Seoul National University
Course 4190.408.001
Artificial Intelligence
Project 1

Due date: April 19, 2019

Student ID: 2019-90142
Name: Hyun Wook Park

Project 1 요약

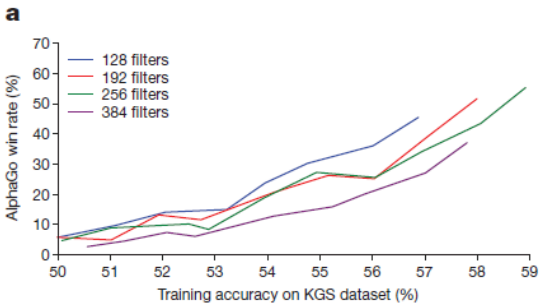
SL, RL Policy Network 와 Value Network의 분석, CNN과 MCTS 분석을 통하여 각각의 기술이 AlphaGo 학습에 미친 영향 분석, 이후 발전된 신기술 동향 (Google Duplex) 조사

• SL policy network 분석

AlphaGo의 SL policy network (Supervised Learning of Policy Network)는 Simple CNN (Convolutional Neural Network)으로, 프로 바둑기사들의 착수 전략을 학습하는 방법이다. *[Mastering the game of Go with deep neural networks and tree search]* 에서 제시한 아래 식에서 $p_\sigma(a|s)$에 대하여 살펴보면, s는 시간이 t일 때의 바둑기보이며, 입력 값이다. a는 시간이 t+1일 때의 바둑기보이며, 출력 값이다. 이 네트워크는 Classification network이기 때문에, 순차적일 필요는 없기에 모든 (s, a)에서 무작위로 Data Sampling을 하여 SGD(Stochastic gradient descent)로 Learning하게 된다.

$$\Delta\sigma \propto \frac{\partial \log p_\sigma(a|s)}{\partial \sigma}$$

또한, 이 방법은 사람이었을 때, 다음 수를 두는 경향을 Modeling 한 것이다. 50개의 GPU를 사용하여 학습(기간: 3주, 3억 4천번의 학습과정)하였다. 13 layer의 CNN을 사용했으며, 프로 6단에서 9단 사이의 실제 대국 16만개의 기보로부터 3000만 가지 바둑판 상태를 추출하여, 이 중 약 2900만 개를 학습에 이용하고, 나머지 100만 가지 바둑판 상태를 시험에 사용하였다(정확도 57%). 기존 최신기술 프로그램의 44%의 정확도 보다 무려 13%의 정확도가 높으므로 경이롭다고 할 수 있다. 또한, 정확도가 높아질수록 AlphaGo의 최종 Winning rate가 상승한다는 사실이 실험결과로 나와있다. 정확도가 높으면 Winning rate가 상승한다는 사실을 아래 <Figure 1> 에서 나타내고 있다.



<Figure 1>

• RL policy network 분석

AlphaGo 의 RL policy network (Reinforcement Learning of Policy Network)는 SL policy network 와 동일한 구조를 가지고 있으며, 스스로 대국하여 지도학습을 강화하는 방법이다.

지도학습의 결과로 구해진 Policy network는 사람의 착수 선호도를 표현하지만, 이 정책은 반드시 승리로 가는 최적의 선택이라고는 볼 수 없다. 따라서 선택을 보완하기 위해서, 지도학습으로 구현된 Policy network 와 Self-play 를 통하여 결과적으로 승리하는 선택을 Reinforcement Learning 하게 된다. 이때, 처음에는 지도학습의 결과를 그대로 이용하여 경기를 진행하지만 학습이 진행되면서 여러 버전의 네트워크가 생성되며,

이들 간의 Reinforcement Learning 을 통해 실제로 승리하는 빈도가 높은 쪽으로 가중치가 학습된다. 약 128 번의 자체대결을 1 만 번씩 수행한다(총 128 만번).

학습방법은 현재 RL Policy network p_ρ 와 이전 Iteration 에서 사용했던 Policy network 중에서 무작위로 하나를 뽑아 둘끼리 서로 대국을 하게 한 후, 둘 중에서 현재 네트워크가 최종적으로 이기면 reward 를 +1, 지면 −1 을 주도록 디자인되어있다. 하지만 reward 는 대국이 끝난 시점의 T 에서의 reward 이지 현재 시점 t 에서의 reward 는 0 이기 때문에, 대신 네트워크의 outcome 을 $z_t = \pm r(s_T)$라고 정의한다.

$$\Delta\rho \propto \frac{\partial \log p_\rho(a_t|s_t)}{\partial \rho} z_t$$

다시 말해서, 이 네트워크의 outcome 은 현재 player 의 time t 에서의 terminated reward 가 된다. 이 네트워크 역시 Stochastic gradient method 를 사용해 Expected reward 를 최대화하는 방식으로 학습이 된다. 여기에서 과거에 학습된 네트워크를 사용하는 이유는, 좀 더 generalize 된 모델을 만들고, Overfitting 을 피하고 싶기 때문이다. 강화학습 후의 정책 네트워크로 기존 바둑 프로그램인 Pach 와 대결하여 85%의 승률을 보였다(MCTS 탐색 알고리즘 미적용).

• Value network 분석

AlphaGo 의 Value network 는 Value 를 계산하기 위해 Deep Learning 을 사용하며, 바둑의 전체적인 형세를 파악하는 방법이다. 인공신경망의 입력 층과 은닉 층 구조는 정책네트워크와 유사한 컨볼루션 신경망이지만 출력 층은 현재의 가치(형세)를 표현하는 하나의 값이 나오는 구조이기 때문에, 특정 게임 상태에서의 승률을 추정할 수 있다. Value network 는 evaluation 단계에서 사용하는 네트워크로, 현재 기보 s 와 Policy p 가 주어졌을 때, value function $v^p(s)$ 를 예측하는 네트워크이다. 즉, 다음과 같은 식으로 표현할 수 있다.

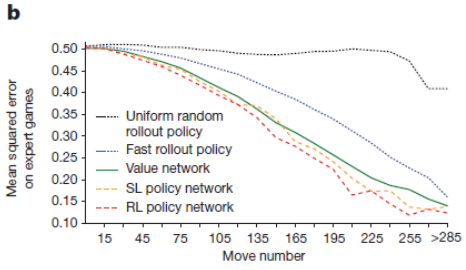
$$v^p(s) = \mathbb{E}[z_t | s_t = s,\ a_{t\ldots T} \sim p]$$

하지만 optimal value function $v^*(s)$ 를 학습할 방법이 없다. 대신에, AlphaGo 는 현재 시스템에서 가장 우수한 policy 인 RL policy network p_ρ 를 사용해 Optimal value function 을 Approximation 한다. Value network 는 앞에서 설명한 Policy network 와 비슷한 구조를 띄고 있지만, 마지막 출력 층으로 모든 기보가 아닌, Single probability distribution 을 사용한다. 이에 따라 문제는 Classification 이 아니라 Regression 이 된다. Value network 는 현재 가장 State−outcome pair 인 (s, z)에 의해서 학습이 된다 (z 는 RL network 에서 나왔던 최종 reward 의 값으로 1 또는 −1 이다).

$$\Delta\theta \propto \frac{\partial v_\theta(s)}{\partial \theta}(z - v_\theta(s))$$

따라서 Value network 는 s 에 대해 z 가 나오도록 하는 Regression network 를 학습하게 되며, Error 는 $(z - v_\theta(s))$가 된다. 그런데 문제는, State s 는 한 개의 기보인데, Reward target 은 전체 Game 에 대해 정의되므로, Successive position 들끼리 서로 강하게 correlation 이 생겨서 결국 Overfitting 이 발생한다는 것이다. 이 문제를 해결하기 위해 AlphaGo 는 3 천만개의 데이터를 RL policy network 들끼리의 자가대국을 통해 만들어낸 다음 그 결과를 다시 또 value network 를 learning 하는 데에 사용한다. 그 결과 원래 training error 0.19, test error 0.37 로 Overfitting 되었던 네트워크가, training error 0.226, test error 0.234 로 훨씬 더 Generalized 된 네트워크로 학습되었다는 것을 알 수 있다.

아래 <Figure 2>는 Random policy, Fast rollout policy, Value network, SL network 그리고 RL network 를 사용했을 때 각각의 value network 의 expected loss 가 plot 되어있다. 결국, RL policy 를 쓰는 것이 그렇지 않은 것보다 훨씬 우수한 결과를 낸다는 것을 알 수 있다.



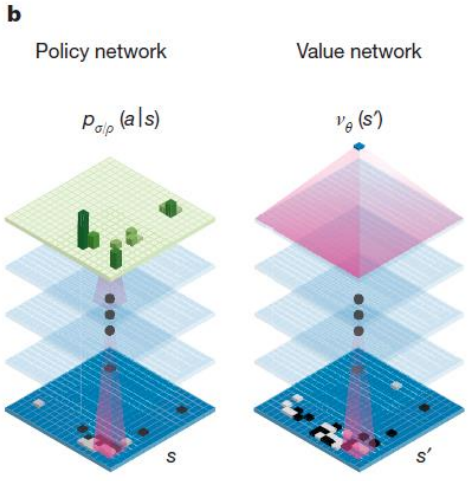
<Figure 2>

• 딥러닝 기술(CNN) 분석

CNN은 이미지나 비디오에서 객체의 분류에 특화된 방법이며, 이미지의 객체분류는 전통적인 인공신경망인 다층 퍼셉트론으로도 충분히 가능했으나, 노드간 링크가 모두 연결되어있는 구조(fully-connected)가 갖는 한계 때문에 그 대안으로 컨볼루션 신경망이 부상하였다. 이미지 처리(Image processing) 분야에서의 컨볼루션은 필터(커널)을 지칭하고, 이 컨볼루션 필터로 원본 이미지를 처리하여 특징을 추출해 낸다.

AlphaGo의 Policy Network에서 사용된 딥러닝 기법은 컨볼루션 신경망(Convolutional Neural Network, CNN)으로 19x19 바둑판 상태를 입력하여 바둑판 모든 자리의 다음 수 선택 가능성 확률 분포를 출력한다 <Figure 3>. 또한, CNN은 페이스북의 얼굴인식 기술인 DeepFace에 적용된 기술로 입력 이미지를 작은 구역으로 나누어 부분적인 특징을 인식하고 이것을 결합하여 전체를 인식하는 특징을 가진다. 여기서 컨볼루션 필터의 의미는 국소적, 지역적인 대국의 특징을 추출해내서 전반적인 형세를 추론하는 도구로 볼 수 있다.

바둑에서는 국지적인 패턴과 이를 바탕으로 전반적인 형세를 파악하는 것이 중요하므로 컨볼루션 신경망을 활용하는 것이 적절한 선택이다.



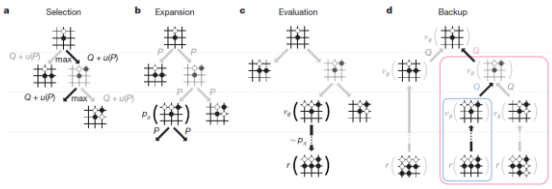
<Figure 3>

• 탐색방법(MCTS) 분석

착수를 결정하는 부분에는 몬테카를로 트리 탐색(Monte Calro Tree Search, MCTS) 기법이 사용됐다. MCTS 는 바둑 인공지능에서 가장 널리 사용되는 알고리즘으로 무한대에 가까운 탐색의 폭과 깊이를 줄이는 역할을 한다. 또한, MCTS 는 Tree search 를 Exact tree traversal 을 하는 대신, Random 하게 node 를 하나 고르고 Sampling 을 통해 확률적인 방법으로 Approximate tree search 를 하는 방법이다. 계속 반복하면 Asymptotic 하게 Optimal value function 으로 Converge 하게 된다.

탐색의 폭을 줄이는 것은 Policy로, 다음 수를 어디에 두는 것이 가장 좋은가에 대한 역할을 한다. 탐색의 깊이는 Value 값으로 정해지고, 이것은 현재 대국에서 승산을 근사적으로 표현한다. 따라서 MCTS의 성능은 정책과 가치의 정확도에 따라 좌우된다. AlphaGo에서는 이 정책과 가치를 Deep Learning으로 구현한 것이다. 바둑의 탐색 범위를 프로기사의 관점으로 좁힌 것이다.

MCTS 는 <Figure 4>과 같은 4 개의 Sequence 를 계속 반복하는 과정이라 할 수 있다.

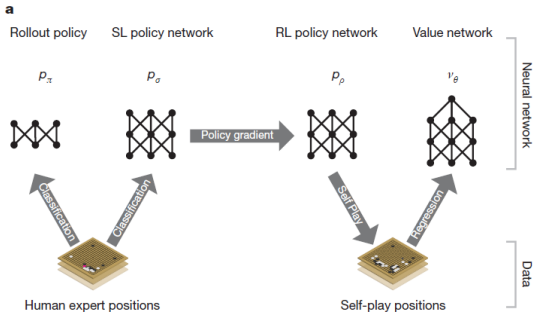


<Figure 4>

MCTS 에서 가장 중요한 핵심은 Tree Policy, 그리고 Default Policy 이다. Tree Policy 란 이미 존재하는 Search tree 에서 Leaf node 를 Select 하거나 Create 하는 Policy 이며, 바둑의 경우에는 특정 시점에서 가능한 모든 수 중에서 가장 승률이 높은 수를 예측하는 Policy 이다. Default Policy 는 주어진 nonterminal state 에서의 value 를 estimation 하는 policy 이며, 바둑의 경우에는 현재 상황에서 얼마나 승리할 수 있을지를 measure 하는 policy 이다.

Backup step 자체는 둘 중 어떤 policy 도 사용하지 않지만, 대신 Backup 을 통해 각 Policy 들의 Parameter 들이 Update 된다. 이 4 개의 Step 이 한 Iteration 으로, MCTS 는 시간이 허락하는 한도 내에서 이 과정을 계속 반복하고, 그 중에서 가장 좋은 결과를 자신의 다음 행동으로 삼는다.

• 각각 기술들의 AlphaGo 학습에 대한 종합적 서술



<Figure 5>

AlphaGo는 MCTS를 Deep learning pipeline을 통해 훨씬 성능을 개선한 것이라 할 수 있으며, <Figure 5>를 보면 알 수 있듯이, Network는 SL, RL 두 개의 Policy network 그리고 Value network 총 세 가지를 learning 하게 된다. Policy network는 MCTS의 Selection에서 쓰이게 되며, Value network는 MCTS의 Evaluation에서 쓰이게 된다.

다시 말해서, AlphaGo는 인공지능 알고리즘을 기존 MCTS 방식에 CNN을 통한 Deep Learning으로 수행하였고, 실제 바둑기사의 착수를 학습한 Policy Network와 국지적인 패턴인식으로 승산판단을 위한 Value Network로 방법을 구현하였으며, 프로바둑 기사들의 착수전략을 학습하는 Supervised learning 방식과 스스로 경기하여 학습된 전략을 강화하는 Reinforcement learning 방식을 적용하였다.

각종 기술을 융합한 AlphaGo의 성능은 Elo Rating으로 최대 3140으로 KGS기준으로 현재 프로 2단 ~ 5단 정도의 수준으로 파악되며, 기존 상용 바둑 프로그램들과의 경기에서 99.8%의 승률을 보인다.

• AlphaGo 논문 이후 발전 기술 조사

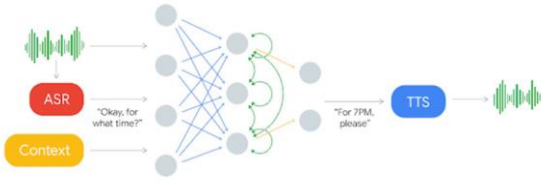
2018년 5월 8일, 구글은 사람 대신 전화를 걸어주는 인공지능(AI)을 공개했다. 구글의 연례 개발자 컨퍼런스 '구글I/O 2018'에서 공개된 새로운 AI 기술은 마치 사람처럼 자연스럽게 대화하는 모습을 보여준다. AI 비서 구글 어시스턴트가 직접 미용실에 전화해 직원과 대화하며 사용자 대신 예약을 해주고 쉬는 날을 확인하는 식이다.

'구글 듀플렉스'라고 불리는 이 기술은 기존의 대화형 AI보다 한 단계 진화한 모습을 보여준다. 일방적으로 묻고 답하는 방식에서 특정한 약속을 잡기 위해 능동적으로 사람과 자연스럽게 대화를 진행한다.

전화 대화를 AI 가 이해하고 반응하기 위해선 많은 과제를 극복해야 한다. 주변 소음으로 인해 음성인식이 어려울 수 있으며, 컴퓨터랑 대화한다고 인지되지 않을 경우 더욱 복잡한 문장이 사용된다. 또 사람의 자연적인 대화 습관은 AI 모델을 혼동시킨다.

구글은 지속적인 대화를 실제 사람과 유사한 보이스와 복합적인 행위로 구현하기 위해 여러 가지 기술을 적용했다. 아래 <Figure 6>는 듀플렉스의 단순화된 아키텍쳐를 나타낸다.

새로운 자연어 처리 기술(Natural Language Understanding), 신경망 기반의 Deep Learning, 문자를 음성화 해주는 TTS(Text-to-Speech), 그리고 가장 중요한 RNN(Recurrent Neural Network) 기술을 적용해 자연스러운 대화를 만들 수 있게 되었다. 익명의 전화 대화 데이터 말뭉치를 순환 신경망에 학습시켰다.



<Figure 6>

RNN 에 Input 으로 들어가는 Feature 는 3 가지 정도이다. (1) 대화 상대방의 음성 신호. (2) 1 의 음성 신호로부터 추출한 발화 텍스트. (텍스트 추출에는 구글이 개발한 ASR

(Automatic Speech Recognition)이 사용된다.) (3) 발화가 이루어진 환경, 문맥과 관련된 정보들. 이러한 Feature 들로 RNN(recurrent neural network)을 사용한다. RNN 은 인풋으로 들어온 상대방의 발화에 대해 AI 어시스턴트가 대답할 내용을 텍스트 형태로 출력한다.

RNN 에서 출력된 텍스트를 구글의 TTS(text-to-speech) 시스템이 음성 신호로 변환한다. 사람만큼 자연스럽게 말하게 하기 위해 딥마인드의 WaveNet 과 구글브레인의 Tacotron 을 사용하는 음성합성 엔진을 사용한다고 한다. 요약하자면 절차는 [음성신호, 발화내용과 문맥 -> RNN -> 대답 텍스트 -> TTS -> 대답 음성]이다.

더하여 실제 통화가 이루어지는 중에 사람이 개입해서 올바른 대화를 지시할 수 있도록 Real-time Supervised learning 을 사용했다고 한다.

AlphaGo 에서 발전하여 구글은 RNN 을 기반으로 Deep Learning, TTS 기술 등을 통해 이제 기계의 완벽한 자연어 처리(Natural Language Understanding)를 꿈꾸고 있다. AlphaGo 의 기술인 SL, RL policy network 와 Value network, CNN, MCTS 등은 참신한 기술이지만 이것은 한정된 프로세스 밖에 수행하지 못한다. 하지만 만약 완벽한 자연어 처리를 기반으로 한 기계의 학습, 수행 알고리즘이 융합된다면, 사람과 인공지능 로봇이 공존하게 되는 날도 멀지 않을 것이다.

Reference

1) **[Nature]** David Silver, Demis Hassabis et al. (2016). "Mastering the game of Go with deep neural networks and tree search"

2) **[SPRi]** 소프트웨어정책연구소 (2016). "SPRi Issue Report - AlphaGo의 인공지능 알고리즘 분석"

3) **[Google AI Blog]** (2016). "AlphaGo: Mastering the ancient game of Go with Machine Learning"

4) **[Google AI Blog]** (2018). "Google Duplex: An AI System for Accomplishing Real-World Tasks Over the Phone"