2015), “Bag-of-Concepts: Comprehending Document Representation through Clustering Words in Distributed Representation,” *SNU Data Mining Center*, 12.

- King, G. and Lowe, W. (2003), "An Automated Information Extraction Tool for International Conflict Data with Performance as Good as Human Coders: A Rare Events Evaluation Design," *International Organization*, 57, 617–642, <http://gking.harvard.edu/files/gking/files/infoex.pdf?m=1360039060>.
- Kovacs, E. (April 13, 2014b), "Security Brief Heartbleed Bug, Google Vulnerabilities, Arrested Hackers," *Softpedia Security News*, [/urlhttp://news.softpedia.com/news/Security-Brief-Heartbleed-Bug-Google-Vulnerabilities-Arrested-Hackers-437327.shtml](http://news.softpedia.com/news/Security-Brief-Heartbleed-Bug-Google-Vulnerabilities-Arrested-Hackers-437327.shtml).
- Kovacs, E. (August 22, 2013a), "Anonymous Responds to FBI's Claims That Hacker Movement Is Dismantled," Online, [http://news.softpedia.com/news/Anonymous-Responds-to-FBI-s-Claims-That-Hacker-Movement-Is-Dismantled-377422.shtml\\_3](http://news.softpedia.com/news/Anonymous-Responds-to-FBI-s-Claims-That-Hacker-Movement-Is-Dismantled-377422.shtml_3).
- Kovacs, E. (February 13, 2013b), "China is Aggressively Hacking US Companies for Economic Gain, Officials Say," *Softpedia Security News*, <http://news.softpedia.com/news/China-Is-Aggressively-Hacking-US-Companies-for-Economic-Gain-Officials-Say-329245.shtml>.
- Kovacs, E. (February 2, 2013g), "Security Brief: Newspaper Hacks, China," *Softpedia Security News*, <http://news.softpedia.com/news/Security-Brief-Newspaper-Hacks-China-326191.shtml>.
- Kovacs, E. (February 21, 2012a), "Anonymous Members Arrested for Hacking Greek Ministry of Justice," *Softpedia Security News*, <http://news.softpedia.com/news/Anonymous-Members-Arrested-for-Hacking-Greek-Ministry-of-Justice-254044.shtml>.
- Kovacs, E. (January 12, 2014a), "Hackers Hijack 3,500 Domains after Breaching Systems of Montenegro Registrar," *Softpedia Security News*, <http://news.softpedia.com/news/Hackers-Hijack-3-500-Domains-After-Breaching-Systems-of-Montenegro-Registrar-416194.shtml>.
- Kovacs, E. (July 6, 2012d), "Tick Tock: It's Lights Out for DNSChanger-Infected Computers on July 9," *Softpedia Security News*, <http://news.softpedia.com/news/Tick-Tock-It-s-Lights-Out-for-DNSChanger-Infected-Computers-on-July-9-Video-279700.shtml>.
- Kovacs, E. (June 24, 2013c), "Edward Snowden: the US Hacked China's Tsingua University, Mobile Phone Companies," *Softpedia Security News*, <http://news.softpedia.com/news/Edward-Snowden-US-Hacked-China-s-Tsinghua-University-Mobile-Phone-Companies-362901.shtml>.
- Kovacs, E. (March 21, 2013f), "Nigerian Ministry of Foreign Affairs, 3 Other Government Sites Hacked," *Softpedia Security News*, <http://news.softpedia.com/news/Nigerian-Ministry-of-Foreign-Affairs-3-Other-Government-Sites-Hacked-362901.shtml>.

- [//news.softpedia.com/news/Nigerian-Ministry-of-Foreign-Affairs-3-Other-Government-Sites-Hacked-339082.shtml\\_3](http://news.softpedia.com/news/Nigerian-Ministry-of-Foreign-Affairs-3-Other-Government-Sites-Hacked-339082.shtml_3).
- Kovacs, E. (March 4, 2013d), “Hundreds of Sites Hacked in Conflict Between Malaysia and Philippines Hacktivists,” *Softpedia Security News*, <http://news.softpedia.com/news/Hundreds-of-Sites-Hacked-in-Conflict-Between-Malaysia-and-Philippines-Hacktivists-334047.shtml>.
- Kovacs, E. (November 20, 2012c), “SQL Injection Vulnerability Used to Deface Israeli Microsoft Sites, Hacker Says,” *Softpedia Security News*, <http://news.softpedia.com/news/SQL-Injection-Vulnerability-Used-to-Deface-Israeli-Microsoft-Sites-Hacker-Says-308229.shtml>.
- Kovacs, E. (November 8, 2013e), “Indonesian Hackers Briefly Disrupt Site of Australia’s ASIO,” *Softpedia Security News*, <http://news.softpedia.com/news/Indonesian-Hackers-Briefly-Disrupt-Site-of-Australia-s-ASIO-398392.shtml>.
- Kovacs, E. (September 24, 2012b), “Iranian Official: We Did Not Launch Cyberattacks on American Banks,” *Softpedia Security News*, <http://news.softpedia.com/news/Iranian-Officials-We-Did-Not-Launch-Cyberattacks-on-American-Banks-294412.shtml>.
- Lee, J. W. and Becker, K. (2014), “Relationship between information communications technology, economic growth and carbon emissions: evidence from panel analysis of the G20,” *Global Business and Economics Review*, 17, 35–50.
- Lee, N. (2013), *Counterterrorism and Cybersecurity: Total Information Awareness*, Springer.
- Leetaru, K. and Schrodt, P. (2013), “GDELT Global Data on Events, Language, and Tone 1979-2012,” *International Studies Association Annual Conference*, <http://www.gdeltproject.org>.
- Lustick, I., O’Brien, S., Shellman, S., Siedlecki, T., and Ward, M. (2015), “ICEWS Events of Interest Ground Truth Data Set,” Online, <http://dx.doi.org/10.7910/DVN/28119>.
- Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014), “The Stanford CoreNLP Natural Language Processing Toolkit,” *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics System Demonstrations*, pp. 55–60.
- McClelland, C. (1978), “World Event/Interaction Survey (WEIS) 1966-1978 (ICPSR 5211),” *Inter-University Consortium for Political and Social Research*.

- Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013a), “Distributed Representations of Words and Phrases and their Compositionality,” *arXiv*, <http://arxiv.org/abs/1310.4546>.
- Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013b), “Efficient Estimation of Word Representations in Vector Space,” *Proceedings of Workshop at ICLR*, <http://arxiv.org/pdf/1301.3781.pdf>.
- Minhas, S. and Radford, B. J. (2016), “Enemy at the Gates: Variation in Economic Growth from Civil Conflict,” *Journal of Conflict Resolution*, pp. 1–25, <http://jcr.sagepub.com/content/early/2016/05/03/0022002716639100.abstract>.
- Minhas, S., Hoff, P. D., and Ward, M. D. (2015), “Relax, Tensors Are Here: Dependencies in International Processes,” *ArXiv e-prints*.
- Morgan, S. (2015), “Cybersecurity Market Reaches \$75 Billion In 2015; Expected To Reach \$170 Billion By 2020,” *Forbes*.
- Nadeau, D. and Sekine, S. (2007), “A survey of named entity recognition and classification,” *Linguisticae Investigationes*, 30, 3–26, National Research Council Canada and New York University.
- Nakashima, E. (2016), “Russian government hackers penetrated DNC, stole opposition research on Trump,” Online, June 14, 2016.
- Norris, C. (2015), “PETRARCH 2: PETRARCHer,” <https://github.com/openeventdata/petrarch2/blob/master/Petrarch2.pdf>.
- Nye, J. S. (2011), “Nuclear Lessons for Cyber Security?” *Strategic Studies Quarterly*, pp. 18–38.
- OED (2014), “Oxford English Dictionary,” [www.oxforddictionaries.com](http://www.oxforddictionaries.com), Definition of cyberspace.
- Open Event Data Alliance (2015a), “PETRARCH Python Engine for Text Resolution and Related Coding Hierarchy,” <http://www.github.com/openeventdata/petrarch>.
- Open Event Data Alliance (2015b), “Phoenix Data Project,” Online, [phoenixdata.org](http://phoenixdata.org).
- Open Event Data Alliance (2015c), “Phoenix Pipeline,” Online, <http://phoenix-pipeline.readthedocs.org/en/latest>.
- Open Event Data Alliance (2016), “Phoenix Data Project,” Online, <http://phoenixdata.org>.

- Palmer, G., D’Orazio, V., Kenwich, M., and Lane, M. (2015), “The MID4 dataset, 2002-2010: Procedures, coding rules and description,” *Conflict Management and Peace Science*, 32, 222–242.
- Pettersson, T. and Wallensteen, P. (2015), “Armed Conflict, 1946-2014,” *Journal of Peace Research*, 52.
- Rajnovic, D. (2012), “Cyberspace - What is it?” [blogs.cisco.com/security/cyberspace-what-is-it/](http://blogs.cisco.com/security/cyberspace-what-is-it/).
- Reed, T. (2005), *At the Abyss: An Inside’s History of the Cold War*, Presidio Press.
- Řehůřek, R. and Sojka, P. (2010), “Software Framework for Topic Modelling with Large Corpora,” in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pp. 45–50, Valletta, Malta, ELRA, <http://is.muni.cz/publication/884893/en>.
- Rid, T. (2013), *Cyber War Will Not Take Place*, Oxford University Press, New York City.
- Rong, X. (2016), “word2vec Parameter Learning Explained,” *arXiv*.
- Ross, A. (2016), “Want a job? Try online security,” *WIRED*.
- Rubin, D. B. (2004), *Multiple Imputation for Nonresponse in Surveys*, John Wiley and Sons, New York.
- Rusu, D., Dali, L., Fortuna, B., Grobelnik, M., and Mladenic, D. (2007), “Triplet Extraction from Sentences,” *Proceedings of the 10th International Multiconference ‘Information Society - IS 2007’*, A, 218–222.
- Sanger, D. E. (June 1, 2012), “Obama Order Sped Up Wave of Cyberattacks Against Iran,” *The New York Times*.
- Sanger, D. E. and Shanker, T. (January 14, 2014), “N.S.A. Devises Radio Pathway Into Computers,” *The New York Times*.
- Santorini, B. (1990), “Part-of-Speech Tagging Guidelines for the Penn Treebank Project,” [/urlhttps://www.cis.upenn.edu/treebank/](http://www.cis.upenn.edu/treebank/).
- Schneier, B. (2014a), “IRONCHEF: NSA Exploit of the Day,” [https://www.schneier.com/blog/archives/2014/01/nsa\\_exploit\\_of\\_1.html](https://www.schneier.com/blog/archives/2014/01/nsa_exploit_of_1.html), January 3, 2014.
- Schneier, B. (2014b), “There’s No Real Difference Between Online Espionage and Online Attack,” *The Atlantic*, March 6, 2014.

- Schneier, B. (2015a), “Hacker or spy? In today’s cyberattacks, finding the culprit is a troubling puzzle,” Online, March 4, 2015.
- Schneier, B. (2015b), “We Still Don’t Know Who Hacked Sony,” *The Atlantic*, January 5, 2015.
- Schrodtt, P., Ulfelder, J., and Ward, M. (2016), “MADCOW: Multi-Attribute Data Collected on Web,” Online, <http://parusanalytics.com/MADCOW/index.html>.
- Schrodtt, P. A. (1998), “KEDS Kansas Event Data System Version 1.0,” <http://eventdata.parusanalytics.com/>.
- Schrodtt, P. A. (2012), “CAMEO Conflict and Mediation Event Observations: Event and Actor Codebook,” <http://data.gdeltproject.org/documentation/CAMEO.Manual.1.1b3.pdf>.
- Schrodtt, P. A. (2014), “TABARI Textual Analysis by Augmented Replacement Instructions,” .
- Schrodtt, P. A., Davis, S. G., and Weddle, J. L. (1994), “KEDS - A program for the Machine Coding of Event Data,” *Social Science Computer Review*, 12, 561.
- Smith, J. F. (2010), “Richard Clarke on Cyber Threats: Defense is Key,” [http://belfercenter.ksg.harvard.edu/publication/20347/richard\\_clarke\\_on\\_cyber\\_threats.html](http://belfercenter.ksg.harvard.edu/publication/20347/richard_clarke_on_cyber_threats.html).
- The Economist (December 8, 2012), “Hype and fear: America is leading the way in developing doctrines for cyber-warfare. Other countries may follow, but the value of offensive capabilities is overrated.” Online.
- The World Bank (2016), “World Deveopment Indicators,” Online, [data.worldbank.org](http://data.worldbank.org).
- Themner, L. (2015), “UCDP/PRIO Armed Conflict Dataset Codebook,” 4.
- Trask, A., Michalak, P., and Liu, J. (2016), “Sense2Vec - A Fast and Accurate Method for Word Sense Disambiguation in Neural Word Embeddings,” *ICLR 2016*, <http://arxiv.org/pdf/1511.06388v1.pdf>.
- Tripwire Guest Authors (2014), “Executive Cyber Intelligence Report: September 1, 2014,” Online, <http://www.tripwire.com/state-of-security/government/executive-cyber-intelligence-report-september-1-2014/>.
- United Nations Office on Drugs and Crime (2016), “UNODC Statistics,” Online, [data.unodc.org](http://data.unodc.org).
- Valeriano, B. and Maness, R. C. (2014), “The dynamics of Cyber Conflict Between Rival Antagonists, 2001-11,” *Journal of Peace Research*.

- Volz, D. and Finkle, J. (March 25, 2016), “U.S. indicts Iranians for hacking dozens of banks, New York dam,” *Reuters*, <http://www.reuters.com/article/us-usa-iran-cyber-idUSKCNOWQ1JF>.
- Ward, M. D., Beger, A., Cutler, J., Dickenson, M., Dorff, C., and Radford, B. (2013), “Comparing GDELT and ICEWS Event Data,” [http://mdwardlab.com/sites/default/files/GDELTICEWS\\_0.pdf](http://mdwardlab.com/sites/default/files/GDELTICEWS_0.pdf).
- Weidmann, N. B. (2015), “A Closer Look at Reporting Bias in Conflict Event Data,” *American Journal of Political Science*, 60, 206–218.
- Weidmann, N. B., Kuse, D., and Gleditsch, K. S. (2010), “The Geography of the International System: The CShapes Dataset,” *International Interactions*, 36.
- Weiss, G. W. (2007), “The Farewell Dossier: Duping the Soviets,” <https://www.cia.gov/library/center-for-the-study-of-intelligence/csi-publications/csi-studies/studies/96unclass/farewell.htm>.
- Zhu, M., Zhang, Y., Chen, W., Zhang, M., and Zhu, J. (2013), “Fast and Accurate Shift-Reduce Constituent Parsing,” *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pp. 434–443.

# Biography

Benjamin James Radford was born in Iowa City, Iowa on September 11, 1986. He earned his bachelor's degree from the University of North Carolina Asheville where he studied mathematics and international relations. He graduated summa cum laude in 2010. Benjamin received his master's and doctoral degrees in political science from Duke University. In addition to cybersecurity and political methodology, he has also published research on the economic impact of civil war (Minhas and Radford, 2016).

Benjamin has worked previously on behalf of Caerus Associates for a defensive cybersecurity research project at the Defense Advanced Research Projects Agency (DARPA). He is currently a Principal Data Scientist with Sotera Defense Solutions.