

Landon Rabern

November 26, 2019

As a child, I was obsessed with machine intelligence. I coded a strong chess AI (codenamed Betsy) and experimented with using neural networks in Betsy; both for the static evaluation at leaf nodes and within the tree for pruning. The networks learned from self-play to get about as good as my hand-tuned functions (discounting the slowdown incurred by sigmoid evaluation). I concluded that to do better, I would need to use raw game state data instead of the set of features I preselected as network inputs; unfortunately, this was 2000 and I did not have nearly enough processing power to do so. To avoid spinning my wheels while waiting for faster hardware, I decided to put my machine intelligence ambitions on hold and study mathematics instead. My first year as a math major was spent abroad at Utrecht University in the Netherlands. My Dutch was pretty poor, but luckily all the math textbooks were in English. Learning to learn from just a textbook was painful at first, but this method of learning has been indispensable ever since. In 2005 I felt the pull of making real things and left my math phd program (where I was researching fun things like quantum groups) to work for a defense contractor. We simulated nuclear weapons effects like damage caused to military hardware by electromagnetic pulse. This work was interesting, but for family reasons, after about a year, I moved to Colorado and took a job for a subsidiary of Goldman Sachs. The Goldman Sachs job was not particularly exciting and I had lots of downtime. I had just read the book *The Man Who Loved Only Numbers* about Paul Erdős—a homeless math genius that traveled around for decades to work on hard problems with other discrete mathematicians. I decided to use my work downtime to teach myself graph theory and see if I was smart enough to play with these discrete mathematicians. This was in 2007. It took three years and a lot more than just my work downtime, but in 2010 I made a breakthrough by proving a conjecture of two prominent graph theorists (Alexandr Kostochka and Hal Kierstead). I made arrangements with Kierstead to come out to Arizona and finish my phd. By early 2018, I had satisfied my curiosity about playing with Erdős’s friends and achieved the best possible Erdős number of 2 (since Erdős no longer lives). Neural networks were all over the news, rebranded ‘deep learning’, researchers were having great success; hardware was now fast enough and there were massive amounts of data to play with. It was past time for me to get back to my childhood obsession. Back in 2010, I had started a software contracting firm, LBD Data, with a couple friends. Our final contract was winding down in 2018, so I started reading the machine learning literature and got a job at a health data analytics company IQVIA to learn data science. After a year, I moved

to the Facebook World.AI team in Cambridge to learn from new problems and people. A year at Facebook was plenty, so I moved to an AI research lab, Performance Star, in California. This journey has made it obvious to me that to solve AI/machine learning problems in a generic reusable way, I need to understand why the human neocortex works so well. Jeff Hawkins and Numenta have made progress with their Hierarchical Temporal Memory, but there are still many missing pieces. Artificial cortex is the future of computer science.