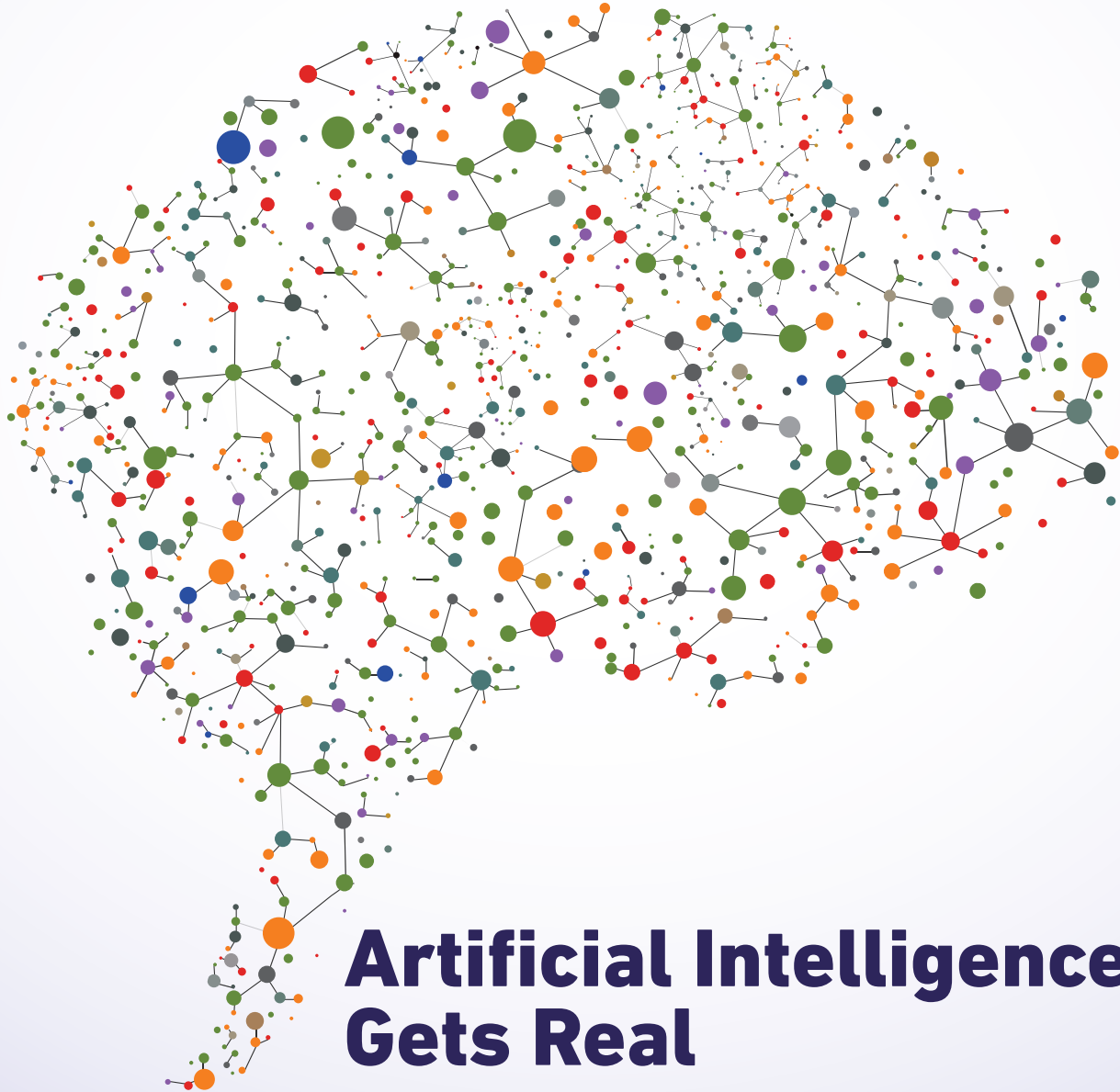


IQT

QUARTERLY

VOL. 7 NO. 2

FALL 2015



Artificial Intelligence Gets Real

IQT
IN-Q-TEL

IQT Quarterly is a publication of In-Q-Tel, Inc., the strategic investment organization that serves as a bridge between the U.S. Intelligence Community and venture-backed startup firms on the leading edge of technological innovation. *IQT Quarterly* advances the situational awareness component of the IQT mission, serving as a platform to debut, discuss, and debate issues of innovation in the areas of overlap between commercial potential and U.S. Intelligence Community needs. For comments or questions regarding IQT or this document, please visit www.iqt.org, write to iqtquarterly@iqt.org, or call 703-248-3000. The views expressed are those of the authors in their personal capacities and do not necessarily reflect the opinion of IQT, their employers, or the Government.

©2015 In-Q-Tel, Inc. This document was prepared by In-Q-Tel, Inc., with Government funding (U.S. Government Contract No. 2014-14031000011). The Government has Government Purpose License Rights in this document. Subject to those rights, the reproduction, display, or distribution of the *IQT Quarterly* without prior written consent from IQT is prohibited.

EDITORIAL

IQT Quarterly, published by In-Q-Tel, Inc.

Editor-in-Chief: Adam Dove

Theme Editor: Sri Chandrasekar

Contributing Editors: Carrie Sessine, Brittany Carambio, and Melissa Hayes

Design by Lomangino Studio LLC

Printed in the United States of America

TABLE OF CONTENTS

On Our Radar: Artificial Intelligence Gets Real 02
By Sri Chandrasekar

A Look Inside the Issue 05

Deep Learning, Big Data, and Problems with Scale 06
By Naveen Rao

Predictions with Big Data 11
By Devavrat Shah

**AI Roundtable: Intelligence
from Lab41's Technical Advisory Board** 15
A Q&A with Steve Bowsher, Jeff Dickerson, and Josh Wills

**AI for the Analyst: Behavioral
Modeling and Narrative Processing** 19
By Adam W. Meade and R. Michael Young

**DeepDive: Enabling Next-Generation Business
Intelligence with Information Extraction** 23
By Michael Cafarella

**Can AI Make AI More Compliant?
Legal Data Analysis *Ex Ante*, *In Situ*, *Ex Post*** 28
By Bob Gleichauf and Joshua H. Walker

From the Portfolio 33

ON OUR
RADARIQT
IN-Q-TEL

Artificial Intelligence Gets Real

By Sri Chandrasekar

I was introduced to the concept of artificial intelligence (AI) when I watched *Star Wars: The Empire Strikes Back* as a 6-year-old. I distinctly remember being in awe of C3PO and his fluency in more than “6 million forms of communication.” And while in retrospect, I probably should have been more impressed with R2D2 (especially since he is the real hero of the *Star Wars* saga), it was the fact that C3PO could converse with beings using real speech and walked like a human that made me remember him.

The seminal AI textbooks (Russell & Norvig, Luger & Stubblefield, etc.) propose that the central problems in AI include reasoning, knowledge, planning, learning, natural language processing (NLP), and perception. That is a daunting list of problems for a single field to solve. But C3PO exhibits all of these abilities in spades. During his time in Cloud City, he is able to reason that Chewbacca soldered him together backwards. He has the knowledge to warn R2D2 from “talking to strange computers,” and to be afraid of blaster fire when running to the Millennium Falcon. He learns that R2D2 is far more capable than himself and is able to fix the Millennium Falcon Hyperdrive. His natural language processing skills are evident throughout the movie and his continued reminders of those skills irritates Han Solo (who nonetheless is unafraid of using C3PO’s skills when he orders Chewie to “plug him into the Hyperdrive”). And of course, he is able to perceive — he has vision, hearing, and the ability to feel.

When I used to look at the list of AI challenges, I was always reminded how far away we were from creating a C3PO. But the last two years have changed my views. Instead of being distraught about how far away we are from true AI, I’ve been heartened by the amount of

progress taking place in each of AI’s core problems. AI technologies surround us on a daily basis. Whether it is asking Siri for the current weather (perception, NLP, knowledge), Google Now telling you that you should leave in 15 minutes to make your flight (reasoning, knowledge, planning, NLP), or Facebook identifying an image’s subjects as your wife and cousin (reasoning, perception), AI technologies have become an integral part of our lives. Even more exciting is the fact that the technologies being applied in these cases are approachable and understandable to those with the interest and desire to learn about them. As a member of Lab41 and the broader In-Q-Tel organization, I get a front-row seat to this AI innovation. Through this article, I hope to share some of today’s most innovative AI technologies.

Perception and Reasoning: Interpretation of sensory information and consciously verifying logic

These are the quintessential tasks that most people think about when they are asked to explain AI. There is no cooler example of perception and reasoning by AI than Google’s self-driving cars. Living in Silicon Valley,

I see these cars driving around about once a week. Each time, I jealously regard the “driver” and “passengers” in the cars as I zoom by (the cars appear to always drive at the speed limit — a failing that I note with happiness each time I pass them). Without exception, self-driving cars are the technology that will be available to the masses in the next 10 years that I’m most excited about.

A video of a Google self-driving car shows the advanced reasoning and perception capabilities possible with today’s AI.¹ There is a truck parked on the right side of the right lane, but there are also cars moving in the left lane. The AI reasons that there isn’t enough room in the right lane to pass the truck without cutting off the driver in the left lane. It slows down enough to let the left lane car pass, and then seamlessly changes to the right lane. This is the kind of scenario that even experienced drivers have difficulty navigating in a seamless manner — and the AI has learned this behavior. There is no slamming the brakes, or panicked lane changing. The riders are treated to a very smooth handling of the situation.

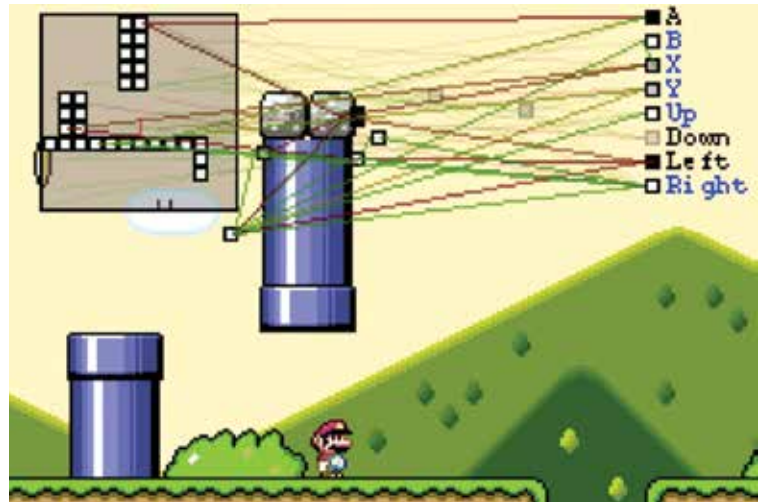
Learning and Planning: Acquire new knowledge and realize strategies

Marl/O is a neural network that learns how to play Super Mario World by trial and error.² When I saw this, it got me excited all over again about video games. I remember learning to play Super Mario World — and watching an AI learn to play it (albeit, slower than me at the start) was truly interesting.

Seth Bling, the author of this AI, leveraged both genetic programming and neural network techniques in order to build Marl/O. One of the things that I like most about his explanation of Marl/O is that he shows how each evolutionary step moves the AI forward. Ultimately, the AI is able to beat the initial level after 34 generations of training (it took him about 24 hours of real-life compute time to train). And while the AI’s style in beating the level isn’t to my taste (it doesn’t get the power-up mushroom and it leaves a bunch of coins behind), I’m reasonably confident that with different success parameters, the AI would learn to play Super Mario World as well as me. The most pertinent thing about this demo is that the author is not a neural networks expert, but was still able to cobble together this software in about two weeks.

Natural Language Processing and Knowledge

If you aren’t following Andrej Karpathy’s (a current Ph.D. student at Stanford) blog, do so now. His most recent blog post, *The Unreasonable Effectiveness of*



Marl/O is a neural network that learns to play Super Mario World by trial and error.

Recurrent Neural Networks, describes teaching a recurrent neural network with character by character input from a text corpus.³ What is amazing about this technique is that once trained with a sufficient amount of data, these nets can learn and possess knowledge (including syntax, proper sentence structure, and punctuation) about a language.

In this blog post, he trains the net with text from several different domains including Paul Graham (founder of Y Combinator) posts, Shakespeare, Linux source code, LaTeX documents, and others. Once the net is trained, you can have it output language by giving it a seed (a character like A) and having it provide a decision on the next letter in the sequence. By doing this continuously, the net can create synthetic documents that look real. For example, here is some sample Shakespearean English that his recurrent net wrote from scratch:

PANDARUS:

Alas, I think he shall be come approached and the day
When little strain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:

They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:

Well, your wit is in the care of side and that.

Second Lord:

They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:

Come, sir, I will make did behold your worship.


VIOLA:

I'll drink it.

Not bad. It doesn't quite make sense, but you had to read it a couple of times to make sure that it doesn't make sense, right? You aren't really sure while reading it whether the author made a mistake or your understanding of Shakespearean English is broken. I included the Shakespeare example here, but I was most impressed by the properly formatted Latex documents that his net was able to create — complete with random, useless mathematical formulae.

At Lab41, we're exploring many of the technologies that are behind the innovations I've described. The Lab's core hypothesis is that automation is a key to the Intelligence Community's continued success. This is particularly true in the big data arena as the volume of data presented to analysts and data scientists continues to grow at exponential rates. AI technologies will underpin our desire to automate various parts of the IC workflow. The Lab is exploring perception and reasoning in the form of

machine vision and image processing challenges. Our work in learning and planning is focused primarily on extracting useful information from user interaction data. And finally, we're exploring the intersection between natural language processing and deep learning. All of Lab41's work is open source and posted on GitHub (lab41.github.io).

I hope I've been able to share with you some of the reasons for the current hype surrounding AI. The continuing exponential increase in computational capacity has powered a number of the innovative technologies I've introduced. Perhaps just as important are the lowered barriers to learning about and adopting these technologies. You and I are probably not going to create our own self-driving car AI, but even I was able to train a recurrent net to write part of this article using writing samples from previous *IQT Quarterly* articles. 

Sri Chandrasekar is a Vice President at In-Q-Tel and the Deputy Director of Lab41. Chandrasekar is responsible for setting the challenge agenda and ensuring execution of the Lab's guiding vision. Prior to joining the Lab, he was a Member of IQT's Technical Staff, where he led investments in communications technologies and mobile security companies before ultimately leading the advanced analytics theme. Previously, Chandrasekar was a Systems Engineer at BAE Systems where he designed communications systems. He has an M.B.A. from the New York University Stern School of Business, as well as a M.Eng. and B.Sc. from Cornell University in Electrical and Computer Engineering.

REFERENCES

¹ <https://www.youtube.com/watch?v=bD0nn0-4Nq8>

² <https://www.youtube.com/watch?t=62&v=qv6UVOQ0F44>

³ <http://karpathy.github.io/2015/05/21/rnn-effectiveness/>

A Look Inside the Issue



This issue of the *IQT Quarterly* examines recent advances in artificial intelligence — the field of computer science that involves equipping machines with human-like intelligence. Recent breakthroughs in deep learning and data science are creating increasingly mature AI technologies, and the industry is inching closer than ever to realizing its full potential.

Naveen Rao of Nervana Systems opens the issue with a discussion on the challenges of processing large data sets. While deep learning has driven massive enhancements in AI tasks like image classification and natural language processing, barriers in scalability and usability limit the adoption of deep learning for big data. Nervana's open source deep learning framework aims to address these problems.

Devavrat Shah of Celect continues the big data conversation with his vision of enhanced decision making through meaningful data. He describes the need for an ultimate prediction engine that can consume large amounts of unstructured data and provide accurate predictions of the unknown.

Next, the *IQT Quarterly* interviews three members of Lab41's Technical Advisory Board to gather perspectives on AI from the Intelligence Community (Jeff Dickerson), the private sector (Josh Wills), and In-Q-Tel (Steve Bowsher). The trio provides insights on the latest AI hype cycle, innovative AI technologies, and the industry's future.

Adam W. Meade and R. Michael Young of North Carolina State University provide commentary on the intersection

of artificial and human intelligence. They argue that the IC should seek to use artificial intelligence to complement the role of human analysts, rather than to replace human judgement and decision making. NC State's Laboratory for Analytic Science focuses on two AI-based studies: sensemaking through storytelling and modeling analyst behavior.

Michael Cafarella of the University of Michigan discusses the importance of unlocking "dark data" — the information buried in text, tables, and images. This type of data contains important information, but is difficult for data management tools to derive meaning from because of its structure. Cafarella describes how the DeepDive project applies information extraction methods to turn dark data into useful, structured data for business intelligence.

Bob Gleichauf and Joshua H. Walker close the issue with an overview of AI's potential for compliance problems facing the IC. Automation tools, such as a data rights model that tracks the lifecycle and transformations of data, provide a framework for addressing growing legal and informational complexities.

Beyond the technologies presented in this issue, IQT and Lab41 continue to monitor the innovation taking place in artificial intelligence research, startups, and the broader tech industry. The IC's awareness of AI's potential will be critical to harnessing the potential of these technologies for national security. **Q**



Deep Learning, Big Data, and Problems With Scale

By Naveen Rao

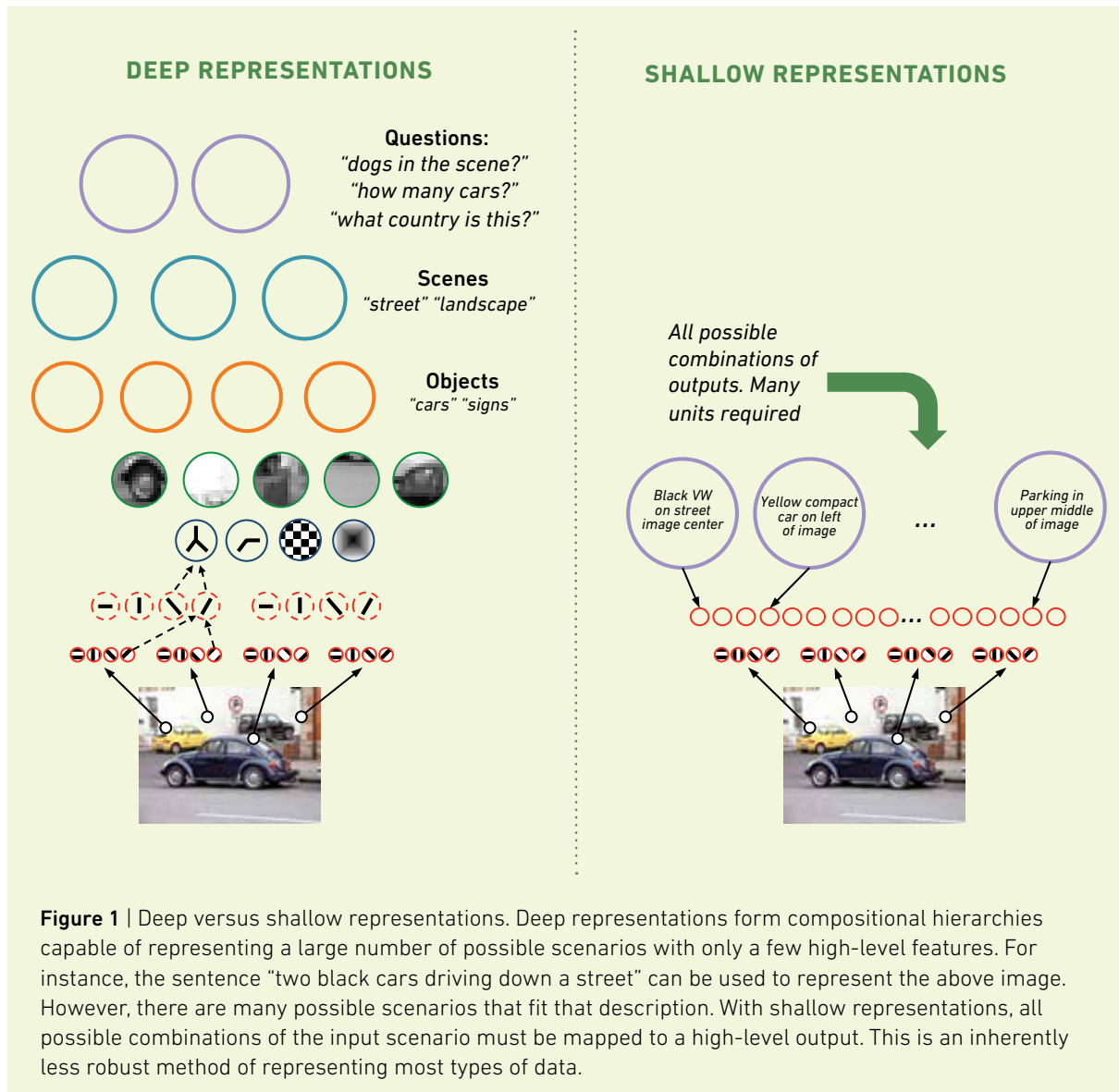
Deep learning, a form of machine learning, has enabled computers to rival human performance on tasks such as image classification, speech recognition, and natural language processing. For this reason, deep learning has been touted as the solution to big data processing, but significant hurdles remain in scaling to large data sets.

Since the discovery of the neuron as the basic unit of computation in the brain in the 1950s, mathematicians and computer scientists have been building mathematical abstractions in hopes of understanding biological learning and computation. Conventional computers have given us the tools to store and manipulate data, but these machines have to be explicitly told how to derive meaning from data. Building machines that learn from data, rather than being programmed, has remained a challenge. The latest revolution in artificial intelligence, deep learning, is changing that.

Deep learning is a form of machine learning loosely inspired by the brain. It uses simple mathematical transformations to uncover inherent structure within data, revealing meaningful assemblages of information known as features. The concept of machine learning is not new, but traditional machine learning approaches have relied on feature engineering, where human experts predetermine what combinations of information are important for interpreting a particular type of data. The power of deep learning is the ability to uncover features automatically from the data itself. What makes deep learning “deep” is the ability to assemble these discovered features into a hierarchy that defines structure within the data (see Figure 1). This is known

as compositional hierarchy, a concept that applies to nearly all types of data. For example, faces can be decomposed into noses, eyes, and mouths, and each of those can be further decomposed into finer grained components. “Shallow” learning attempts to represent all features together at a single level, which tends to be more fragile and wasteful in terms of representation. The expressivity of depth makes for not only a more compact description of data, but also a more generalizable model that is tolerant to noise and missing information.

Current deep learning architectures trace their roots to the 1980s, when researchers at the intersection of computer science, neuroscience, and statistics developed a method for learning representations from data in neural networks known as backpropagation. At the time, backpropagation was applied to the problem of handwriting recognition. This resulted in systems much more robust and error-tolerant than previous techniques and was actually widely deployed to assist in automatic address recognition for postal routing. These early neural network architectures showed much promise, but were limited by two very important factors: the lack of training data and the lack of computational power. By the early '90s, much of the community had moved away from neural network techniques,

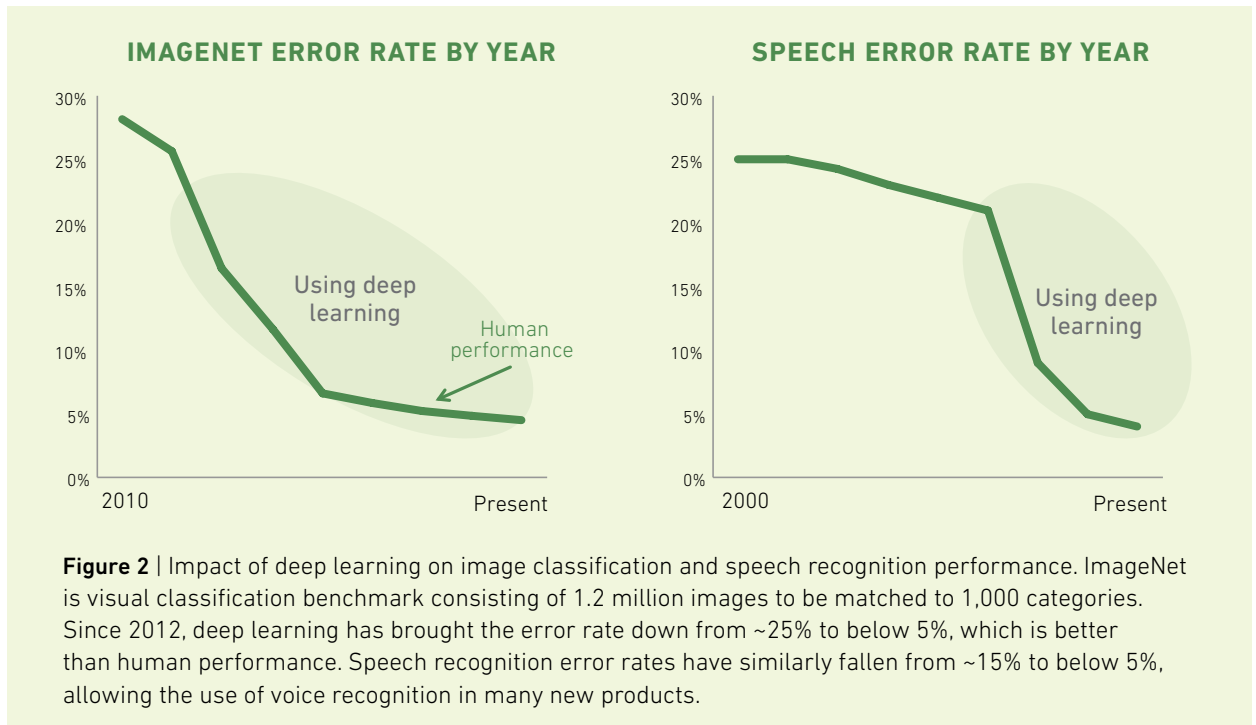


as simpler regression techniques could equal or beat neural network performance and were easier to implement on existing hardware.

As sizes of data sets grew and computational power increased, there was renewed interest in neural network techniques. The statistics of most natural data, like images and speech, tend to have "heavy tails;" there are many rare cases that become extremely important to the interpretation of the data. Simple regression techniques do not capture these subtleties, and tend to hit a performance ceiling as data sets become larger. Due to the expressivity of depth, deep learning techniques are able to better represent data and continue to improve performance by learning the heavy tails.

By the mid-2000s, image classification became a standard task for computer vision and machine

learning. To formalize a benchmark for image classification, researchers designed the ImageNet challenge. ImageNet consists of 1.2 million natural images and 1,000 labels. The task is to accurately apply those 1,000 labels to the 1.2 million images. This is quite a challenge since the content of the images must be parsed to actually apply the label accurately. To give the reader some bearings on performance (see Figure 2), an untrained human has approximately an 18 percent top-5 error rate on this task. A human who trained on the labeled images can get close to a 5 percent top-5 error rate. Feature-based computer vision algorithms seemed to hover around 25-30 percent error rate. In 2012, a group from Geoffrey Hinton's lab at the University of Toronto applied deep learning techniques to ImageNet and achieved a top-5 error rate of approximately 16 percent. This represented a huge step forward for



computer vision. The following year, deep learning techniques brought this down to nearly 11 percent, then 6 percent, and finally below 5 percent. Deep learning has now exceeded human performance on ImageNet.

Similar trends have occurred in other data modalities. The state-of-the-art in speech recognition had hovered around 15 percent error rate with only marginal improvements year over year. With the application of deep learning, recent error rates dropped below 5 percent. This performance leap is the difference between a solution that works and one that does not. Deep networks have shown promise in other areas like drug discovery, agriculture, and medicine. It's a good bet that this trend of supplanting existing machine learning methods with deep learning will continue for nearly every type of data.

Why Is Deep Learning So Important?

With the advent of the sensor-enabled, Internet-connected smartphone, the cost of gathering data has fallen to nearly zero in the last 10 years. Where previously custom equipment had to be deployed into the world to gather data, such as mapping information (Google Maps Street View car) or traffic conditions (DOT highway sensors), much of this can now be accomplished simply by writing a smartphone app. User behavior, user engagement, and customer preferences are now being measured and tracked with precision. Companies began seeing the potential in logging this information, and the hackneyed term "big data" was

born. In the last 10 years, IT spending had a compound annual growth rate of 30 percent, mostly driven by storage needs. This is true across nearly all industries: retail, pharmaceutical, agriculture, oil and gas, finance, and Internet businesses. In today's world, big data infrastructure largely equates to storage infrastructure.

A consequence of this obsessive tracking of all information surrounding a business has been a shift in the business decision process. Executives are asking, "Where is the data to support this?" rather than, "Who's responsible for this?" more than ever before. Traditionally, enormous teams of human experts were hired to make sense of data. This approach simply can't scale with growing data sizes. Companies are now interested in interpreting their very large data sets to find actionable results to improve their business. This is the area of rapid innovation and growth and where deep learning will have an enormous impact on the future. Deep learning offers the ability to automate this process and scale it to data sizes beyond the capabilities of even the largest teams of humans.

Problems in Deep Learning

So, if deep learning is the answer to our data interpretation problems, why not simply deploy software at scale to implement these algorithms? Deep learning is computationally intensive and requires specialized compute infrastructures to work at speed and scale. Current web-derived infrastructures don't actually do

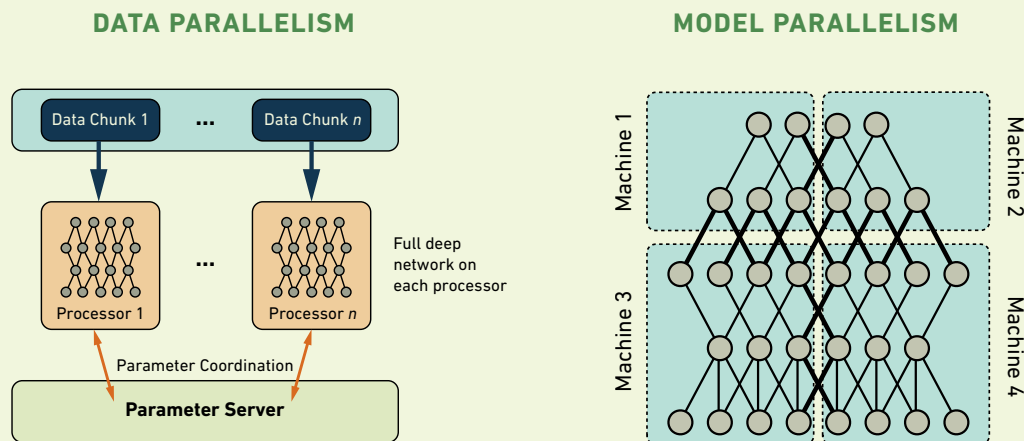


Figure 3 | Data parallelism: Consists of replicating a particular deep learning model across n compute nodes. Each of those nodes receives $1/n$ of the training data points (a data point could be a whole image). Each node trains its full model in parallel with all the other nodes, but synchronizes its trained parameters (weights) on regular intervals. During synchronization, all learned weights are transferred to a central server where they are averaged and the new weights are transmitted to all n compute nodes.

Model parallelism: The neural network model itself is broken apart and distributed across n nodes; the model and its parameters are not replicated. Each training data point is split across all n nodes such that a particular part of the model receives the portion of the data point relevant as input to it (e.g., if each data point is an image, the upper left quadrant of each image might be fed to one node). All n nodes run in relative lock step with one another and individual weights are trained separately on separate nodes.

what is required. There are no standard tools to employ. At this stage, companies hire their own teams to build specialized infrastructure with currently available hardware and custom software. This is costly and only solves a part of the problem; scale is inherently limited by the hardware.

Deep learning has some fundamentally different computational challenges. In order for a neural network to find structured features within data, it must be trained on a large corpus of data. This process can be lengthy, as training even a moderately sized model on a fast Intel CPU can take upwards of a month. In addition, model building is an iterative process where many possible model configurations must be evaluated against the training data. Faster model iteration means faster time to solutions. One saving grace for deep learning speed is that it is inherently parallelizable. Given the constraints of current computing technologies, researchers have defined ways to distribute deep learning computations to achieve acceleration across multiple compute nodes. There are two main classes of parallelism in deep networks: model and data parallelism (see Figure 3).

Data parallelism is generally more straightforward to implement on current compute infrastructures.

However, data parallelism has two main disadvantages: slower model convergence and limited model size.

Model convergence is slowed due to the many averaging operations during training, which results in a loss of learning signal. Model size limitations are due to memory limits per node. Since the entire model must be replicated on each node, the model must fit into the memory of each node. Model parallelism mitigates these disadvantages by distributing the actual model and its parameters across multiple compute nodes. True model parallelism, however, is difficult to accomplish due to I/O constraints of existing hardware. Most real-world solutions consist of some mix of these two approaches. The constraints of I/O, memory, and model convergence times are the main barriers to scaling deep learning to larger data sizes.

We Need a Common Set of Scalable Tools

Data scientists and deep learning researchers are still a feral breed of developer. They have deep roots in academia, publish papers, and discuss their work with their community. Like most academics, they tend to "roll their own" when it comes to software infrastructure. They like to understand and control all aspects of the

software they use, and usually build upon packages like Matlab or Python. For this reason, there does not exist a standardized tool chain for building and scaling deep learning solutions. This is in stark contrast to the web development world, where there is an entire ecosystem of tools to build scalable solutions. When deep learning researchers have to build solutions to real problems in the enterprise world, they generally cobble together some unsupported open source tools as a starting point. However, most of these open source tools aren't up to snuff for the enterprise world, so deep learning teams have to heavily modify the tools for their own use. This process occurs at nearly every large firm currently employing deep learning as a solution. Homemade tools are not portable from one place to another, nor do they best solve general performance and scalability issues; they work well enough for the problem the researchers have to solve. This is an inefficient use of resources and limits innovation.

The Scalable Deep Learning Solution

Nervana believes that with the right set of tools, a small team can do the work of much larger data science organizations. There is distinct expertise in building a fast, scalable deep learning solution. By focusing specifically on that problem, we can provide a much higher level of innovation than individual companies building their own infrastructure. This is inherently a more scalable approach and is far more efficient than each organization replicating the effort.


Nervana's tools are easy to use and can be standardized across many different types of data problems. Our cloud stack gives users access to state-of-the-art infrastructure for both performance and scale. Due to the unique nature of deep learning computation, even the best existing hardware solution, the graphics processing unit (GPU), scales very poorly beyond eight

processors. This historically meant that researchers needed to be creative in the application of data and model parallelism to achieve any kind of meaningful scaling to very large data sets. Nervana's goal is to remove these roadblocks and achieve near perfect scaling for model parallelism with its hardware and software solution. Nervana went truly full-stack by optimizing the infrastructure from the deep network definition all the way down to silicon. This new processor and software will not only enable much faster training on existing models, but also enable new types of model explorations. These innovations enable:

True model parallelism: Many of the problems with current compute infrastructures are related to the I/O limitations of the hardware. Nervana is building a new kind of processor for deep learning that has tremendous I/O and compute capabilities tailored to deep learning. This will enable true model parallelism well beyond eight nodes (current state-of-the-art).

Seamless scaling: Deep learning practitioners will no longer need to explicitly think about parallelism in their models. Nervana's tools take a high-level definition of a deep network and automatically distribute the computations to achieve multi-node scaling of speed.

Ease of use: Neon is Nervana's open source deep learning framework. It makes defining and using neural networks easy by providing appropriate high-level abstractions in the standard Python programming language and easily integrates with existing data pipelines. Unlike other open source frameworks, Nervana supports, maintains, and applies a rigorous testing methodology to Neon.

The future of deep learning as the solution to big data processing is bright. With the right selection of tools and technologies, existing impediments to usage and scale can be surmounted. 

Naveen Rao, Ph.D., is CEO and co-founder of Nervana Systems. Prior to Nervana, he was a part of Qualcomm's neuromorphic research group, leading the effort on motor control and supporting business development. Rao previously worked in finance doing algorithmic trading optimization at ITG, designing novel processors at Sun Microsystems and Teragen, as well as specialized chips for wireless DSP at Caly Networks, video content delivery at Kealia, Inc., and video compression at W&W Comms. Rao holds a Ph.D. in Neuroscience from Brown University and a BSEE in Computer Science and Electrical Engineering from Duke University.

Predictions with Big Data

By Devavrat Shah

The primary vision of big data is the ability to make better decisions using the wealth of information available in data. So far, we have failed in realizing it.

Even though big data infrastructure is a solved problem, we cannot efficiently predict unknowns using data. This is the main impediment when making meaningful decisions using data. To address this challenge, we need an ultimate prediction engine that can consume large amounts of unstructured data and provide accurate predictions of the unknowns.



Background

Big data infrastructure is a solved problem. We know how to collect massive amounts of data (e.g., web scraping, social media, mobile phones), how to store it efficiently to enable queries at scale (e.g., Hadoop File System, Cassandra) and how to perform computation at scale with it (e.g., Hadoop, MapReduce). And yes, we can visualize it, too (e.g., *New York Times* visualizations). However, we are unable to make meaningful decisions using the data. Actually, we are terrible at it. The primary reason for this is our inability to grapple with the uncertainty — we cannot predict the unknowns well.

The next important innovation has to overcome this fundamental limitation. We need an ultimate prediction engine, an oracle, a prophet, the clairvoyant, the astrologer, and the fortune teller of big data. We can throw all the data we have at it to be processed, resulting in answers to questions about the unknowns. Of course, it cannot always predict the unknowns with certainty, but it will provide us the confidence associated with predictions based on the data made available to it. If we can achieve this, the ultimate dream and vision of big data will be realized.

Thinking About Data

It is important to understand how to think about the available data in order to build an ultimate (or for that matter, any) prediction engine. To help us model the data in our minds, consider the following example of data being stored in a potentially giant Excel file with multiple sheets. The data is organized in each sheet separately depending upon the type. The columns in each sheet correspond to different fields, while rows correspond to different records and each cell contains the specific information. Unlike a standard Excel file, in this model, each cell can have all types of information such as text, an image, an audio clip, or even a video in addition to the standard Excel data formats.

Through the lens of this mental model, let's consider an example scenario for the Intelligence Community. There is a wealth of data available about various forms of resource deployment including human and physical; communication between individuals and groups is available across various media; and financial transactions along with historical information about events of interest are available.

For all of this information, we may have data stored in different Excel sheets: one describing resource deployment, another describing communication between individuals and groups for each type of media, one containing financial transactions, and one containing historical information about events of interest. The information in cells of Excel sheets may be complex, such as audio, video, or text. Such a collection of information could be useful for predicting an impending event before it happens.

As another example, consider a retail organization where any operational decision involves knowing the unknowns about customers. Concretely, a retailer like Amazon would like to know what a customer may be interested in purchasing based on her or his recent and past history. The corresponding data involves the customer's history of purchase and transaction data, browse and search logs, reviews provided, and complaints filed. In addition, the retailer may have access to customer demographic information potentially obtained using third-party databases. And of course, product catalog information such as brand, textual description, price, and image are known.

Similar to the earlier example, all of this data can be viewed as a giant Excel sheet. Purchases on each row correspond to different transactions with each column containing a different attribute of the transaction, such as the time it was executed, the customer, product, price, discount, etc. Similarly, separate sheets for browse and search logs, reviews, and complaints filed may exist. Furthermore, another sheet can describe customer demographics with rows corresponding to customers and columns corresponding to different properties like age, sex, address, zip code, or ethnicity. Finally, there is a sheet storing product catalog information with each row corresponding to a product, and columns corresponding to price, brand, image, and description. While this data may actually be stored in Cassandra or Postgres, it perfectly fits our mental model of a giant Excel file.

An even simpler example is that of the highly popularized Netflix Prize challenge. Here, information on a large collection of movies and their star ratings given by a number of users are known. The goal is to predict a rating that a user might give to a movie for which his or her rating is unknown. Because the challenge is based only on

such ratings, the corresponding data can be represented in an Excel file with a single sheet where each row corresponds to ratings of a user and each column corresponds to a movie. The value in a cell corresponds to the rating of a given user for a given movie.

Predicting the Unknown

In a world where everything is known, all the cells in all the rows and columns of all the Excel sheets are filled. In reality, many of these cells have missing information. The goal of the prediction engine is to fill the missing information for these cells based on all other available information.

In the Netflix example, in an ideal world, we would know the ratings of all users for all movies. In reality (and as in the challenge), only a few ratings for each user are known. The goal is to predict the unknown movie ratings for users — i.e., fill the empty cells in the only Excel sheet of the file. This is known as the recommendation problem, for which collaborative filtering is a popular solution. Indeed, viewed in this special case, the single sheet Excel file with cell information being ratings is the well-examined problem of recommendations or personalization.

Suppose we have an additional sheet in the same Excel file with information about various users' opinions expressed on the Internet Movie Database (IMDb). Then the information about user preferences expressed through IMDb can be used to further enhance predictions of users' ratings on the Netflix sheet. Indeed, this insight was used to show that the release of the ostensibly anonymous Netflix data set during the challenge was not really anonymous, as information about an anonymous Netflix user could be used to identify users from their public profiles on IMDb.¹

Generally speaking, it makes sense that combining information across the sheets of an Excel file, when they are available, can greatly boost prediction of the missing information.

In the Intelligence Community example discussed earlier, using communication patterns, financial transactions, and mobilization of resources, an impending event, potentially rare, can still be predicted if we have enough collective data across all sheets of the Excel file.



**There lies true value in big data,
and its extraction relies on an
effective prediction engine.**

In summary, an ultimate prediction engine should solve the problem of missing information. And it's much more than the classical problem of recommendation or any other known prediction problem, including regression.

The Celect Engine

Celect has made significant progress towards realizing this dream of building an ultimate prediction engine. Celect's prediction technology, powered by the Celect Engine, can accept data essentially in the form of a giant Excel file with multiple sheets. Celect asks the end user to identify the type (or in Celect's language, *action*) for each data unit. In the mental model discussed earlier, types correspond to different sheets. Each data unit then corresponds to a row of one Excel sheet. The columns have natural association to what are called actors, businesses, or features. The value of each data unit can be effectively anything, including numbers, text, images, audio, or video.

The end user, after throwing all the data at Celect Engine, can query the Engine to predict the unknown value in a given cell. And Celect Engine responds with the prediction and a confidence score based on all the available information.

Although this may appear a simple task, in reality there is a problem of sparsity: information is incomplete and often rare. This makes it really hard to predict well. For example, in the case of predicting rare events, we are faced with exactly such a difficulty: the auxiliary information across domains may seem innocuous individually. Only by cleverly stitching together all the data, as performed by Celect Engine, an accurate prediction may surface from the collective data.

Using the retail example we discussed earlier: the interest of a retailer is primarily in the action of customer purchase. However, the data associated with it is actually very sparse — an individual customer buys very few products in any given retailer's catalog over the duration of a year. On the other hand, the data associated with browse and search logs is quite a bit richer. Therefore, by using all such information to predict the relatively rare event of a purchase, one can achieve significantly better accuracy. Indeed, for Celect's retail customers, more accurate predictions lead to better online personalization (20 percent increase in revenue) as well as in-store assortment optimization (7 percent increase in revenue). A reader may be left wondering how well an approach like collaborative filtering (CF) performs. The performance gains obtained by Celect in online personalization are primarily with respect to CF-based solutions. This is principally because CF-like approaches do not stitch together data across Excel sheets and they do not naturally handle complex data forms like text, images, audio, and video.

Similar insights have been remarkably effective across different domains, including financial markets and social media. For example, when utilized for predicting the price of Bitcoin, a simple trading strategy using the resulting predictions led to doubling of investment over a period of 50 days without incurring a giant volatility penalty (concretely, Sharpe ratio is 4.1).² In the context of Twitter, it led to accurate prediction of future trends.³ Specifically, it predicts trends with a true positive rate of 95 percent and a false positive rate of 4 percent. And it does so by delivering predictions on average 1 hour and 45 minutes in advance. A priori, we did not expect it to perform so well given that all prior


attempts in the literature, some with detailed context-specific model, failed at getting close to such a remarkable performance.

Prediction Provenance

Any prediction system will have errors or mispredictions. Therefore, it is important to understand how to handle such scenarios. In the context of Netflix, presenting a wrong movie to customers only so often is inconsequential. However, in scenarios where humans are involved in making decisions based on predictions, if the decisions have critical consequences such as mobilizing expensive resources for the Intelligence Community, mispredictions are expensive. In such scenarios, one way to guard against misprediction is to explain to the end user why the system has made a given prediction and provide the provenance of the prediction. Then the end user can judge whether the prediction is meaningful or not. Providing such evidence for a prediction can also help decision makers interpret the prediction and justify consequential decisions to

the rest of the organization, if needed. Therefore, it is important for a prediction system to not only provide accurate predictions and confidence, but also a proof, certificate, or provenance of predictions. The Celect Engine naturally provides a narrative proof — for every prediction, it produces existing data points that are effective witnesses for the prediction, and by expressing the data in Celect's language, it leads to semantic understanding of the proof.

Conclusion

The ultimate vision of big data is to aid decision making using a wealth of information from the data. The key impediment in realizing this vision is the inability to make accurate predictions. We need an ultimate prediction engine. The classical recommendation systems, in a sense, took the first steps towards designing such an engine. The Celect Engine has made significantly more progress towards achieving this ultimate goal. There lies true value in big data, and its extraction relies on an effective prediction engine. 

Devavrat Shah is an Associate Professor with the department of Electrical Engineering and Computer Science at MIT. He is a co-founder and Chief Scientist of Celect, which helps retailers decide what to put where by accurately predicting customer choice using omni-channel data. His primary research interest is in developing large-scale machine learning algorithms for massive unstructured data. Shah has made contributions to the development of "gossip" protocols and "message-passing" algorithms, which have been pillars of modern distributed data processing systems. He received the 2010 Erlang Prize from INFORMS. He is a distinguished alumni of IIT Bombay, from where he graduated in 1999 with the President of India Gold Medal.

ACKNOWLEDGEMENTS

This article is based on numerous discussions and collaborations author has had with colleagues at MIT and Celect. In particular, author would like to acknowledge George Chen, Vivek Farias, Ying-zong Huang, and Vighnesh Sachidananda.

REFERENCES

- ¹ *How to break anonymity of the Netflix Prize Dataset*, A. Narayanan and V. Shmatikov, IEEE symposium on security and privacy, 2008.
- ² *Bayesian regression and Bitcoin*, D. Shah and K. Zhang, Allerton 2014.
- ³ *A latent source model for non-parametric time-series classification*, G. Chen, S. Nikolov and D. Shah, NIPS 2013.

AI Roundtable: Intelligence from Lab41's Technical Advisory Board

A Q&A with Steve Bowsher, Jeff Dickerson, and Josh Wills



Lab41® brings together four distinct communities — the Intelligence Community (IC), academia, industry, and In-Q-Tel — to solve IC challenges in big data and analytics. Challenges are vetted by Lab41's Technical Advisory Board (TAB), which includes representatives from each of these four communities. The *IQT Quarterly* recently spoke with several TAB members to gather some human intelligence (HUMINT) about artificial intelligence, including their thoughts on the most innovative organizations and applications, the industry's road ahead, and a showdown of AI buzzwords and pop culture references.

In what fields has AI really hit its stride?

Steve Bowsher (IQT): Recommendation engines such as Amazon and Netflix. Voice recognition such as Siri. Anti-spam engines.

Jeff Dickerson (IC): With the growth of deep learning techniques, I think the ability for AI to aid in the interpretation of unstructured data has really matured. I think this has been most notable in the areas of image and speech processing. For both of those domains, the improvement in machine learning capabilities over the last couple years has been outstanding.

Josh Wills (Cloudera): Video games, specifically the researchers at DeepMind (acquired by Google last year) that created a deep learning model that incorporated reinforcement learning and was able to teach itself to play Atari 2600 video games at an expert level. To me, the most significant limitation on the ability of AI

systems to solve problems is their lack of access to contextual information about the problem domain that is available in a form that they can access. In the case of a video game, all of the information about the system is digital by definition, and thus is amenable to what we call AI.

In what fields is AI still unproven? Why?

Bowsher: Cyber. Analysts need to see the path to the conclusion in order to trust it and AI/machine learning struggles to demonstrate this to an analyst. It wants everyone to trust that the system has provided the right answer without being able to show how it got there.

Dickerson: I think AI techniques have remained largely confined to very specific domains and to assist with relatively low-level tasks. While there are impressive natural language techniques, they are not as mature or ubiquitous as many initially hoped.

Wills: It's essentially the opposite of my first answer: the fields where AI has the furthest to go are the fields where most of the context about problems does not have an obvious digital representation, or where there isn't any incentive or inclination to create machine-friendly representations. The fields that are most resistant to rational thought, like literary criticism and politics, are safe from AI for the time being.

Where are we in the AI hype cycle — trigger, peak, trough, slope, or plateau?

Bowsher: Somewhere on the downslope of peak heading to the trough of disillusionment.

Dickerson: I think AI has already been through several iterations of the hype cycle and I'm frankly surprised that the term has returned as strongly as it has. To some extent, I think we do the impressive emerging techniques in machine learning and particularly deep learning a disservice by using the phrase "artificial intelligence."

Wills: Here's a better question: which AI hype cycle are we in? There were AI hype cycles in the '60s and '80s that crashed pretty hard, so I think that in a macro sense, we're on the slope of enlightenment. In the micro sense of the current AI hype cycle, it feels like we're heading towards yet another peak of inflated expectations.

What is the most interesting application of machine learning/AI that you have seen?

Bowsher: Automatic image recognition that Google is demonstrating.

Wills: I've been pleasantly surprised by how well the convolutional neural networks created by Berkeley's Caffe project have been able to generalize to different kinds of image classification problems, including on images that the network itself was never trained on. I think that ML tools aren't really that interesting until you no longer have to be an ML expert to use them, and Caffe is probably the most advanced tool I've seen that is easy enough for your friendly neighborhood data scientist to use on their own image classification problems.

What tool, technology, or technique has been most transformative in the last few years?

Bowsher: Deep learning.

Dickerson: I think voice recognition has rapidly progressed from a few niche tools with questionable performance to a core aspect of a number of consumer and professional products, starting with mobile devices but now moving into the home in a big way.

Wills: I have to give the award to graphics processing units (GPUs). Most of the advances for the last few years would not have happened had GPUs not become so powerful and so cheap. The math behind the deep learning models has been pretty clear since the 80s, we just didn't have the computational horsepower to be able to find parameterizations that were actually useful on real problems until recently.

Who is the most innovative organization in AI currently?

Bowsher: Google then Baidu then Microsoft Research. Google has put the most effort into it, for the longest time, in the most diverse ways.

Dickerson: This is a very difficult question for me, largely because of the breadth of organizations active in this space. I think the dramatic increase in consumer home AI/ML applications has been particularly noteworthy in recent years and the creativity in finding more and more new applications is just awesome.

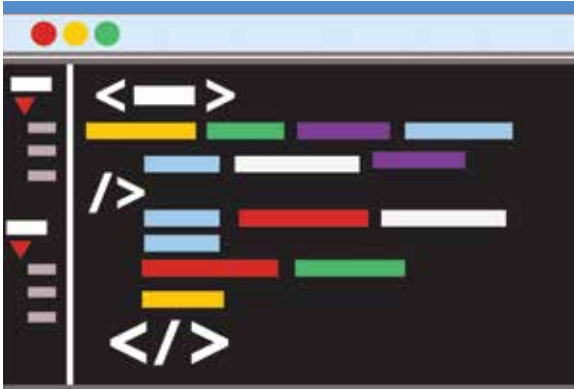
Wills: It's a tough call between Google and Baidu, but I'm going to give the edge to Google. I'm biased, because I have friends at Google, and I've had the opportunity to do things like ride in one of their self-driving cars. It's a bit eerie at first, because you never really notice the subtle imperfections of human drivers until you're in a computer-driven car that accelerates and brakes perfectly. Then, once you get used to the computer driving, having the human driver take over again becomes unsettling. I don't quite know how to explain how this feels; riding in a self-driving car is sort of like riding on the monorail at Disney World, but without the track. After about 10 minutes, I swear you'll never want to go back.

What human task will computers take over in the next five years and why?

Bowsher: Day trading in the stock market. Computers can just do it faster than humans.

Dickerson: I'd love to see us get to viable driverless cars, but I think five years is a bit too optimistic for that. I do think we'll see far more common ML-aided safety features in cars (collision avoidance, etc.).

Wills: I'm bullish on automated financial advisory services like Wealthfront, Betterment, and Future Advisor. Investing and financial planning is one of those fields where emotions are actively detrimental to good decision making, and so I expect that robo-advisors will move up the chain to take over more and more of our



"There were AI hype cycles in the '60s and '80s that crashed pretty hard, so I think that in a macro sense, we're on the slope of enlightenment. In the micro sense of the current AI hype cycle, it feels like we're heading towards yet another peak of inflated expectations." – Josh Wills

financial lives — estate planning, life insurance, helping us negotiate home sales, etc.

What is your favorite algorithm?

Bowsher: Amazon's recommendation engine.

Dickerson: It's been far too long since I was personally implementing any algorithms, but I think I'll go old school and say k-means clustering. While it has many faults, its simplicity and ease of use have kept it as a workhorse even as new algorithms come online.

Wills: Weighted reservoir sampling. It's so simple and so pretty, but humanity only discovered it a decade ago. I even wrote a blog post about it.¹

What role is open source playing in AI?

Bowsher: Developers gravitate to open source and this world is all about the developers, so it is a very strong role.

Dickerson: I think open source has been a major enabler of recent advances. While there are certainly proprietary implementations with their own advantages for particular algorithms, the widespread availability of solid open source versions has really levelled the playing field.

Wills: I think the role is two-fold: first, much of the cutting-edge research in deep learning is done with two open source frameworks, Torch and Theano. So open source is the environment where all of the latest and greatest stuff happens. Second, the popularization of deep learning is coming via more user-friendly frameworks like Caffe and DL4J, and in those cases, open source is democratizing deep learning and making it available for everyone to use on their own problems.

Pick A or B:

Machine Learning or Deep Learning

Bowsher: Deep learning. All the cool kids are doing it.

Dickerson: Machine learning. Deep learning feels a little too much like a buzzword to me.

Wills: Machine learning — deep learning is great, but there are a lot of useful problems where the computational overhead of deep learning is overkill.

Google or Baidu

Bowsher: Google. Bigger, better, and more diverse.

Dickerson: Google, just because I've used it more.

Wills: Google, modulo my previous answer.

Siri or Google Now

Bowsher: Siri because Apple gets the consumer better than Google does.

Dickerson: Siri because it was the first really effective use of generic voice recognition I saw.

Wills: Siri. She has a better sense of humor.

Python or R

Bowsher: Python. I have a Python book on my desk and am trying to learn it. There is no way that I could learn R.

Dickerson: R. I probably use Python more for generic tasks, but it's hard to beat a domain-specific language

Wills: The data analyst in me loves R, the engineer loves Python. I think I'm more of an engineer these days, so I'm going to go with Python.

OS X or Linux

Bowsher: OS X for user facing machines or apps. Linux for headless machines in the cloud. Highly disappointed that there is no Windows option.

Dickerson: Linux, just on general open source principles.

Wills: OS X, because I give too many PowerPoint presentations.

Data science or big data

Bowsher: Data science because it provides answers. Big data is just a pile of information without data science.

Dickerson: Really neither, but data science if I have to choose. Both have transformed into buzzwords, but data science at least implies some rigor.

Wills: Oh, Alan Wilkis [creator of electronic music project Big Data], without a doubt.

Chappie or Johnny 5

Bowsher: Johnny 5. I am old school! Much funnier and more kid friendly.

Dickerson: Twiki/Dr. Theopolis. I'm still holding out hope for a Buck Rogers reboot.

Wills: Johnny 5.

Her or Ex Machina

Bowsher: Ex Machina because it is better than Fast & Furious 7.

Dickerson: Ex Machina, I found the premise far more interesting than Her.

Wills: Her. [Q](#)

Steve Bowsher serves as Managing Partner and Executive Vice President at In-Q-Tel, leading the company's technology investment strategy. Bowsher joined IQT from InterWest Partners, where he served as General Partner. He specialized in the enterprise software and Internet sectors, and led and managed 13 investments in those areas. Previously, Bowsher worked at E*TRADE, managing its value-added products and services. During his tenure there, he helped launch Destination E*TRADE, the company's award-winning website. Bowsher was also an early stage employee at two startup companies, where he was responsible for revenue and distribution targets. Bowsher graduated magna cum laude from Harvard and received his M.B.A. from Stanford.

Jeffrey C. Dickerson is the NSA/CSS Chief Architect. Prior to his current position, he was the lead technical director at a field site and the Technical Director and Deputy Technical Director of the Signals Intelligence Directorate (SID). Post-9/11, Dickerson served as one of the founding technical leaders of the Network Analysis Center within the SIGINT Development Strategy and Governance Organization, where he was actively engaged in developing new analytic techniques and strengthening foreign partnerships. He holds an M.S. in Electrical Engineering and Computer Science, and a B.S. in Electrical Engineering from MIT.

Josh Wills is Cloudera's Senior Director of Data Science, working with customers and engineers to develop Hadoop-based solutions across a range of industries. He is also the founder of the Apache Crunch project for creating optimized data pipelines for Hadoop in Java and Scala. Prior to joining Cloudera, Wills worked at Google, where he worked on the ad auction system and then led the development of the analytics infrastructure used in Google+. Wills earned a B.S. in Mathematics from Duke University and an M.S.E. in Operations Research from the University of Texas, Austin.

REFERENCES

¹ <http://blog.cloudera.com/blog/2013/04/hadoop-stratified-randosampling-algorithm/>



Figure 1 | Data-driven behavior analysis and modeling enables anticipatory sensemaking through storytelling. A rich source of such data are multi-player online game logs.

AI for the Analyst: Behavioral Modeling and Narrative Processing

By Adam W. Meade and R. Michael Young

Human analysts always have, and will continue to, lie at the heart of the work of the Intelligence Community (IC). However, the IC is not immune to challenges presented by big data. Analysts are experiencing an ever-increasing need to rely on artificial intelligence (AI) to process, evaluate, and present data in a manner that they can understand and communicate to decision makers. Some approaches to AI attempt to replace human analysts with AI counterparts by replicating the processes of perception, judgment, and decision making inherent in intelligence analysis.

As a community, the IC cannot and should not be working to replace its most valuable assets with machines. Instead, the Laboratory for Analytic Science (LAS) seeks to improve the efficiency of analysts by utilizing methods and processes derived via AI research. With this approach, AI is not used to replace human judgment and decision making, but rather to facilitate alternative, more accurate, and more efficient ways for analysts to accomplish their tasks.

The LAS is charged with developing the science of analysis and analytic methodologies. This simultaneous, blended study of tradecraft and technology serves as the foundation upon which all LAS efforts are positioned.

AI methodologies inform the modeling of measured observations, as well as the characterization of analytic workflows manifested by IC analysts. AI comprehensively informs the understanding of static and slowly varying data, data rapidly observed at the sensor level, and the evolution of possible future states of entities through behavior modeling. At LAS new advances in AI-inspired tradecraft, technology, and user experience are divided into three stages of investigation: reflection on the past, observation of the present, and imagining possible futures. Two of the foundational AI-informed activities at the LAS are sensemaking through storytelling and modeling analyst behavior.

Sensemaking through Storytelling

LAS is building mission-relevant technologies that understand and complement human cognitive activities through narrative. Psychologists tell us that narrative is one of the fundamental modes of human sensemaking: intelligent systems that leverage narrative can take advantage of our existing story-centered cognitive machinery to help us comprehend complicated event-based data through the frame of stories and their telling.

The narrative systems being designed to support sensemaking will have a range of explanatory capabilities that support reflecting upon previous data, observing current data, and imagining possible future worlds that extend what is known or believed. Because some narratives intentionally raise more questions than they answer, explanation is a critical capability linked to reflection on a narrative's past progression. Narrative systems must be able to explain the who, what, when, where, why, and how of their stories.

Narrative explanation often involves questions about the internal structure of a specific story. The process of internal analysis of a single story is called intraspection. Intraspection is a critical part of sensemaking in a narrative context because a story's structure is often not immediately and completely clear to readers. Well-constructed narratives are intentionally designed to obfuscate or elude underlying detail; failure to do so would make the story incoherent. Because stories are based on significant low-level data, the ability to provide insight into that detail upon demand is a critical capability for narrative systems.

To facilitate narrative introspection, the systems being developed build up models of story structure that leverage existing approaches to reasoning about action and change in a sub-field of AI called planning. Planning systems were initially designed to synthesize action sequences that could be used to guide autonomous agents or robots in the manipulation of their environment to achieve a set of goals. Because these plan structures also characterize the actions, plans, and goal-directed behavior of characters within a story, we leverage generative methods from AI planning to build data structures that model the complicated plot-related activity of a story. These models then can be referenced

to give answers to introspective questions about the previous trajectory of an unfolding narrative.

For users to understand the current state of an unfolding narrative, the narrative systems are being built to leverage the coordinated use of narrative discourse in one or more media. The systems are extending work in computational linguistics to design text, cinematics (short video segments filmed using intelligent camera systems within a 3D virtual set), and map-based story content to effectively convey a data set's story. By coordinating the content of a narrative discourse across several media, an author can express and focus on different elements of stories. The models adapt a story's telling by distributing communicative goals across available media resources.

One central problem for narrative generation is the effective modeling of an analyst's understanding of a narrative as he or she experiences it. As readers experience a narrative discourse, they update their mental model of the story so that the new elements that they confront form a coherent structure with their existing beliefs. To do so, they have to draw inferences and make assumptions about missing links: actions that might have happened but are not presented, goals of the characters that would justify their actions, etc.

While questions about the structure of a given story are important to answer, questions that arise when imagining alternative or future narratives have equal importance. Questions like "why not...?", "what if...?" and "what else...?" require the consideration of stories and discourses related to, but different from, some narrative starting point. To answer these questions, a system must look beyond a single story to consider the space of narrative possibilities that might account for or extend the given data. In contrast to introspection, the reasoning that spans elements in a space of narratives is called extraspection. Extraspection is a critical element in support of the imagination of the as-yet unexecuted portions of a story.

A critical formal property for the algorithms that generate narrative spaces is that of completeness. Informally, completeness for a narrative space generation algorithm is the property that guarantees that all valid narratives that cover a specific data set are



While AI is at the heart of efforts to improve analytic workflow, ongoing efforts must be focused not only on automated methods and processes, but human analysts.

Figure 2 | Analysts will require new user experiences to interact with AI-enhanced analysis of large-scale data at rest and in motion, both measured and hypothesized through behavior modeling.

created. This is a valued property for several reasons. For instance, this means that if the generation algorithm halts and reports that it cannot construct a story for given data, then no such story exists. Further, it means that if there is some obscure story that covers the data, the system will find it. Finally, it means that if there are a large but finite number of stories that cover a given data set, the system will generate all of them (given adequate time and space resources).

These kinds of narrative spaces are challenging for humans to navigate for a number of reasons. Because elements or nodes in these spaces represent the many ways to fill in story structure around a given set of data, narrative spaces typically contain thousands or tens of thousands of nodes even for short stories. Any two nodes connected in the space are quite similar to one another, so local navigation through the space yields little change in the stories being reviewed. To support an analyst's effort to imagine the future extensions to an unfolding narrative, we are developing qualitative and quantitative characterizations of story structure that can help analysts explore this space.

Modeling Analyst Behavior

Another fruitful approach to the use of AI is trying to understand the behaviors and processes of analysts with particular focus on the analytic workflow. Analytic workflow refers to the method by which users apply personal knowledge, experience, skills, tools, and

organizational resources to accomplish tasks through work processes. The ultimate goal of an AI-focused approach to improving analytic workflow is to generate a series of processes, methods, systems, and tools by which users can produce work products more efficiently and/or that are of higher quality, via methods that they find more rewarding and that support positive psychological effects such as job satisfaction and skill development. The goal of this work is to create a flexible and self-sustaining analytic workflow methodology for extracting, representing, characterizing, evaluating, and recommending workflow across novel situations and work environments.

The use of AI in the workflow process could afford improvements in a variety of ways. Specifically, for analysts, methods are being developed that could suggest tools and work processes that yield improved performance outcomes given measurable attributes of the analyst and the environment. For instance, given the analyst's previous usage history (e.g., tool preferences, workflow patterns) these methods could categorize the current workflow and suggest alternatives. Such alternatives could attempt to maximize criteria such as efficiency with respect to task completion or quality of product. Alternatively, in some contexts, AI approaches could suggest an alternative workflow intended to maximize novel criteria such as mastery of new tools or user engagement. Ideally, the methods and tools will be capable of suggesting workflow alternatives even in

cases where the work or its environment exhibits novel features. Moreover, via mixtures of direct data collection of individual analyst data and usage over time, tailored workflow suggestions will be proposed for the individual.


For a team, this could suggest ways to promote effective workflows at the team level. AI tools could capture work patterns relating to particular areas of expertise and suggest workflows that divide tasks along relevant areas of knowledge, skill, and ability among team members. Additionally, these tools could suggest workflows to encourage interactions and collaboration at critical points in the workflow to improve the selection and enactment of work processes.

For large organizational units, LAS's work focuses on propagating user-level and team-level improvements by aggregating workflow information and lower-level tool usage data in order to suggest workflow improvements across the organization. Additionally, the goal is to leverage information about variations within the organization to optimize workflow. For instance, AI methods may preferentially suggest workflows that are more common within more similar organizational units as unit norms, and data usage restrictions may directly influence workflows.

While AI is at the heart of efforts to improve analytic workflow, ongoing efforts must be focused not only on

automated methods and processes, but human analysts. The best AI-based tools and recommender systems are useless unless the results of those recommendations are heeded by the user. Consequently, current work is focusing on how to present recommendations to analysts to encourage adoption. Additionally, for AI-based systems to aid in recommending alternative workflow, it is critical to understand how to measure the quality of intelligence output in context. LAS is also pursuing research related to understanding the differences in quality of intelligence processes and the extent to which perceptions of quality vary across different producers and consumers of intelligence products.

Conclusion

LAS is helping write the future of AI through the development of intelligent systems that simultaneously take advantage of what machines do well and what humans do well. By using narrative as a platform, the technology can present data in ways that facilitate probes, hypotheses, and inquiries by analysts and decision makers to maintain global awareness and enable strategic foresight. In addition, LAS is focused on analysts and their use of this mission-inspired technology to characterize, understand, and optimize technology-driven tradecraft. Focusing on only one or the other will not yield the results the IC demands to meet its mission. 

In 2013, the National Security Agency (NSA) and North Carolina State University (NCSU) launched a unique partnership by creating the Laboratory for Analytic Sciences (LAS; las-ncsu.org) in Raleigh, North Carolina. The mission of LAS is to imagine, investigate, and implement innovative solutions for a variety of tactical and strategic analytic challenges. Government personnel, stationed in Raleigh, collaborate with a diverse set of academic and industry partners to create advanced technology and analytic tradecraft in accordance with the U. S. Intelligence Community objective to maintain global awareness and strategic foresight.

Adam W. Meade, Ph.D., is a Professor of Psychology at North Carolina State University and a Fellow of the Association for Psychology Science and the Society for Industrial/Organizational Psychology. His areas of research center around psychometric issues in organizational research. Meade serves as an Associate Editor at *Organizational Research Methods* and on the editorial boards of the *Journal of Applied Psychology* and *Applied Psychological Measurement*. He also is the founder of Scientific Organizational Solutions, which specializes in employee selection test development, computer adaptive testing, and organizational assessment. Meade completed his Ph.D. in Applied Psychology from the University of Georgia.

R. Michael Young is a professor of Computer Science at North Carolina State University, where he is Director of the Digital Games Research Center and leads the Liquid Narrative research group. Young is a Senior Member of the Association for the Advancement of Artificial Intelligence and an ACM Distinguished Scientist. His work focuses on the computational modeling of interactive narrative, especially in the context of computer games and virtual worlds.

DeepDive: Enabling Next-Generation Business Intelligence with Information Extraction

By Michael Cafarella



Many organizations have huge amounts of information buried in text, tables, and images. This “dark data” contains important information but lacks the relational structure that most data management tools — such as business intelligence (BI) pipelines and analytics systems — rely on. This is a shame: if we could somehow unlock this dark data, its use for relational analytical pipelines would likely yield many better data-driven decisions.

DeepDive — an academic system from Stanford that is part of the DARPA Memex project — aims to solve this problem. It is a system for information extraction. It populates structured relational databases using information found in natural language text and images. The resulting structured data reflects the contents of this dark data but can be processed using standard tools such as Tableau, SQL databases, BI systems, or even Excel.

The usefulness of information extraction is not a huge surprise: it has been an academic topic of interest since at least the mid-1990s. What is novel is the extremely high accuracy that DeepDive is able to obtain. In several areas, we have been able to obtain results that are competitive with human beings. This level of accuracy unlocks a huge number of possible applications, including many relevant to challenges faced by the U.S. government. Further, DeepDive is unusual for an

academic project in the maturity of its code base and the extent to which it has been evaluated on many different applications.

Design Goals for an Information Extraction System

An effective information extraction system should yield high-quality data at a low engineering cost. We believe reaching this goal entails several important desiderata:

Mechanism independence: Users should be able to succinctly describe a knowledge base construction task and ignore the details of the machine learning algorithms necessary to accomplish the task. Instead, users should focus on domain knowledge in the form of features and data choice.

Integrated processing: A single system should perform all the typical data manipulation steps: extraction, integration, reduplication, and cleaning.

Iterated improvements: It should be possible for a user to iteratively improve output data quality, much as a software engineer does for software quality.

Mechanism Independence

Unlike machine learning toolkits such as MLib, Mahout, or SAS, an effective extraction system does not have to offer the user an elaborate suite of statistical algorithms. Indeed, the user ideally has little choice over what training or inference algorithm is actually used. Instead, the user should describe domain-specific information about the knowledge base construction task at hand. That information can include details about the structure of the problem; for example, the user should indicate that when the system determines whether two individuals belong in the married relation together, it should consider whether their age values are similar. The user can also indicate the input corpora to process, useful feature code, and distant supervision rules. The user's focus should be on domain knowledge, not algorithms.

The details of formulating the machine learning problem and choosing the algorithm can be daunting. Further, we and others have found that choosing a specific learning algorithm is usually a narrow technical question that most users are ill-suited to make. For an experienced practitioner, algorithm choice is usually not the primary barrier to high accuracy, but an inexperienced user can easily make a poor choice that makes high accuracy impossible. Mechanism independence allows the user to focus on what he is uniquely qualified to handle: domain-specific details.

Integrated Processing

The system should handle multiple previously-distinct data processing steps — extraction, integration, reduplication, and cleaning — in a single platform. This approach may seem unnecessarily ambitious, as past systems have treated each individual step as the focus of entire isolated software products. In contrast to the past siloed processing model of isolated software components, we believe that integrated processing is crucial for obtaining high quality.

Integrated processing allows the human user to fix problems where they are easiest to fix. Consider a case when the user tries to create a book sales catalog out of information that was crawled from an e-commerce site's web pages. The knowledge base schema is something like (bookTitle, author, uniqueness, price). Imagine that the extractor has very high precision of 95 percent.

The extractor produces data which is eventually sent to the cleaning module. Imagine that 5 percent of the tuples in the knowledge base are not books, but instead are movies that were improperly extracted. In a non-integrated system, the downstream cleaning module must now attempt to distinguish between tuples that have book titles from tuples with movie titles (a challenging task, unless there is a flawless dictionary of media titles available). This is essentially an extraction error that has been propagated to a software module that is ill-equipped to fix the error. It would have been easier to fix the problem in the extraction module, which has access to useful hints embedded in the raw HTML.

It would be tempting to say that the extractor module in this case was simply a failed piece of the siloed processing system: 95 percent is not good enough. But if the extractor bug had yielded incorrect television shows rather than movies, the cleaner could have simply filtered the price column to efficiently fix the problem. Indeed, it might have been more difficult to fix such a bug in the extractor, which likely has no access to aggregate price statistics.

The core observation is that in a siloed system, software teams have no choice but to allocate effort based on the silo's measurable output quality. But errors from one step unexpectedly change requirements and costs in downstream steps. Consider the problem of evaluating whether a given function in a piece of software is "fast enough." For a function considered in isolation, the question is impossible to answer. One needs to know overall performance requirements, how often the function is called, etc. Each silo's near-term measurable output quality is basically worthless as a guide for allocating engineering effort; as a result, siloed systems cannot reliably improve their data product quality at a reasonable engineering cost. Their only options are low precision or extremely high costs (often in the form of humans who construct the knowledge base by hand).

In contrast, an integrated processing system enables users to fix problems where they are easiest to fix. Moreover, it does not ask the user to consider in a vacuum whether an intermediate data product is "good enough." Instead, it only asks the user to evaluate whether the knowledge base construction system's final data product is good enough for the target application. If the user's answer is no, the system enables her to fix errors effectively: by picking the worst errors first, and by fixing them in the most convenient stage in the processing pipeline.

Iterated Improvements

Our final design goal for a knowledge base construction system is that it should enable iterated improvements. That means that a user of moderate skill should be able to improve the quality of a data product through assiduous application of engineering time and effort. In other words, improving a data product should be like improving a software product: there is an improvement loop of diagnose-debug-repair that, when applied over and over again, allows users to eventually achieve the desired level of system quality.

It is well known that this improvement loop does not always apply in software products; some projects have dysfunctional designs or practices that prevent the loop from completing, or introduce new bugs as old bugs are resolved. However, the loop is common enough in practice that vast numbers of independent groups are able to create usable software products.

We believe this improvement cycle is crucial if knowledge base construction systems are to produce high-quality outputs at a reasonable cost. Without it, systems must ask for omniscient domain engineers (who get the answer exactly right the first time), or AI-complete systems (who do not need any human assistance), or for cost-insensitive projects (that can afford either human curation or extremely low-efficiency domain engineers). These options are not practical for most projects.

DeepDive Implementation

DeepDive is an information extraction system that meets the above design criteria. Figure 1 shows the primary processing steps.

The system starts with an input document corpus. In Step 1, DeepDive applies a large number of user-written functions to create features. These are small human-understandable observations about the data. For example, we might include the part-of-speech for each word in the input document. Other user-written code creates candidates: possible data extractions that should be considered probabilistically. These features and candidate functions form the bulk of what human engineers do for each novel extraction task. They will form the bulk of the intellectual property that ClearCut (the company behind DeepDive) builds over time.

In Step 2, DeepDive creates a probabilistic factor graph representation of the extraction problem. Factor graphs are a well-known construct in AI and are considered state-of-the-art for certain extraction and inference tasks. DeepDive is able to materialize extremely large factor graphs that can represent many millions of extraction candidates simultaneously.

In Step 3, DeepDive performs probabilistic inference, using the document data and human-written features to estimate a probability for each extraction candidate in the factor graph. Probabilistic inference has historically been considered extremely computationally burdensome. Thanks to work that is aware of the computer system's memory hierarchy, DeepDive is able to obtain vastly more samples than previous systems, and hence to compute vastly more and better probabilities.

Iteration and Improvement

A crucial question is how quickly the information extraction engineer can produce a novel database (sometimes called a knowledge base, or KB). As with

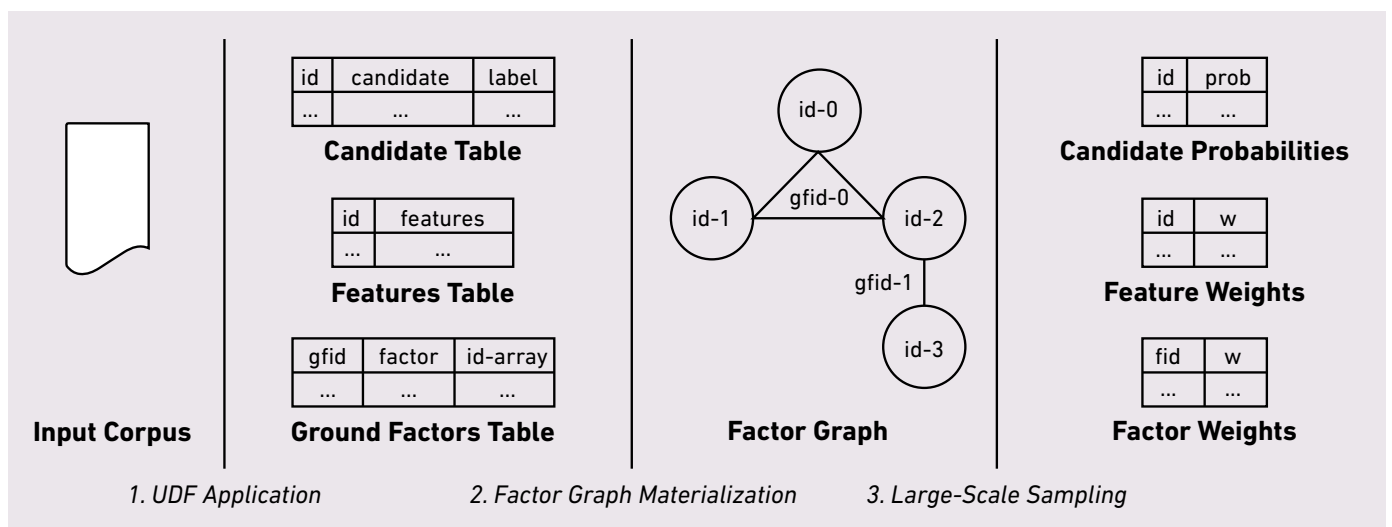


Figure 1 | DeepDive's primary processing steps.

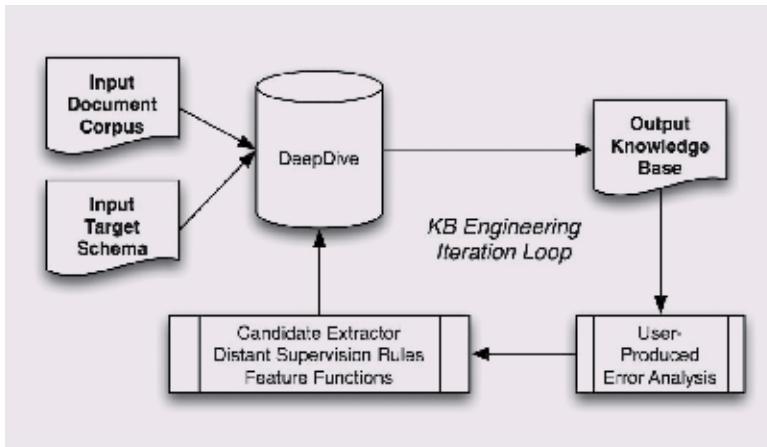


Figure 2 | DeepDive's knowledge base engineering iteration loop.

traditional software engineering, there is a well-defined development cycle for the DeepDive engineer.

The system starts with the input document corpus and a domain expert's target schema. The information extraction engineer then engages in an iterated improvement cycle:

Step 1: Run DeepDive and produce an initial knowledge base.

Step 2: Analyze the result for errors.

Step 3: Rewrite extractors, candidate generators, and other code to address problems in Step 2. Then go to Step 1.

DeepDive Applications

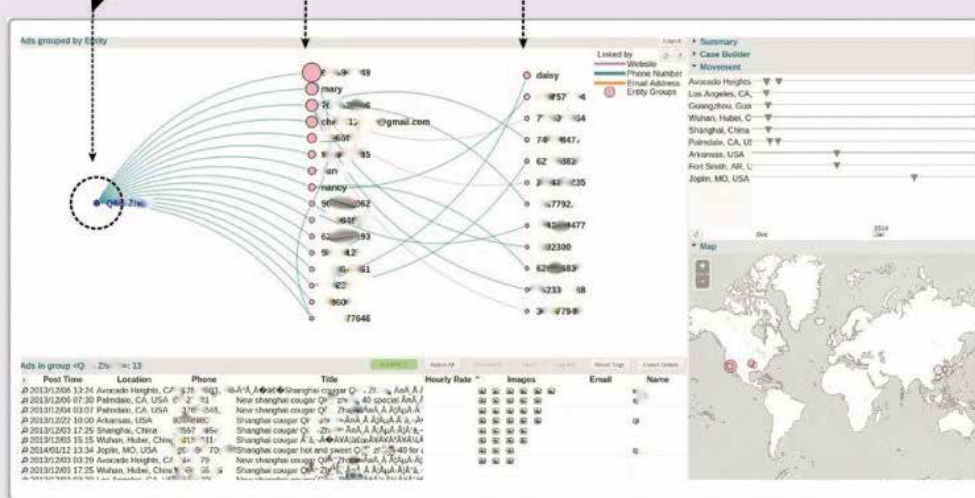
Unlike many academic projects, DeepDive has a depth of real uses and applications. We describe two below.¹

Human Trafficking

Memex is a DARPA program that explores how next-generation search and extraction systems can help with real-world use cases. The initial application is the fight against human trafficking. In this application, the input is a portion of the public and dark web in which human traffickers are likely to (surreptitiously) post supply and demand information about illegal labor, sex workers, and more. DeepDive processes such documents to extract evidential data, such as names, addresses, phone numbers, job types, job requirements, rates of service, etc. Some of these data items are difficult for trained human annotators to accurately extract and have never been previously available, but DeepDive-based systems have high accuracy (Precision and Recall in the 90s, which may exceed non-experts). Together with provenance information, such structured, evidential data are then passed on to both other collaborators on

The Big Data Behind Online Sex Trafficking

This circle is a name that appears in a sex ad. It's connected to email addresses, photos and phone numbers on other ads across the internet.



A powerful data-mining tool created by Darpa allows investigators to capture and visualize patterns of online criminal networks. Here, evidence of a possible sex trafficking ring is shown by capturing the relationship between content in ads across the web.

A timeline shows when and where those ads were placed. It also shows the movement of the ads over time.

By plotting thousands of ads investigators can see the geographic scope of networks involved in the sex trade for the first time.

Note: Private information is obscured

Source: U.S. Defense Advanced Research Projects Agency



An effective information extraction system should yield high-quality data at a low engineering cost. We believe reaching this goal entails several important desiderata: mechanism independence, integrated processing, and iterated improvements.

the Memex program as well as law enforcement for analysis and consumption in operational applications. Memex has been featured extensively in the media and is supporting actual investigations. For example, every human trafficking investigation pursued by the Human Trafficking Response Unit in New York City involves Memex. DeepDive is the main extracted data provider for Memex.

TAC-KBP and Web Text

TAC-KBP (Text Analysis Conference, Knowledge Base Population track, organized by NIST) is a research competition where the task is to extract common properties of people and organizations (e.g., age, birthplace, spouses, and shareholders) from a few million newswire and web documents — this task is

also termed Slot Filling. In the 2014 evaluation, 31 U.S. and international teams participated in the competition, including a solution based on DeepDive from Stanford. The DeepDive-based solution achieved the highest precision, recall, and F1 score among all submissions.¹

Conclusions and Future Work

The DeepDive project is continuing as an academic effort, with support from DARPA and others. A number of the people behind DeepDive have also created a company, ClearCut Analytics, with the goal of applying information extraction methods to novel problems in finance, government, and other industries. ClearCut has already identified several compelling opportunities in turning commercial dark data into useful structured data for business intelligence. [Q](#)

Michael Cafarella, Ph.D., is an Assistant Professor in the division of Computer Science and Engineering at the University of Michigan. His research interests include databases, information extraction, data integration, and data mining. Cafarella has published extensively in venues such as SIGMOD, VLDB, and elsewhere. He received the NSF CAREER award in 2011. In addition to his academic work, he co-started (with Doug Cutting) the Hadoop open source project, which is widely used at Facebook, Yahoo!, and elsewhere. Cafarella received his Ph.D. from the University of Washington, Seattle.

FOOTNOTES

¹ Other real-world uses and applications can be found at deeplive.stanford.edu.

REFERENCES

¹ See Table 6 in Mihai Surdeanu and Heng Ji. Overview of the English Slot Filling Track at the TAC2014 Knowledge Base Population Evaluation. Proceedings of the TAC-KBP 2014 Workshop, 2014.

CAN AI MAKE AI MORE COMPLIANT?

Legal Data Analysis Ex Ante, In Situ, Ex Post

By Bob Gleichauf and Joshua H. Walker

The Problem

A number of senior Intelligence Community (IC) officials describe compliance as one of the IC's biggest problems, perhaps the *biggest*. The underlying legal and informational issues are bound to become more acute and complex. How can artificial intelligence (AI) help?

To answer this question, we need to understand two things: data rights and application uncertainty. Data rights are data attributes derived from laws and dependent institutional policies. Data rights include but are not limited to classifications, access policies, source limitations, privacy constraints, etc. While such data rights are entailed in the data itself, the interpretation and application of these rights are contextual and will vary. More specifically, application of laws on a data set may be indeterminate: they may vary by time, user, and/or geography; the Second Circuit may issue an unexpected, divergent opinion; access may occur before or after a seminal FISA decision; interpretation of law and policy evolves and changes with time; or the legal state of a data set at the time of collection may be indeterminate.

Background

The IC protects our nation by analyzing the relationships between people, places, and things — essentially “connecting the dots.” Doing so while remaining compliant with policies such as Executive Order 12333 and Presidential Policy Directive 28 is a balancing act. The interpretation, implementation, and enforcement of policy vary across organizations and administrations. This frequently leaves analysts struggling to determine what data they can and cannot see. The Internet, mobile, and big data generally further complicate the problem. The sheer volume, velocity, and variety of data that is constantly generated necessitates automation, and even AI, to manage. However, the benefits of analytic automation over the data deluge will remain limited until the IC finds a way to scale the processing of legal judgments at a comparable rate.

We are dealing with two fundamental complexities here:

Data Complexity: The net complexity of the effective big data environment is increasing at a non-linear rate.

Legal Complexity: The median complexity and *ex ante* indeterminism of the policy environment pertaining to any given IC analysis event (automated or not) is static or increasing. Point-legal complexity is likely increasing in part due to data heterogeneity and speed/scale. More importantly, it is impossible to predict all beneficial and harmful access scenarios (or even the types of analysts that qualify) on data intake. It is partly indeterminate. Again, the data rights entailed in any data set are complex, as are the application of those data rights on access.

A complex feedback loop has developed between these two areas that is increasing the risk of disrupting important IC analysis. This “breakage” may occur because: (a) analysis was *or becomes* unlawful; (b) lawfulness of access is unprovable, generating toxic or inefficient policy controls on the IC; and/or (c) analysis was *prevented*.

Ten years ago there were virtually no products or communities handling real world legal automation. Since then, there has been a “Cambrian explosion” of startups and serious academic projects in the field. In the Bay Area alone, this has included Lex Machina, SIPX, Judicata, Casetext, Occam, CodeX (Stanford), and many others; not to mention CMU, Harvard, and myriad other efforts. The combination of increasing digitization, AI, and pent-up demand for legal innovation has finally spurred development of practical, enterprise-grade legal technology solutions and models. The federal judiciary uses them, not just technology companies. There is now a robust, rapidly evolving field of scholarship and innovation called Legal Informatics. The IC should be allowed to begin experimenting with these innovations.

There are also opportunities to learn from other policy-driven solution spaces such as finance

(Gramm-Leach-Bliley Act, Sarbanes-Oxley Act), healthcare (HIPAA), and more generally software-defined networks (SDNs). All of these ecosystems have found tractable, if imperfect, ways to apply policy that strike a balance between policy compliance and operational efficiency.

Baking Law Into Data Notation

Analytic Data Rights

From an analyst's perspective, the application of data rights essentially boils down to, "What am I allowed to look at?" For example, an analytic process might reveal that an individual who was born in the U.S., but has lived overseas for most of his life, appears to be a key participant in a terrorist cell. Some of the policy questions the analyst may face include:

- Can I query/search against data that may include U.S. persons?
- Can I access this data?
- Can I use the data to drive subsequent analysis?
- Can I share this data with other analysts; and if so, which ones?

In order to properly answer these type of questions we need to adopt a data model that tracks the lifecycle and transformations of data — from collection through subsequent query, access, analysis, annotation, dissemination, and commingling ("recollection") with other data. Policy and variant data rights may apply to any or all of these analytic processes. Automation can obviously complicate the application of policy. But, more specifically, how can it be used to reveal latent legal and intelligence issues? How can automation catalyze existing analytic opportunities and create new ones, while preventing illegal or otherwise undesired access? A data rights model provides a foundational, scalable framework for addressing such issues.

The process of engineering abstract law into efficient technological data access controls can be operationally challenging, especially given that law and policy are dynamic. In many instances different people (e.g., lawyers, programmers, and compliance officers) are responsible for each policy process. The associated tools also tend to be hard to use and the approval process can be drawn out. As a result, the model tends to be slow to react to change and reduces analytic agility. Achieving and demonstrating compliance can also consume immense amounts of an organization's time and resources. We cannot afford for the model to be out

of step with the realities of the world it is supposed to represent. It must be responsive — as well as accurate, efficient, and scalable.

Policy Application

The primary concern here is that filtering intermediate results in an analytic sequence may introduce information loss that could undermine pattern discovery. For example, it may prove difficult to determine that A knows C via B if a policy causes B to be obscured too early in the query process. The ability to make these connections is essential in a data environment focused on *model discovery* because it is difficult (if not impossible) to predict how data will be consumed by subsequent queries — especially when the query path may be an ad hoc analytic sequence (i.e., an acyclic graph). In other words, applying policy controls to a query or query sequence may prematurely filter information that may be relevant to subsequent queries.ⁱ While the rate at which this occurs will vary across data sets and analytic environments, this type of information loss will increase with scale.

The National Research Council of the National Academies appears to have come to the same basic conclusion in its recent response to Presidential Policy Directive 28.¹ In its report, the council stated that:

1. Bulk data collection is required to discover unknown patterns as premature filtering of data increases the risk of missing patternsⁱⁱ, and
2. In lieu of creative data filtering techniques, the USG should develop automated controls for the usage of bulk data.

In other words, policy application matters. More often than not, this comes down to when and how policy is applied to query results. What follows is a brief explanation of these two factors.

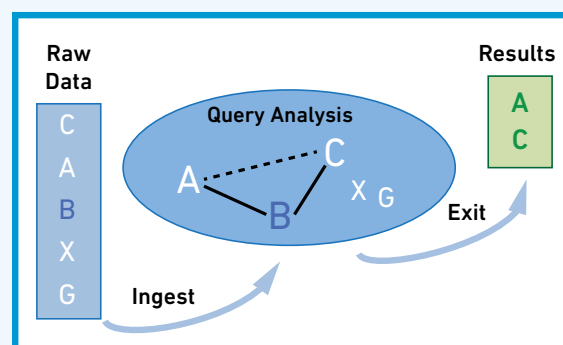


Figure 1

When?

Policies today are largely filters expressed as predicates or code. They transform and reduce data in order to comply with rules. A simple example would be, "Omit all records that contain U.S. persons' data." While filter methods may vary (e.g., obscure, encrypt, remove, deny) there are four stages when they typically apply: at *Rest*, *Input*, *Query and Analysis*, and *Output*.ⁱⁱⁱ

How?

The way queries operate within an automated analytic system is a key question. More specifically, "Are automated queries subject to the same policy constraints as an analyst?" Subject matter experts within the IC tell us that automated queries are currently subject to the same policy controls as manual queries issued by analysts. While filtering results accessed by an analyst makes sense, it could have a negative impact when applied to intermediate results in a series of automated queries.

On-Access vs. On-Query

The primary concern here is that analytical results may vary depending on how policy is applied. In particular, an analyst may obtain different results from a multipart query or analytic sequence when policy is only applied to results accessed by an analyst (on-access) versus uniformly applied to intermediate results generated by each sub-query (on-query). Moreover, applying policy earlier in an analytic sequence increases the risk of being unable to connect the dots. In AI terms, deterministic logic rules imposed on capture (i.e., *ex ante*) inevitably eliminate some "true positive" analytic conclusions (legal analysis of IC value) and increase "false positive" events (non-legal or inefficient access). Risk managers will inevitably seek to reduce embarrassing breaches through stronger, broader

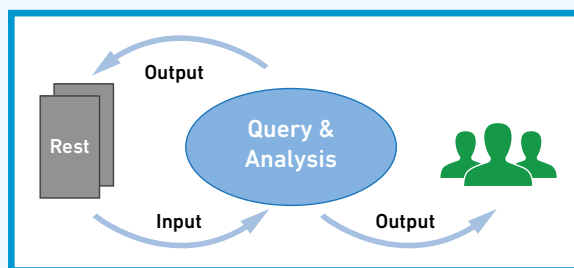


Figure 2

logical policies. But this is a blunt instrument; knocking out potential true positive analysis that could save lives or otherwise improve IC outcomes in future.

Figure 3 demonstrates the application mechanics. The hypothetical query sequence consists of five queries labeled A, B, C, D, and E. Query E generates results based on the output from a nested series of queries (((A+B)C+D)E).^{iv} While policy filters are applied to C's and E's output when accessed by analysts, they are not applied when results are passed between queries.

Another issue to consider is that policies asserted earlier in a query sequence must be broad in scope in order to cover all possible end states. This is very difficult to do in large systems — especially in an automated analytic environment where query sequences may be nondeterministic and the potential for information loss is much higher. Again, broadly scoped, pure logic-based policies imposed on capture increase the risk of missed connections by prematurely eliminating available data.

On-Query and On-Access: A Simple SQL Example

The following SQL queries show how the placement of a filter (citizenship EQ "U.S.") can impact the results of a sequence of queries against a simple toy data set. Experiments involving more complex query sequences and data sets are required to understand more fully the impact of these alternate filter methods.

RAW-PLAYER-DATA TABLE

PLAYER	CURRENT-TEAM	PAST-TEAMS	CITIZENSHIP
P1	Cubs	Braves	US
P2	Mets	Yankees, Cubs	Cuba
P3	Yankees	Mets, Dodgers	US
P4	Mariners	NULL	US

```
CREATE VIEW all-players SELECT *
FROM raw-player-data;
```

```
CREATE VIEW us-players SELECT *
FROM raw-player-data WHERE citizenship EQ "US";
```


ON-QUERY RESULTS: {null set}

```
SELECT player FROM all-players
WHERE current-team IN
  (SELECT past-teams FROM us-players);
/* inner on-query filter*/
```

ON_ACCESS RESULTS: {P1, P3}

```
SELECT player FROM all-players
WHERE current-team IN
  (SELECT past-teams FROM all-players)
  AND citizenship EQ "US";
/* outer on-access filter */
```

An SDN Analogy

Another way of explaining policy application is by means of analogy. Software-defined networks are a good place to start. SDNs build upon Access Controls Lists (ACLs), a ubiquitous form of policy control within TCP/IP networks. They are omnipresent on switches, routers, and firewalls. Historically ACLs have been applied on ingress (the entry points to a network) to keep unwanted traffic out of the network core. They also help minimize the potential for network denial-of-service (DoS) attacks. A major side effect of this approach is that the ACLs accumulate in an ad hoc manner. This can end up making them, and thus the network overall, difficult to maintain. Ingress filters require administrators to understand and keep abreast of network topology and business rules for the entire enterprise. This is difficult to do as a business ebbs and flows. An administrator in New York cannot reasonably be expected to understand what is going on in Seattle. As a consequence, it is not uncommon to find routers that contain thousands of ACLs that few, if any, know how to maintain. Administrators are reticent to change anything other than the last hundred or so ACLs for fear of breaking something. These types of network configurations are commonly referred to as brittle.

SDNs evolved to address this brittleness by migrating the relevant ACLs to the network exit points (egress). The number of egress ACLs ends up being a fraction of what are required using ingress filters. This makes access control policy more manageable because network administrators only need to know what is going on at their sites, not the entire global network. Egress ACLs also

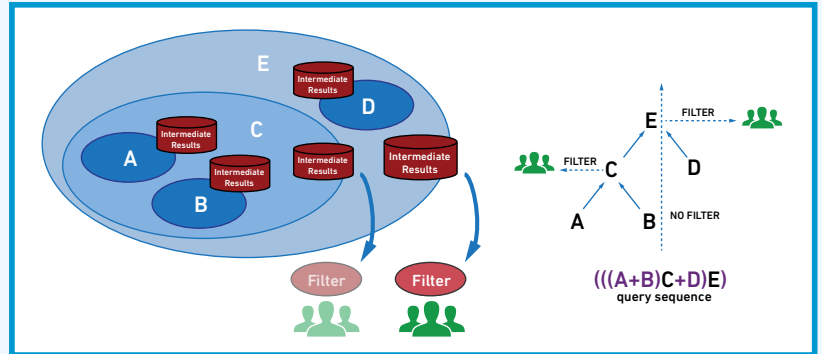


Figure 3

make the networks more resilient to ongoing changes in topology and business operations. Administrators only need to manage rules that apply to their business and partners. This approach is sometimes referred to as a Communities of Interest model and it has gained favor, as network cores have been able to deal with heavier traffic loads. This type of egress filtering is functionally equivalent to the proposed on-access policy controls.

Data Rights as Computational Notation

As noted earlier, policy is typically represented by a text-based document, formal declarative logic, and/or executable code. Regardless of the form it takes, policy has a many-to-many relationship with data: multiple policies may apply to a given set of data and multiple data sets may be subject to a policy.² From a data modeling perspective, this means that metadata describing collected data contains policy bindings, frequently instantiated as labels. These bindings ultimately translate into some type of filter or access control (typically an Attribute Based Access Control, or ABAC).

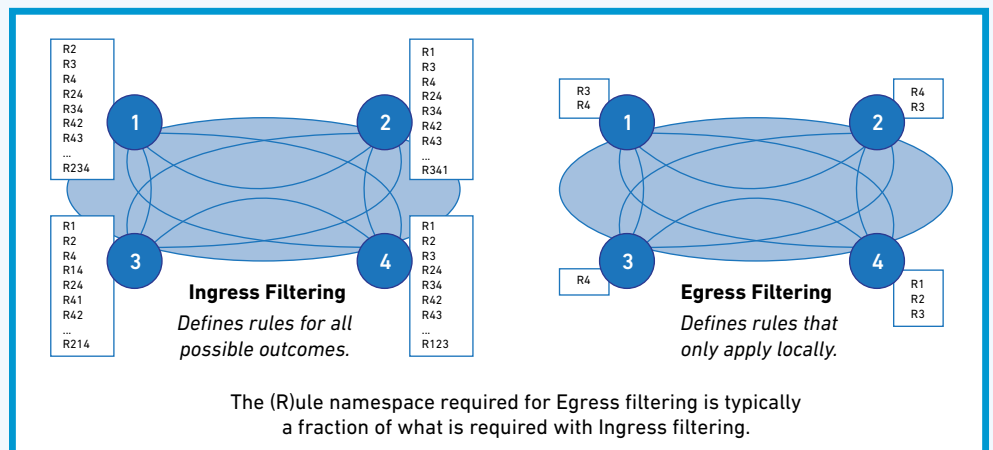


Figure 4

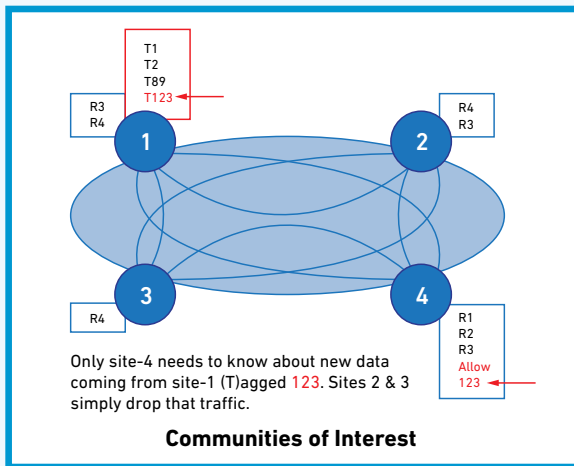


Figure 5

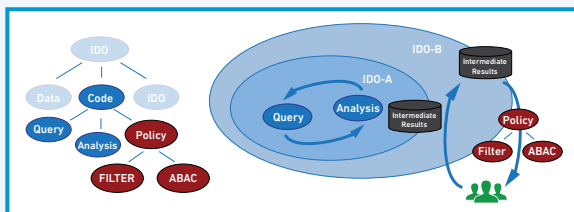


Figure 6

Lab41 uses an object-oriented Intelligent Data Object (IDO) construct to model metadata. The IDO data model makes it possible to annotate raw data collections with references to code as well as data (or text). The

assumption being made here is that by combining code and data references, we can build semi-automated analysis into the metadata layer of a database system (or federated metadata namespace). Just as with text, analysts can create code annotations — that is, references to code they choose to archive for later reference. In the first instance, these code references may make it easier for analysts to leverage each other's work. Over the long term, they may provide the basis for automating pattern discovery and insight. However, they can only do that if such automated or semi-automated discovery is compliant.

Lab41 is using challenge experiments to test various aspects of the IDO model and automated analysis. This preliminary work has made it apparent that policy is yet another form of annotation that is functionally equivalent to code. Integration of these types of policy controls may be a fairly straightforward extension of the existing IDO model.

At the beginning of this paper we stated that the interpretation and application of data rights may vary based on a variety of factors. Binding policy to data at the metadata level establishes the necessary relationships to assess data rights within a given context. This in turn lays the foundation for higher-level AI functionality that is compliant as well as insightful. Q

Bob Gleichauf is an Executive Vice President at In-Q-Tel and Director of IQT's Lab41, a unique, Silicon Valley-based challenge lab that provides "innovation through collaboration" in the area of big data analytics. Gleichauf joined IQT from Cisco Systems, where he spent a decade working on the development of secure network infrastructures across a variety of the company's products. Gleichauf, who has more than a dozen patents in network security, served as CTO for the Wireless and Security Technology Group at Cisco, and is respected globally for his work in information security. He previously served as head of product engineering for the WheelGroup prior to its acquisition by Cisco. Earlier, he was with IQ Software, a leader in the development of database report writing tools. Before making the leap into technology, Gleichauf pursued a Ph.D. in Early Human Prehistory at the University of Michigan, where he earned a fellowship and had the privilege of working in East Africa with the celebrated Leakey family.

Joshua H. Walker is a legal informatics entrepreneur, and an IP partner at a major law firm, handling all aspects of IP strategy and transactions. Walker has built his career at the nexus of law and computer science. Historically, as an analyst, his work has included helping prosecutors convict orchestrators of the 1996 Rwandan genocide to, now, as an attorney, helping many of the largest and most dynamic technology and financial entities in the world improve IP and data rights outcomes in the M&A, open source/licensing, and strategic litigation contexts. To help clients solve IP governance, transactional, and risk management problems, Walker co-founded the first law and computer science lab in the country (CodeX), at Stanford University, as well as the top "big data" company for IP litigation (Lex Machina; founding CEO & Chief Legal Architect). Walker received his J.D. from the University of Chicago Law School, and an A.B. in Conflict Studies (Special Concentrations) from Harvard College, m.c.l.

FOOTNOTES

- ⁱ It remains to be seen if this concern holds true for large, multipart queries as it does for a series of smaller queries.
- ⁱⁱ This determination was made based on the timing of events. It should apply to other data dimensions, such as geo-location.
- ⁱⁱⁱ Within the context of this analysis Bulk Data Collection is viewed as yet another form of Query and Analysis.
- ^{iv} This is just one example of a generalized model where membership, sequence, etc. may be nondeterministic.

REFERENCES

- ¹ https://www.nsa.gov/civil_liberties/_files/BulkCollectionofSignalsIntelligenceTechOptions.pdf
- ² <http://nvlpubs.nist.gov/nistpubs/specialpublications/NIST.sp.800-162.pdf>



The *IQT Quarterly* examines trends and advances in technology. IQT has made a number of investments in advanced analytics technologies, and several companies in the IQT portfolio are garnering attention for their unique technologies.



Digital Reasoning Systems

Digital Reasoning Systems's machine learning platform, Synthesys, identifies threats, risks, and opportunities by transforming information into a private Knowledge Graph. The company was recently featured in a *Wall Street Journal* article about analytics companies whose technologies are used by government and financial customers. Digital Reasoning joined the IQT portfolio in December 2010 and is based in Franklin, Tennessee.



Expect Labs

Expect Labs is the creator of MindMeld, a cloud-based service capable of powering intelligent assistants for any app, device, or website. Companies use the platform to create voice-driven assistants that understand what users say and automatically find the information they need before they type a search query. Expect Labs was recently featured in a *Forbes* article about deep learning technologies that have greatly improved the accuracy of speech recognition on mobile devices. The company is based in San Francisco and has been a part of the IQT portfolio since December 2013.

www.expectlabs.com



Narrative Science

Narrative Science is the leader in automated narrative generation for the enterprise. The company's Quill platform uses AI to identify the most relevant information in data and create conversational narratives for customers including *Forbes*, USAA, and financial services firms. The company recently released a book, *Practical Artificial Intelligence for Dummies, Narrative Science Edition*, to educate readers on the current state and future of AI. Narrative Science is based in Chicago and joined the IQT portfolio in March 2013.

www.narrativescience.com



Skytree

Skytree's machine learning platform gives organizations the power to discover deep analytic insights, predict future trends, make recommendations, and reveal untapped markets and customers. The company recently built a machine learning model with 10 billion rows and 100 columns, and ran and completed an industry-first 1 trillion element benchmark using a sophisticated GBT model. Skytree became an IQT portfolio company in June 2013 and is located in San Jose. www.skytree.net

