

# Trend in Computing - Big Data, Small Devices

Terry Yin

September 21, 2014

In year 1997, The IBM supercomputer Deep Blue beat the World Chess Champion Garry Kasparov in a human-computer chess match, [Wikipedia \(2014a\)](#). After that, supercomputers have been consistently outperforming the human top chess players. In year 2009, a software program named Pocket Fritz 4 running on a mobile phone reached the grandmaster level and achieved the same performance as IBM Deep Blue, [Wikipedia \(2014c\)](#).

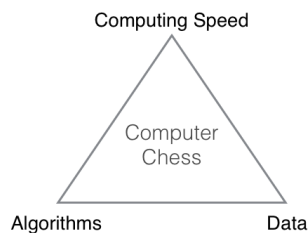


Figure 1: The 3 factors of computer chess performance

There are 3 important factors that decide the performance of a computer chess player, computing speed, algorithm and data, as shown in Figure 1.

Does the successful story of Pocket Fritz 4 mean that the computing power of a normal mobile phone in 2009 is already at the same level as the IBM supercomputer Deep Blue in 1997? No. the phone that run Pocket Fritz 4 is 10,000 times slower than Deep Blue. It only searches fewer than 20,000 positions per second.

Then is it because now we have much better algorithm for the same chess problem? The algorithm has been improving all the time. Chess playing has got fairly large amount of attention from people studying computer science as stated by *Is chess the drosophila of artificial intelligence? A social history of an algorithm* [Ensmenger \(2011\)](#). From the same paper, we can see that the algorithm hasn't changed much until 2011. Some 20 years ago, I learnt the Minimax algorithm for the first time when I was in high school. Today, the essence of the algorithm is still the same. Still, people wouldn't stop finding better searching techniques and stronger killer heuristic. But this alone would not have led us this far.

So what do data do here in solving logically complicated computing problem?

Say, we need to check whether a number is a prime number. A slow but complete solution could be something like:

```
def isprime(n):  
    for i in range(2, n):  
        if n % i == 0:  
            return False  
    return n != 1
```

But what about the following solution?

```
def isprime(n):  
    return n in {2, 3, 5, 7, 11, 13, 17, 23, 29, 31, 37,  
                41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97}
```

Some will call the second one cheating if it's a test in an Algorithms class. But the two solutions are

effectively the same if the teacher only verify the result with ( $n < 101$ ). And the second solution is a lot faster. This simple example shows that data and algorithms are tradable to some degree. We don't have always to model the logic in the real world but "letting the data speak for themselves" Gould (1981). And often new opportunities emerge when big data is involved. One example is from the speech recognition domain, as stated in Google's report Chelba et al. (2012).

Coming back to the story of computer chess. The secret ingredient is also data. Modern computer chess programs all use large opening books and endgame books. Now small devices can also have large storage, and they can also utilize the data in the "cloud". According to Levy & Newborn (1991), computer gains about 50-70 Elo points by doubling the speed. Elo point is a measurement for playing strength in chess. In 2012, there were 5839 human players at the "Candidate Master" (2200 to 2299 Elo points) level in the world Wikipedia (2014b). There were 3 human at the top having 2800 points. So for a Candidate Master level computer chess program, it has to be 7 to 12 times faster to become the top level master. The famous Moore's law Wikipedia (2014d) stated "computer speed doubles every 18 months." Then we will have to wait 18 years to see it beats all the human masters! We can't wait that long. Big data comes to the rescue!

## Conclusion

Just like human brain is not just a logic machine, we actually solve problems quite often by pattern matching, big data is another important dimension in solving problems in computer science. The big question left here is "how". My prediction is the exploration of how to better exploit the big data will continue to be a trend in computing.

## References

- Chelba, C., Bikel, D., Shugrina, M., Nguyen, P. & Kumar, S. (2012), Large scale language modeling in automatic speech recognition, Technical report, Google.
- Ensmenger, N. (2011), 'Is chess the drosophila of ai? a social history of an algorithm', *Social studies of science* p. 0306312711424596.
- Gould, P. (1981), 'Letting the data speak for themselves\*', *Annals of the Association of American Geographers* **71**(2), 166-176.
- Levy, D. N. & Newborn, M. (1991), *How computers play chess*, Vol. 8, Computer Science Press.
- Wikipedia (2014a), 'Deep blue versus garry kasparov — wikipedia, the free encyclopedia'. [Online; accessed 19-September-2014].  
**URL:** [http://en.wikipedia.org/w/index.php?title=Deep\\_Blue\\_versus\\_Garry\\_Kasparov&oldid=623303193](http://en.wikipedia.org/w/index.php?title=Deep_Blue_versus_Garry_Kasparov&oldid=623303193)
- Wikipedia (2014b), 'Elo rating system — wikipedia, the free encyclopedia'. [Online; accessed 20-September-2014].  
**URL:** [http://en.wikipedia.org/w/index.php?title=Elo\\_rating\\_system&oldid=624923344](http://en.wikipedia.org/w/index.php?title=Elo_rating_system&oldid=624923344)
- Wikipedia (2014c), 'Human-computer chess matches — wikipedia, the free encyclopedia'. [Online; accessed 19-September-2014].
- Wikipedia (2014d), 'Moore's law — wikipedia, the free encyclopedia'. [Online; accessed 20-September-2014].  
**URL:** [http://en.wikipedia.org/w/index.php?title=Moore%27s\\_law&oldid=626293387](http://en.wikipedia.org/w/index.php?title=Moore%27s_law&oldid=626293387)