Musings

RIK FARROW

Rik is the editor of *;login:*.
rik@usenix.org

While I was at OSDI in Atlanta (2016), I heard a couple of paper presenters mention the use of machine learning (ML) as part of their research for those papers. I was immediately intrigued because ML, part of artificial intelligence (AI), seemed much too complicated to simply drop in as part of a graduate research project. So I set out in search of someone who could write an ML article for beginners.

I first asked Mihai Surdeanu [1], whom a fellow attendee at OSDI suggested I contact. Surdeanu considered the possibility of writing an article but backed out because he was already too busy. He suggested a list of names in the machine learning field. I tried Andrew Ng, the person who developed the famous Coursera class on machine learning, and of course he was too busy. Then I tried his course, but soon found myself flummoxed by the math. I had taken three years of advanced calculus in college, but that was 40 years ago. And I hadn't taken the class on linear regression, which turned out to be as important as matrix manipulation early in Ng's course.

Mihai comforted me, saying that Ng's class is for people designing new ML algorithms and that most people just use ready-made ones. That got me thinking: perhaps I could read some books and learn enough about ML to be dangerous.

The first book I found was *Machine Learning in Python* by Michael Bowles [2]. I also found a PDF of the book online, along with the code examples and data. That was good because I was in a hurry.

Bowles' book focuses on just two classes of ML algorithms: penalized linear regression (e.g., elastic net and lasso) and ensemble methods (e.g., stacking and gradient boosting). Bowles has an easy-to-follow writing style, and it's in Chapter 2 of his book where I learned about the importance of understanding the data you want to use for training data for your ML model. If you use bad data, your model will turn out to be useless. Your data also determines which class of algorithm is most appropriate. Bowles spends a lot of time performing statistical analyses of his data sets, helping me understand the importance of this step.

I began to understand why no one—and yes, I had asked more than just the two people mentioned here—wanted to attempt to write a "beginners guide" to ML. There really is a lot involved.

Mihai suggested a newer book, *Deep Learning with Python* by Francois Chollet [3], saying that Chollet uses newer algorithms than the Bowles book, which is two years older. I think I will continue with Bowles for now, then move on to Chollet.

I came out of my ML adventure realizing just how important training ML data is. Garbage-in, garbage-out was an old programmer saying, and it's just as relevant today as it was when I learned it decades ago. The goal of ML is to produce a model that can be used to predict output given new data. The ML algorithms build these models through a recursive process, refining, for example, weights for features that guide the decision process.

Stayed tuned, as I don't plan on giving up yet. I've heard there will be an extreme need for programmers who understand ML in the near future…

The Lineup

When I heard about the Cyber Grand Challenge (CGC) during USENIX Security '16 in Austin, people there suggested watching a long video. I was too impatient to ever get very far into the video. Other articles about CGC were rehashes of press releases: all alike without much info. But David Brumley, CEO of ForAllSecure and, through CMU, mentor to the hacking team PPP, had a participant's first hand perspective. His team not only won the CGC, it was the first autonomous computer system ever to compete in a DefCon Capture the Flag, making him the ideal person to describe the contest; ForAllSecure's Mayhem program was up against the best human hacking teams at DefCon and did very well.

Travis McPeak had also given a talk at Enigma, but I wanted more details and so engaged Travis in an interview about how they handle least privilege at Netflix. Netflix uses AWS, which in turn provides extensive configuration for controlling permissions. Travis makes points about mistakes with granting permissions, problems with setting up permissions when an app is first deployed, and how to automatically remove permissions from apps over time.

I also interviewed Swami Sundararaman, one of the co-chairs of the new HotEdge workshop (associated with USENIX ATC '18). I wanted to know *why*, suddenly, the edge is hot, and *what* this edge consists of. Turns out that, of course, things change, and there are new demands best met by services that will live at the *edge* of the Internet.

I was strolling through the FAST '18 posters and came across some familiar faces talking about something that just seemed too weird. Philip Kufeldt and his associates from many different storage companies were introducing *eusocial* devices. Like ants or bees, one device alone can't do much. But together, they can accomplish a lot. In this case, they want to take over data management to help unload CPUs and the memory pathways from work that could be handled by the eusocial storage devices.

Haryadi Gunawi et al. had presented a very cool paper at FAST about research into slow failures. When memory, or a networking or storage device, fails quickly, the failure is obvious. But when the failure is partial, such as a slowdown but not death, uncovering the failed component is much harder, particularly in current layered architectures.

Sean Kamath at LISA17 said he wanted to write another article about LISA. His first, "Whither LISA" [4], written in 2010, disturbed marketing folks but did stir up discussion. The LISA conference is undergoing more changes this year [5], and Sean has written a retrospective of his 25 years of attending LISA, how things have changed, and why they will be different in the future.

Mac McEniry continues to evolve the remote execution example with the addition of command-line processing and execution of multiple modules on the server side. Mac introduces Cobra, a Go package that is more flexible than the native Golang interface to the command line.

David Blank-Edelman takes a look at using graph databases from Perl. He focuses on Neo4j, explaining how graph databases work through examples, then covers who to do this via the REST::Neo4p module.

Dave Josephsen, as excited as ever, interviews Matt Broberg, the VP of Community for Sensu Inc. Sensu is an open source monitoring system that by design is very flexible and is already popular. After introducing Sensu 1.x, Dave moves on to asking about Sensu 2.0, written in Go, and how that will change things for Sensu users.

Dan Geer and Michael Roytman write about recall and precision, terms defined in their article. Their interest lies in how a security practitioner finds the crucial nuggets among all the events being generated by their security monitoring applications. They show, through a hypothetical example, how an organization couldn't possibly uncover all the significant events, and suggest how security software should be improved. Apart from the work reflected in this column, Michael turns out to have a very amazing project [6].

When I mentioned my interest in machine learning, Robert Ferrell wanted to participate too. Robert focuses on some of the failures of the not-so-smart machines around his house.

Mark Lamourine has reviewed three books this time. The first is *Teach Yourself Go in 24 Hours*, a Sams book that Mark was favorably impressed by. The second seemed a bit of a stretch for our community, but Mark came away from reading *Crucial Conversations* feeling there was a lot to learn there. Mark also read a relatively thin tome, *Linux Hardening*, and while he yearned for a bit more depth, concurs that the author succeeds by following a narrow path.

While I've been interested in AI and ML for several years now, a propaganda video about the dangers of autonomous killer drones (little tiny ones) really motivated me. We can expect to see more autonomous weapons going forward: just search for *autonomous paintball sentry gun*. People have been building these for years, and the South Koreans have taken their version a bit more seriously, replacing the paintball guns with machine guns.

But these are a far cry from little drones that can recognize their targets, with each drone working cooperatively, so that two drones don't gang up on one target, then approaching and killing the victim. The video rather loses the point that you can swat a little drone from the air with your hand, and certainly would do so before it got close enough. When current image recognition software still has problems describing what's in a photograph, forget about drones recognizing specific faces, much less cooperating during an attack.

The more common concern these days about AI is what is termed *general intelligence*. General intelligence is what humans are supposed to be able to manage: flexible behavior in varying circumstances, the ability to communicate intelligently, and so on. If you note a bit of cynicism here, just consider world politicians today. General intelligence in AI is supposed to lead to the extinction of humanity.

General intelligence in AI is as far off today as fusion power. Instead, AI works best when trained to operate in a particular, narrow field: for example, playing Go or providing medical advice; the game-playing Watson has moved on [7]. And to be honest, I'd much rather believe in a far-off future like the one created by Iain M. Banks in his Culture series. While Banks' AI machines (called *minds*) have godlike intelligence, they choose to continue working with humans—for reasons we might consider inscrutable.

Today, our AI still struggles with speech comprehension, and our autonomous paintball sentries don't even need AI. Instead, we need to focus on what ML can do to help us to understand the mountains of data we are acquiring every day.

References

[1] Mihai Surdeanu: http://www.surdeanu.info/mihai/.

[2] M. Bowles, *Machine Learning in Python* (Wiley, 2015), ISBN: 978-1-118-96174-2.

[3] F. Chollet, *Deep Learning with Python*: (Manning, 2018), ISBN-13: 978-1617294433.

[4] S. Kamath, "Whither LISA," *;login:*, vol. 35, no. 1 (February 2010): https://www.usenix.org/system/files/login/articles/102-kamath.pdf.

[5] LISA18: https://www.usenix.org/conference/lisa18.

[6] A. Maxmen, "Out of the Syrian Crisis, a Data Revolution Takes Shape," *Nature*, October 25, 2017: https://www.nature.com/news/out-of-the-syrian-crisis-a-data-revolution-takes-shape-1.22886.

[7] "IBM's Watson AI Recommends Same Treatment as Doctors": https://futurism.com/ibms-watson-ai-recommends-same-treatment-as-doctors-in-99-of-cancer-cases/.

Letter to the Editor

Rik,

In your Spring 2018 *;login:* column you asked about references to disk (controllers) taking over block/sector placement. The first I heard of this was in the early 1990s; after a bit of hunting I located a paper that would correspond to what I remember:

Robert M. English and Alexander A. Stepanov, "Loge: A Self-Organizing Disk Controller," in *Proceedings of the USENIX Winter 1992 Technical Conference,* pp. 237–251.

I don't remember who I heard present this when I was working for DEC Networks AD.

I found the paper online at http://stepanovpapers.com/Loge.USENIX.pdf (it appears to be just a little too old to be in USENIX's online archive, and I haven't found an entry for it in ACM's portal).

Cary Gray

Rik responds:

That's an amazing find. And I'm guessing that lots of people don't know that Hewlett-Packard manufactured disk drives in the past.

Although there are no links from the USENIX site yet, old proceedings are being digitized and hosted on archive.org. You can find the full Proceedings of the Winter 1992 Annual Technical Conference here: https://archive.org/details/winter92_usenix_technical_conf.

Thanks!
Rik

Register Today!

usenix
THE ADVANCED
COMPUTING SYSTEMS
ASSOCIATION

2018 USENIX Annual Technical Conference

JULY 11–13, 2018 • BOSTON, MA
www.usenix.org/atc18

The 2018 USENIX Annual Technical Conference will bring together leading systems researchers for cutting-edge systems research and the opportunity to gain insight into a wealth of must-know topics, including virtualization, system and network management and troubleshooting, cloud and edge computing, security, privacy, and trust, mobile and wireless, and more.

The program includes a Keynote Address by Dahlia Malkhi, *VMware Research*, 76 refereed paper presentations, a poster session, Birds-of-a-Feather sessions (BoFs), and more.

Co-located with USENIX ATC ’18

HotStorage ’18

10th USENIX Workshop on Hot Topics in Storage and File Systems
July 9–10, 2018
www.usenix.org/hotstorage18

Researchers and industry practitioners will come together for this two-day workshop on the cutting edge in storage technology and research and explore and debate longer-term challenges and opportunities in the field.

HotCloud ’18

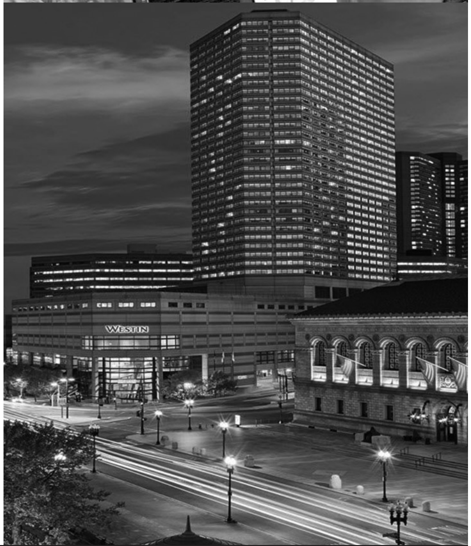
10th USENIX Workshop on Hot Topics in Cloud Computing
July 9, 2018
www.usenix.org/hotcloud18

HotCloud brings together researchers and practitioners from academia and industry working on cloud computing technologies to share their perspectives, report on recent developments, discuss research in progress, and identify new and emerging trends in this important area. While cloud computing has gained traction over the past few years, many challenges remain in its design, implementation, and deployment.

HotEdge ’18

USENIX Workshop on Hot Topics in Edge Computing
July 10, 2018
www.usenix.org/hotedge18

Join researchers and practitioners from academia and industry to discuss work in progress, identify novel trends, and share approaches to the many challenges in design, implementation, and deployment of different aspects of edge computing.



Register by June 19 and save!