AFRL-RI-RS-TR-2022-112

DATA DISCOVERY AND COLLECTION IN SUPPORT OF DATA ANALYTICS

HYPERION GRAY

JULY 2022

FINAL TECHNICAL REPORT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

STINFO COPY

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE

■ **AIR FORCE MATERIEL COMMAND** ■ **UNITED STATES AIR FORCE** ■ **ROME, NY 13441**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09. This report is available to the general public, including foreign nations. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RI-RS-TR-2022-112 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE CHIEF ENGINEER:

/ S /

MICHAEL J. MANNO
Work Unit Manager

/ S /

SCOTT D. PATRICK
Deputy Chief
Intelligence Systems Division
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED	
		START DATE	END DATE
JULY 2022	FINAL TECHNICAL REPORT	FEBRUARY 2019	FEBRUARY 2022

4. TITLE AND SUBTITLE

DATA DISCOVERY AND COLLECTION IN SUPPORT OF DATA ANALYTICS

5a. CONTRACT NUMBER	5b. GRANT NUMBER	5c. PROGRAM ELEMENT NUMBER
FA8750-19-2-0013	N/A	62702E
5d. PROJECT NUMBER	**5e. TASK NUMBER**	**5f. WORK UNIT NUMBER**
		R2QY

6. AUTHOR(S)

Jason Hopper

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Hyperion Gray 5650 Fetzer Ave NW Concord NC 28027	

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)	11. SPONSOR/MONITOR'S REPORT NUMBER(S)
Air Force Research Laboratory/RIED 525 Brooks Road Rome NY 13441-4505	AFRL/RI	AFRL-RI-RS-TR-2022-112

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for Public Release; Distribution Unlimited. This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The goal of this effort was to support the D3M domain discovery systems, to achieve acceptable recall rates against ground truth datasets in program evaluations, and deliver an easily re-trainable, model-agnostic data discovery, collection, and extraction system that could be centrally provided and leveraged across multiple programs. This required addressing a number of challenges inherited from the underlying web crawling technologies such as reliable handling of dynamic content, anti-bot mechanisms such as CAPTCHA puzzles, and other annoyances like soft 404 errors, parked domains, and page loading delays.

15. SUBJECT TERMS

Recall rates, data discovery, data collection, data extraction

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES
a. REPORT	b. ABSTRACT	C. THIS PAGE		
U	U	U	SAR	16

19a. NAME OF RESPONSIBLE PERSON	19b. PHONE NUMBER *(Include area code)*
MICHAEL J. MANNO	**N/A**

TABLE OF CONTENTS

List of Figures i
List of Tables i
1.0 SUMMARY 1
2.0 INTRODUCTION 2
3.0 METHODS, ASSUMPTIONS, AND PROCEDURES 3
4.0 RESULTS AND DISCUSSION 6
5.0 CONCLUSIONS 11
LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS 12

LIST OF FIGURES

Figure 1: Architecture diagram of the implemented Data augmentation system……... 4
Figure 2: Logic flow of Data collection system…….. 5

LIST OF TABLES

Table 1: Systems and repositories……8

1.0 SUMMARY

Under the D3M program, our team developed a series of Domain Discovery (DD), Data Collection and Extraction tools, based on technologies that we developed under the DARPA Memex program.

At the start of the D3M program, state-of-the-art domain discovery systems still suffered from a number of challenges, many of which we encountered during our work on the Memex program. For example, none of the systems developed under Memex were able to consistently achieve acceptable recall rates against ground truth datasets in program evaluations. While precision scores fare slightly better, there was still significant room for improvement, which required addressing a number of challenges inherited from the underlying web crawling technologies such as reliable handling of dynamic content, anti-bot mechanisms such as CAPTCHA puzzles, and other annoyances like soft 404 errors, parked domains, and page loading delays.

The goal of our proposed work under D3M was to combine and extend a set of existing capabilities to provide an easily re-trainable, model-agnostic data discovery, collection, and extraction system that could be centrally provided and leveraged across multiple programs.

The plan was to integrate our technologies into the DataMart systems being developed by at least two other teams on the D3M program. The DataMart indexed domain-specific datasets, curated by domain discovery crawlers, and ingested to the index through sophisticated ETL pipelines that extracted metadata and identified potential joins and unions between disparate datasets, both within and across domains. Based on our experience under the Memex program and existing suite of tools, we proposed to build the backend discovery crawlers that the DataMart systems could leverage to populate their indexes. However, due to a year-long contracting delay, we joined the program a year after it kicked off, and so the DataMart teams were well into their research, and we struggled to find opportunities for integration and collaboration. The integration would have required additional, sometimes retroactive, work on the part of the DataMart teams, and while everyone had the best collaborative intentions, ultimately this just proved technically infeasible. We *did* accomplish proof-of-concept integration of our dataset discovery system with both the NYU and ISI Datamart systems, but we didn't move forward beyond end-to-end testing.

Nonetheless, we continued to focus on building discrete domain discovery tools and utilities that could be used by the DataMart systems, other performers, or potential transition partners. We continued to work with DARPA to identify gaps in the program capability portfolio that we could help address, and we continued to look for opportunities to respond to new use cases and challenge problems in the realm of domain discovery and dataset ETL.

2.0 INTRODUCTION

State-of-the-art of domain discovery (DD) systems suffer from several challenges, including those inherited from the underlying web crawling technologies such as reliable handling of dynamic content, anti-bot mechanisms such as CAPTCHA puzzles, and other annoyances like soft 404 errors, parked domains, and page loading delays. But even holding those constant, the task of domain discovery has its own limitations, such as achieving acceptable recall rates against subject matter expert-curated truth data sets. While precision scores fare slightly better in performance tests, there is still a lot of room for improvement here as well.

There is a distinct separation between data collection and data analytics capabilities, and most research teams prefer to focus on one or the other, because they are both equally labor-intensive. Our qualitative goal was to blur this line a bit by streamlining the data collection process and make it more feasible for the analytics-focused teams. Our quantitative goal was to enable the rapid generation, to include complementing and "filling out" existing data sets, and to reduce the time required to do this by a factor of 10, 20 or even fifty in some cases, from weeks and months to days and hours. Our approach to achieve this relies on a simple interactive user-in-the-loop interface for bootstrapping and tuning the data discovery and collection capability.

3.0 METHODS, ASSUMPTIONS, AND PROCEDURES

Our high-level technical plan was to build and deliver a Reinforced Learning-Based Domain Discovery System in a multi-phase approach. The aim of Phase 1 was to deploy the basic Reinforced Learning-Based Domain Discovery System in a centralized, accessible location. Next was the extension of the system to include extensible post-processing ETL operations over crawl data. That kind of system supported basic data processing operations natively (e.g., basic regex extraction over a page, allow page rendering using a headless browser, etc.) and provided a strong, intuitive API to retrieve data for additional analytics. The latter was likely to be in the form of a JSON or BSON-based API that is simple, flexible, and appropriate for all types of analytics. In our varied experience as data providers, we learned that it is important to support multiple ways of accessing data – some use cases require "data streaming" capabilities, while others require "data dump" capabilities, for example, while still others require query-based retrieval capabilities.

In Phase 2 the goal was to expand the feature set to support richer data annotations. We were not yet asking why results are on topic or not, but only asking whether they were or were not relevant. This limited machine learning inputs to an inferred why, which in many cases works fine but in other, more nuanced, cases it does not. Data discovery is highly nuanced, there are many dimensions within each topic, or domain, and often there is a fine line between what is relevant and what is not. Our solution to this issue involved a series of guided questions designed to prompt the user for more information about the quality of / problems with the results they are getting. These questions allow us to build out additional, explicit features that may perform better than implicit features in some cases. Deciding the right questions to ask was not a trivial task, so to start with we proposed a pencil-and-paper experiment, wherein we take several human-curated, domain-relevant datasets across a variety of domains and asking domain experts to circle, highlight, comment, and otherwise provide freeform feedback on why pages are relevant to their topic. This gave us a baseline for the volume and variety of explicit features that our system needed to support for any arbitrary data discovery task.

In addition, we planned to support more advanced ETL operations and more complex crawler policies / actions – as we expanded the complexity of the definition of "relevant" we also needed to expand our crawlers to support this complexity to efficiently find and choose optical crawl-paths to relevant content.

Our Phase 3 plans involved the expansion of the feature set to support other-than-text-based relevancy models. While much of our data discovery work to date used text-based relevancy

models, there were plenty of use cases for other-than-text models, such as image-based, style-based, structure-based, or even source code-based (technically a subset of text-based).

The diagram below portrays the proposed architecture and workflow of the D3M system we envisioned, which we ultimately ended up implementing most of.

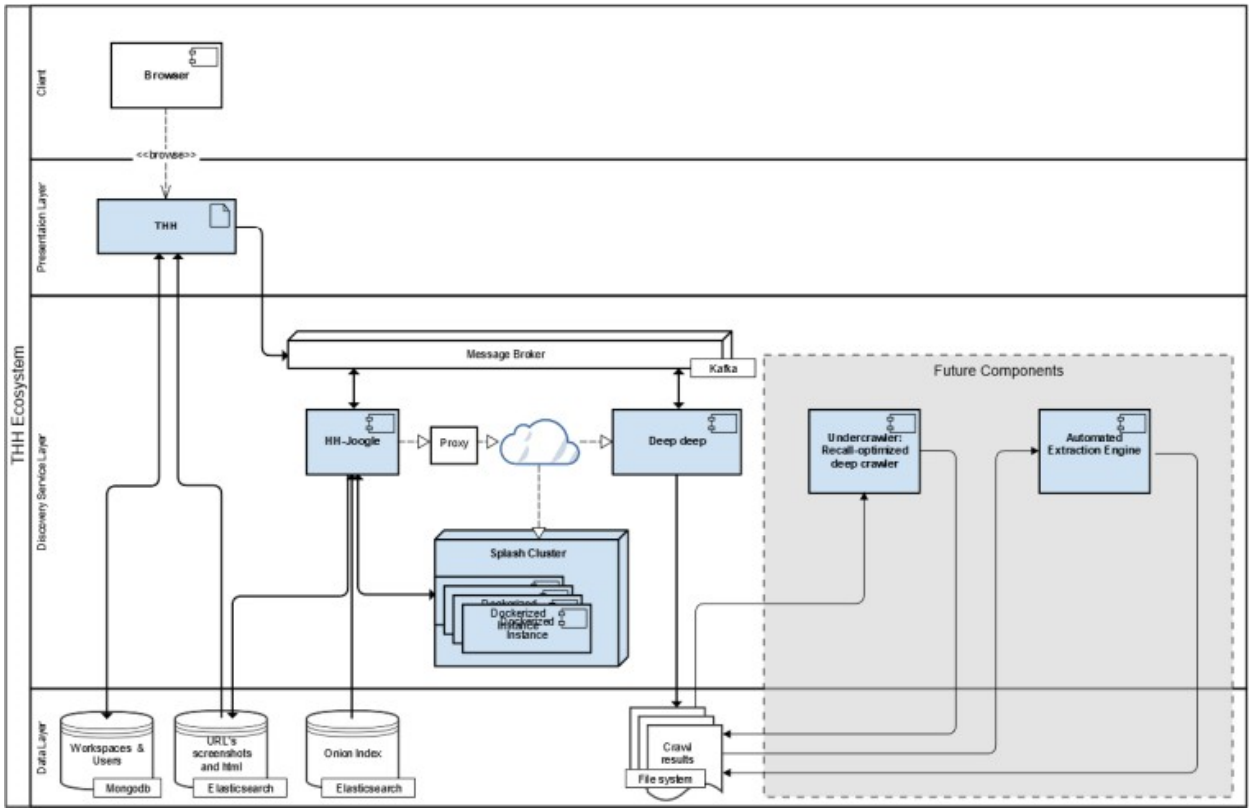

Figure 1: Architecture diagram of the implemented Data augmentation system

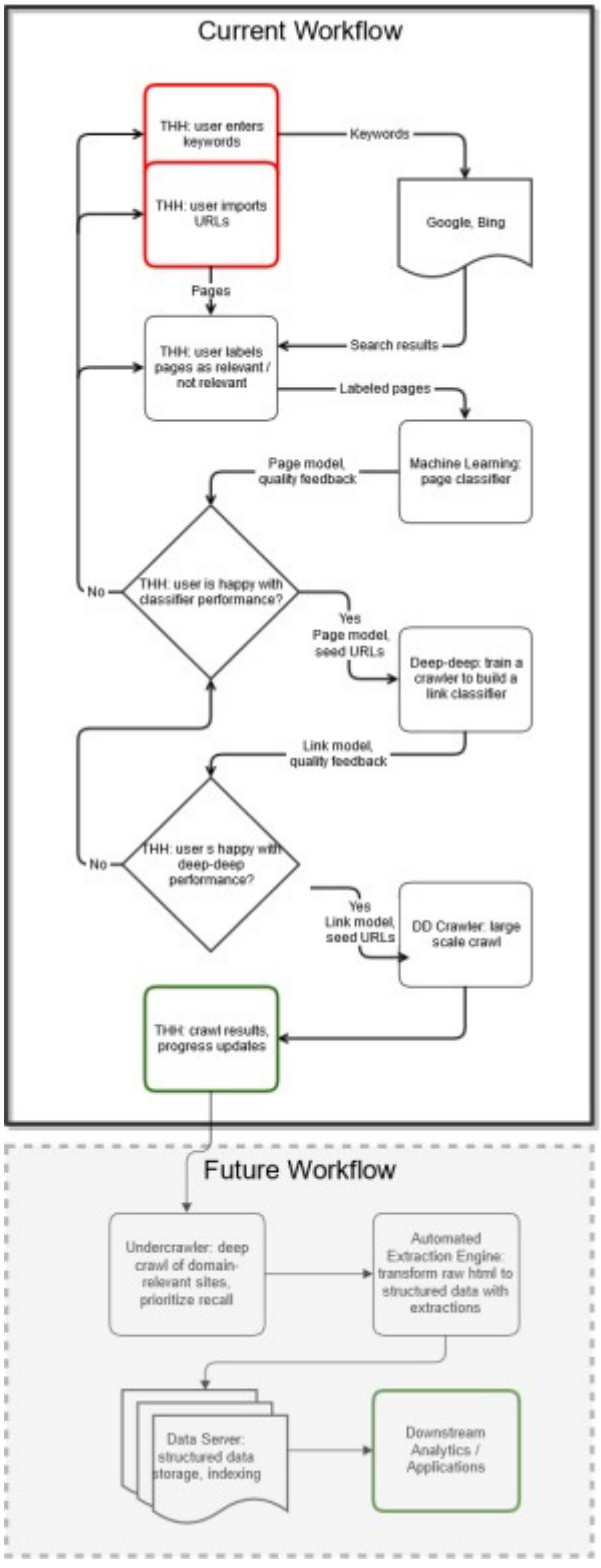

Figure 2: Logic flow of Data collection system

4.0 RESULTS AND DISCUSSION

To start our performance under D3M (again, about a year after program kick off), we developed a small proof-of-concept dataset discovery engine based on our previous work under the Memex program. The goal of the POC was to try and curate *domain specific datasets* for the DataMart systems, based on a user query. We engineered a small set of headless browser systems which took a user-provided keyword, passed it to a Google search or a kaggl.com search (Kaggl is a machine learning competition website that contains many searchable datasets), and then crawled and downloaded datasets associated with the query results. This was the "lowest hanging fruit" that provided a starting point for setting up our basic dataset discovery framework.

Next, we worked to improve the proof-of-concept dataset discovery engine in collaboration with the our fellow performers in the DataMart Working Group. As noted, their systems had been under development for the past year, so we needed to try to retrofit the expected inputs and outputs and points of integration with our system. In discussions with our colleagues, we decided that our best fit in the larger program was to build an extensible headless browser framework that would allow us to access datasets from any arbitrary dataset index or from the web more generally, and to extracted some metadata that the other teams' dataset profiling services were not covering.

Next, we focused on expanding the number of sites we crawled from, beyond just Google and Kaggle. Previously, each new data source required a custom entity or field extractor, which was not sufficiently scalable to support our goal to discover as many data sources as possible. Thus, we began working on a more abstracted, generalized extractor framework that could manage a wider variety of sources and with better efficiency.

Our initial approach used a computer vision-based approach, where a visual representation of a webpage was parsed by a CV model using semantic login. However, after evaluating this implementation, we determined that this approach did not generalize well enough for our needs. We pivoted to an innovative approach in hopes that it would yield a truly generic extractor using a deep neural network architecture. This involved developing a labeled dataset, which we generated by performing headless crawls of millions of webpages and then leveraging html markup that references schema.org representations as "free" labels, which yielded us a dataset of twenty-four million labeled records. This dataset was being used to develop a deep neural network capable of performing structured extraction from generic websites. The biggest technical change was to move away from the computer vision-based approach, which was promising and worth exploring, but ultimately not the best use of our time.

Next, we wanted to incorporate an open-source data cleaning service / library into our pipeline. We first investigated a feature rich ETL and data cleaning tool called Dataiku

(https://www.dataiku.com/product/features/data-preparation/). We reviewed the available code and discovered that there are thousands of underlying open source and proprietary libraries. Using an API, we could use a tool like this as a plugin to perform data prep and infer file schema on discovered datasets.

We also spent time improving dataset handling from Google Dataset Search. We triaged several issues that were preventing higher recall and precision on the GDS results. Specifically, we improved our crawler to manage more "hops" to detect a dataset, improved our document type classifier (csv vs excel vs database, etc.), and identifying datasets that require a login.

When the COVID-19 pandemic broke out in early 2020, the program sought to apply the D3M technologies to this new and pressing challenge. We proposed an idea to start collecting data from online marketplaces Amazon.com and Ebay as an indicators of items that were becoming essential or unusually high demand as a result of the pandemic, such as toilet paper, gloves, masks, and even hair dye. Using our existing headless browser cluster, we began collecting the daily Amazon Movers & Shakers list, a list of products that have made the biggest shifts in demand (i.e., sales rank) in the previous 24 hours. Our idea was that this could be used as an early indicator of what products were becoming the next "pandemic panic buy" items. We began collecting on the US Amazon domain and eventually added the domains for Italy and Spain, and Canada countries which at the time were leading the world in COVID-19 cases. We also identified new websites related to COVID-19 as they were deployed, and developed models to predict if these websites were malicious (eg: phishing). We continued this collection for months and made the data available via a public website, covid.hyperiongray.com, for performers or the broader research community. However, aside from conducting some of our own statistical analyses, this mini project didn't get broader traction, perhaps because the community was focused more on case loads and death rates at the time. However, we still believe this data could yield some interesting insights into how consumers responded to the economic effects of the pandemic.

For the final months of our period of performance of this program we completed our Google Dataset crawler, added a new Reddit /r/datasets crawler to our pipeline, and conducted a self-evaluation experiment on recall rates for the GDS and Reddit collectors. We discovered that the subreddit /r/datasets contained a lot of datasets, particularly covid related datasets that are distinct from those found in Google Dataset Search.

As a final challenge problem, we worked with the other DataMart teams to apply our dataset discovery system to the task of collecting novel, relevant datasets specific to Ethiopian food security, as part of a collaboration with the World Modelers program.

At the end of our performance period, we documented, archived, and spun down our system and the various components, and spent the remainder of the program supporting transition work as applicable. For example, we built a system to support other D3M groups by crawling specific news sites requested by transition groups and translating this data into a standard D3M format which was then ingested by the DataMarts systems.

Table 1: System and repositories.

Tool / Utility	Repository	Description
Nina News crawler	https://gitlab.com/hg-d3m	Webcrawler for Nina
	/d3m-ninanews-crawler	news site
Autoextract Deploy	https://gitlab.com/hg-d3m /d3m-autoextract-deploy	Deployment code for Autoextract service
Syriahr News Crawler	https://gitlab.com/hg-d3m /d3m-syriahr-crawler	Webcrawler for Syriahr news site
Global Incident map crawler	https://gitlab.com/hg-d3m /d3m-globalincidentmap-crawler	Webcrawler for the Global Incident map
Articles backen	https://gitlab.com/hg-d3m /d3m-articles-backend	System for organizing news articles for augmented data
Crawler node	https://gitlab.com/hg-d3m /d3m-crawler-node	Node for running news site crawlers
Reddit node	https://gitlab.com/hg-d3m /d3m-reddit-node	Crawler node for Reddit incident data
Reddit Crawler	https://gitlab.com/hg-d3m /d3m-reddit-crawler	Reddit crawler for Reddit COVID data
Convert2CSV	https://gitlab.com/hg-d3m /convert2csv	Tool for converting Datamart data to CSV files
CSV Insights Core	https://gitlab.com/hg-d3m /d3m-csv-insights-core	System for processing and extracting insights from Datamart data
Analytics Dataiku	https://gitlab.com/hg-d3m /d3m-analytics-dataiku	Analytics pipeline using Dataiku
Status	https://gitlab.com/hg-d3m /d3m-status	Status utility for internal systems
Datamart Integration	https://gitlab.com/hg-d3m /hg-datamart-integration	Compatability layer/API for interacting with D3M datamarts

Headless Amazon Collector Client	https://gitlab.com/hg-d3m/headless-amazon-collector-client	Headless crawler for collecting information on the most sold products on Amazon
Headless Amazon Collector Client	https://gitlab.com/hg-d3m/headless-amazon-collector-client	System for processing raw crawl data from the Headless Amazon Collector Client
EKS cluster config	https://gitlab.com/hg-d3m/d3m-eks-cluster	Kubernetes configuration system
Datamart UI	https://gitlab.com/hg-d3m/d3m-datamart-ui	User interface for the Hyperion Gray datamart
COVID19 data	https://gitlab.com/hg-d3m/covid19-data	System of tools for collecting and monitoring COVID websites
Xtract	https://gitlab.com/hg-d3m/xtract	Deep Learning system and model to extract labelled product information from webpages
Analytics	https://gitlab.com/hg-d3m/d3m-analytics	Analytics engine for collected Datamart data
Ebay Crawler	https://gitlab.com/hg-d3m/d3m-ebay-crawler	Service for crawling Ebay and providing augmented data
Schema Parser Stream	https://gitlab.com/hg-d3m/schema-parser-stream	Infer data schema from streamed data from crawlers

Crawler Core	https://gitlab.com/hg-d3m/d3m-crawler-core	Google Datasets core system
Dataset Crawler	https://gitlab.com/hg-d3m/dataset-crawler	Crawling system for Google Datasets

5.0 CONCLUSIONS

While we were able to explore and address the challenges of domain discovery through our work on this problem, and while we built many novel capabilities and practical technical solutions, ultimately our ability to fully integrate our work into the broader program was hampered by our late start. Rather than being able to position our research into the original program architecture, we had to force fit ourselves into it after the fact, and we never quite found our fit. This was less than ideal, of course, but ultimately we were still able to adapt our research plans, build newly defined capabilities that addressed niche challenges of data discovery and collection, and we supported our fellow performers and the program goals wherever we could.

LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS

CDS	Cross Domain System or Solution
DD	Domain Discovery
DARPA	Defense Advanced Research Projects
I2O	Information Innovation Office
SME	Subject Matter Expert
POC	Proof of Concept
THH	The Headless Horseman
ELI5	Explain it like I'm Five
UI	User Interface
NER	Named Entity Recognition
IARPA	Intelligence Advanced Research Projects Activity
CAUSE	Cyber-Attack Automated Unconventional Sensor Environment
D3M	Data Driven Discovery of Models
UX	User Experience
NYU	New York University
GDS	Graphic Display System
HTML	Hypertext Markup Language
ETL	Extract Transform Load ISI Information Sciences Institute
API	Application Programming Interface
WG	Working Group
AWS	Amazon Web Services
JSON	JavaScript Object Notation
BSON	Binary Javascript Object Notation