논문 리뷰

Mastering the game of Go witout Human Knowledge

SEUNGWOO LEE (이 승우)

처음의 논문은 딥마인드 社의 논문인 "Mastering the game of Go Without Human Knowledge" 이였지만, 한번 구현해 보기로 기조를 잡아서 몇가지의 논문을 리뷰하게 되었다.

이번 "강화학습을 이용한 바둑 두기" 에서는 다음 4 가지의 논문을 리뷰해 보고, 향후 실제로 적용시켜 볼 수 있도록 해볼려고 한다.

1. Mastering the game of Go without human knowledge. David Silver 외 16 명. 네이쳐지. doi:10.1038/nature24270
2. Playing Go without Game Tree Search Using Convolutional Neural Networks. Jeffery Barratt 외 1 명. Stanford University. cs231n.stanford.edu/reports/2017/pdfs/603.pdf
3. Deep Residual Learning for Image Recognition. Kaiming He 외 3 명. Microsoft Research. arXiv:1512.03385
4. Training Deep Convolutional Neural Networks to Play Go. Christoper Clark 외 1 명. arXiv:1412.3409

▶ 서론

지금까지의 인공지능은 지도학습을 통하여 사람의 결정을 복재하는 수준 이였다. 그러나 질좋은 데이터를 구하기 위해서는 시간과 비용이 많이 든다. 비용이 적거나 없는(무료인) 데이터를 구하더라도 데이터의 품질이 좋지 않을 수 있다. 이런 문제를 극복하기 위하여 강화학습을 사용하게 되었다. 강화학습 안에서 일어나는 데이터를 가지고 학습하기 때문에 데이터가 필요가 없다. 이 기법을 사용하게 되면 단 시간 안에 사람보다 더 좋은 성능을 내는 것을 만들어 낼 수가 있다. Atari 라는 게임이나, 3 차원의 가상현실 게임에서 좋은 성과를 내었다. 이처럼 인공지능이 좋은 성과를 내게 되면서 인간과 바둑을 두는 것과 같이 인간의 지성과 경쟁을 하게 된 것이다.

알파고는 첫 번째로 인공지능이 인간을 바둑에서 뛰어넘은 프로그램이다. 첫 번째 알파고 판(유럽 바둑 챔피언 판 후이와 바둑을 겨뤘을 때 사용한 프로그램이다.) 프로그램에서는 두 개의 뉴럴 네트워크를 가지고 학습을 하였다. 규칙(Policy) 망과 가치(Value) 망을 가지고 학습을 하게 되었다. 규칙망은 인간의 바둑 데이터를 통하여 학습을 하여 바둑을 둘 자리를 확률적으로 결정한다. 가치망은 앞서 규칙망이 둔 바둑을 통하여 이 게임의 승패가 어떻게 될 지를 예측하는 역할을 한다. 이 둘개의 네트워크가 학습을 하게 되면 몬테-카를로 탐색 트리와 같이 연동되어 앞을 내다보는 검색을 하게 된다. 규칙망에서 바둑돌의 위치를 두게 되면, 가치망에서 둔 바둑돌을 확인한 후에 몬테-카를로 탐색 트리를 수정하게 된다.

알파고 제로는 알파고 판(판 후이)과 알파고 리(이 세돌)버전과 몇 가지가 다르게 설계 되었다. 앞서 인간이 둔 게임을 토대로 학습을 하는 것과 달리, 이 버전에서는 기계가 랜덤으로 플레이한 데이터를 토대로 학습을 하도록 바뀌었다. 또한 검은색 돌과 하얀색 돌의 위치만이 입력 값으로 들어가게 되었다. 그리고 기존의 두 가지(규칙, 가치)의 네트워크를 사용한 것과 달리, 이번에는 하나의 뉴럴 네트워크를 사용하게 되었다. 마지막으로 앞서 사용했던 몬테 카를로 나무 탐색 알고리즘에서 벗어나 간단한 나무 탐색 알고리즘을 사용하게 되었다.

▶ 알파고 제로에서의 강화학습

알파고 제로에서는 강화학습이라는 기법을 사용해서 학습을 하였고, 기존에 두개의 가치망과 규칙망을 하나의 뉴럴 네트워크로 통합하였다. 그리고 이 뉴럴 네트워크는 배치 정규화와 렐루(RELU)함수로 이루어진 수많은 Residual 블록 들로 이루어져 있다. 뉴럴 네트워크라고 지정하는 f_θ를 지정하고, 여기에서 S라고 하는 값을 넣어주게 된다. S라고 하는 값은 바둑돌의 위치와, 그 전까지의 바둑돌 위치를 넣어주는 값이다. 이 값들이 f_θ에 들어가게 되면, 출력값으로 p와 v를 출력해주게 된다. p는 착수포기(영어로는 Pass 라고 한다.) 값을 포함하여 Move a 파라메터를 아웃풋으로 준다. $P_a = \Pr(a|s)$ 라고 정의할 수 있는데, 이는 $방향_{행동} = 확률(행동|위치)$이라고 정의할 수 있다. 모든 a의 행동이 실행되면, 몬테 카를로 나무 탐색이 실행된다. 탐색이 실행되면 효율적인 바둑돌의 위치를 찾아서 Π_n의 형태로 출력해주거나, 후에는 z의 형태로 승부를 알려주게 된다.

이 과정을 통해서 알파고 제로는 36시간만에 알파고 리(이세돌)를 36시간만에 뛰어넘을 수 있었다. (알파고 리는 학습하는데 몇 달이 소요되었다.) 그리고 72시간 후에는 알파고 제로가 알파고 리(이세돌)을 100 게임 모두 이겼다.

아울러 알파고 리(이세돌) 버전까지는 48개의 단일 TPU(Tensor Processing Unit)을 사용했으나, 알파고 제로는 4개의 TPU를 사용하여 학습 효율을 달성하였다.

그래서 일련의 결과로 인해서 강화학습을 이용하여 자가학습한 알파고 제로가, 전까지 지도학습을 한 알파고 리(이세돌)를 포함한 버전보다 더 빠르고 더 좋은 결과를 낼 수 있게 되었다.