Grand-master Level Chess without Search Modeling Choices and their Implications

A naive reader (and I am such) may conclude the following:
DeepMind demonstrated a machine learning model which satisfies all of:

1. Learned and is able to follow the rules of chess
2. Learned and is able to play the game at a grandmaster level
3. Learned purely from observing chess games and their outcomes
4. Learned using a purely-ML system, without any game specific biases encoded in the training procedure

However, **none of these points are true**! And to be fair, the paper doesn't claim them. It is well written, and doesn't over-claim (except for claiming a resulting system with no search... we'll get to that). Upon some more careful reading and thinking, one can actually realize that:

- Points (1) and (2) were not achieved, and in fact are not possible at all with the learning setup used in the paper.
- Moreover, point (1) and (2) are likely not possible in *any* setup that follows points (3) and (4).
- Points (3) and (4) are not supported by the setup used in the paper.

Procedure

1. The researchers took 10M human-human chess games
2. For each board position and legal move they asked the Stockfish 16 engine for its estimate of the "winning probability" up to 0.05 seconds per move.
3. They then trained a transformer model with action-value as: position+move
    - **Input** is board position, **output** is the winning probability of a move.
    - **Target output** is the winning probability of the move

That's it. Only model predictions, no looking-ahead search.

Human learning:

Let's assume a hypothetical extremely dedicated professional human player, who studies chess 16 hours a day, and is able to analyze a game per minute, 365 days a year. Such a player will go over 10M games in about 28.5 years. If this player takes not 1 minute but 5 minutes to analyze a game, they will need 142 years to go over these 10M games. There are grandmaster-level chess players who are 13 years old. So clearly human learn differently and much more efficiently.

Is the model really learning by observing? (Hint: No)

The paper mentions that the model could not follow the "don't repeat the same board position three times in a row" rule (aka "threefold repetition").

- By restricting the input to a single board position, the *model designer* is using *their own knowledge of the game* to *inject game knowledge* to the learning process: the knowledge that previous moves are irrelevant in most situations.

In cases where the position of one player is much stronger than the others, the "winning probability" can be 100% for several different moves: each of them will lead to winning the game down the line. However, some moves will achieve victory using fewer future moves than others. In our case, the model cannot choose between these moves: it only knows to maximize winning probability of a single move, it does not even "know" that the move will be followed by future moves, doesn't know how winning will be achieved, and doesn't even known what it means to win.

- It has no notion of "a plan"

Consider, for example, the model having two rooks, while the opponent has a king. This is a textbook example of check-mating, so every move will be a guaranteed 100% probability of winning. However, in order to actually win, one has to make a specific series of moves with their rooks, not just move them aimlessly around. But the board is large enough to allow rooks moving aimlessly around. This will be an absurd behavior, associated with very weak chess players, and which will never ever happen at grand-master level. Yet, the model, as trained in this paper's setup, has no way of avoiding it, and has a real chance of exhibiting it.

Learning from the Backseat Driver

The setup in the paper is not learning from observations. Rather, it looked at a game position, then used a strong chess-engine to expand a tree of possible game continuations from this position spanning hundreds of thousands if not millions of moves into the future. The game engine then used its internal knowledge of chess to assess the percentage of winning board positions within the possible continuations, which is the "probability of win" for a position. Then, the learner tried to learn *this* number.