SPRi Issue Report
2017.01.26. 제2016-013호

인공지능의 핵심 인프라 -
고성능컴퓨팅 환경의 중요성

추형석 선임연구원†
안성원 선임연구원

SPRi 소프트웨어정책연구소

○ 본 보고서는 「미래창조과학부 정보통신진흥기금」을 지원받아 제작한 것으로 미래창조과학부의 공식의견과 다를 수 있습니다.

○ 본 보고서의 내용은 연구진의 개인 견해이며, 본 보고서와 관련한 의문사항 또는 수정・보완할 필요가 있는 경우에는 아래 연락처로 연락해 주시기 바랍니다.

- 소프트웨어정책연구소 SW융합연구실 추형석 선임연구원(hchu@spri.kr)

《 Executive Summary 》

제4차산업혁명은 인간의 지적노동을 대체할 수 있는 인공지능 기술의 급격한 발전으로부터 시작한다. 현대의 인공지능은 영상에서 객체를 인식하는 정확도가 이미 인간의 수준에 버금가고, 인공지능 바둑 프로그램이 세계 최고 챔피언을 꺾었으며, 두 개의 언어를 동시에 인식할 정도로 향상됐다. 이러한 배경에는 인터넷을 통한 빅데이터의 대중화, 저렴한 고성능 하드웨어의 보급, 공개 SW를 통한 공유의 문화가 확산 등 정보통신기술의 환경 변화에 있다.

그 가운데 인프라 역할을 한 것이 저렴한 하드웨어의 보급이다. 지난 60년간 인공지능의 부침의 역사를 살펴보면 컴퓨팅 환경이 항상 높은 장벽으로 존재해 왔다. 현실적인 문제를 해결하는 도구로써의 인공지능은 시간과 비용이 허락하는 한 수많은 시도를 해보는 것이 일반적이다. 따라서 고성능 컴퓨팅 인프라의 대중화는 인공지능 발전에 매우 중요한 역할을 했다. 컴퓨팅 환경의 중요성은 빅데이터를 처리한다는 점에 있어서 이미 공감대가 형성됐지만, 구체적으로 얼마나 필요하며 어느 정도 중요한지에 대해서는 분석해볼 필요가 있다. 이번 보고서에서는 인공지능의 연구사례와 글로벌 고성능컴퓨팅 동향을 분석하여 제4차산업혁명 시대에 컴퓨팅 환경의 중요성을 피력하고자 한다.

고성능컴퓨팅(High Performance Computing, HPC) 분야는 전통적으로 슈퍼컴퓨터를 효율적으로 다루기 위해 파생된 학문이다. 슈퍼컴퓨터는 실험이 불가능한 극한 자연현상의 시뮬레이션을 수행하는 부분에 본질적인 역할이 있다. 그러나 인공지능이 점차 현실적인 문제해결의 도구로 각광받음에 따라, 미국과 중국을 비롯한 HPC 선진국들은 인공지능 연구를 위한 컴퓨팅 환경 확보에 박차를 가하고 있다. 특히 캐나다와 유럽은 HPC를 플랫폼화하여 산학연을 아우르는 연구지원 체계를 갖추고 있다.

인간과 대결하기 위해 개발된 게임인공지능은 이미 슈퍼컴퓨터급 환경을 갖추고 도전했다. 최근 이세돌 9단과 대결한 AlphaGo는 개발 당시 300위권에 해당하는 슈퍼컴퓨터를 사용했다. AlphaGo의 인공지능 기술인 딥러닝은 행렬곱 연산으로 분해할 수 있고, 이것은 전통적으로 HPC의 성능을 십분 활용할 수 있는 것이다. 따라서 인공지능 연구를 진흥하기 위해서는 고성능 컴퓨팅 인프라가 필수적이고 가장 선결돼야 하는 과제다.

《 목 차 》

1. 연구 배경

□ 제4차산업혁명의 원동력은 인공지능으로, 현대 인공지능의 눈부신 성과는 고성능컴퓨팅(High Performance Computing, HPC) 환경이 핵심 역할

○ 사람의 지능을 모사하여 지적노동을 대체할 수 있는 인공지능은 현대 정보통신기술 발전의 결정체

- 그 저변에는 저렴한 컴퓨팅 하드웨어(HW)의 보급, 빅데이터의 대중화, 공개SW 활성화 등의 요인이 있음

- 특히 지수적으로 성장한 고성능 HW의 보급은 빅데이터를 효율적으로 처리할 수 있는 도구로, 그간 인공지능 연구의 고질적인 장벽이었던 계산 수요를 해소

* 빅데이터가 제4차산업혁명의 원유라면, HPC 환경은 원유를 정제하기 위한 인프라이며, 인공지능은 원유를 효율적으로 정제하기 위한 기술

○ 바둑 세계챔피온을 꺾은 인공지능 프로그램 AlphaGo는 슈퍼컴퓨터급 계산 환경 활용

- AlphaGo에 사용된 컴퓨터는 최대 1,920개의 중앙연산처리장치(Central Processing Unit, CPU)와 280개의 그래픽연산처리장치(Graphical Processing Unit, GPU)를 보유

* 이 컴퓨팅 환경의 연산처리능력을 수치화하면 세계에서 약 380위에 해당하는 슈퍼컴퓨터 (개발 당시 2015년 11월 기준)

○ 인공지능 연구에 연산 능력이 중요한 이유는 빅데이터를 활용한다는 관점에서 설득력 있게 다가오지만, 기술적인 측면에서 분석한 정량적 정보와 글로벌 동향을 토대로 그 중요성을 구체화 할 필요가 있음

□ 그동안 HPC의 본질적인 역할은 과학기술 시뮬레이션에 있었으나, 해외 동향을 살펴보면 인공지능 분야에도 컴퓨팅 수요를 반영

○ HPC는 우주 시뮬레이션, 나노단위의 물성 변화, 전 지구 시뮬레이션 등 물리적인 실험이 힘들거나 불가능한 자연현상을 모사하는 것이 주된 목적

○ 최근 인공지능에도 HPC의 수요가 급증하면서 제4차산업혁명의 핵심기술을 육성하기 위한 인프라로 재조명

○ HPC는 국가 R&D 능력을 대변하는 지표로서, 선진국은 보다 강력한 계산 자원을 확보하기 위해 국가차원에서의 확보 전략을 수립

- 미국은 지난 2015년 7월 엑사스케일컴퓨팅 프로젝트(ECP)[1]를 발족하여 2020년대 중반까지 약 10억 달러의 규모 투자

  * ECP에서도 향후 응용분야로 데이터 분석과 기계 학습에 활용할 계획

- 중국은 HPC의 신흥 강국으로 지난 3년간 세계 최고의 슈퍼컴퓨터 보유국으로 발돋움

  * 지난 2016년 6월 1등을 차지한 슈퍼컴퓨터 선웨이 타이후라이트(Sunway TaihuLight)는 중국자체기술로 이룩한 성과로 미국 기술 의존성을 낮춤

- 우리나라는 “국가 초고성능컴퓨터 활용 및 육성에 관한 법률”이 2014년 실행되어 차세대 슈퍼컴퓨터 도입과 초고성능컴퓨팅 기술 확보를 위한 연구개발이 진행 중

○ 최근 연산처리장치 제조업체들도 인공지능 연구에 최적화된 방향으로 HW를 설계

  * GPU의 세계 최대 벤더인 NVIDIA는 딥러닝(Deep Learning)[2] 전용 워크스테이션 DGX-1을 출시

□ 인공지능으로 대두되는 제4차산업혁명의 대비는 HPC 환경 확보가 선결돼야 하는 과제

○ 컴퓨팅 성능의 발전과정을 분석하여 현대 연산처리장치의 특성 고찰

○ 인공지능 연구의 성공사례와 AlphaGo의 인공지능 기술을 계산량 측면에서 분석하여 HPC 환경의 필요성 증명

○ 글로벌 HPC 동향을 통해 제4차산업혁명을 대비할 HPC 환경 확보 전략 파악

1) Exascale Computing Project, https://exascaleproject.org/exascale-computing-project/
2) 딥러닝은 인간의 뇌를 모사한 인공신경망에서 발전된 예측·분류 기법으로 최근 이미지 인식, 음성 인식 등에 활용되어 사람 수준과 가까운 성능을 보임

2. 고성능컴퓨팅 환경과 인공지능

(1) 컴퓨팅 성능의 발전과 이슈

□ 현대 연산처리장치의 성능

○ 스마트폰의 어플리케이션 연산처리장치(Application Processor, AP)의 이론성능은 약 20년 전 세계에서 가장 빠른 슈퍼컴퓨터 성능에 육박

Flop/s(초당 부동소수점 연산수) - 계산성능의 척도

- Flop/s 는 Floating-point operations per second 의 약자로 초당 부동소수점 연산수를 의미
  - 연산처리장치의 연산능력을 표현하는 지표로 슈퍼컴퓨터의 성능 비교 등에 사용
  - 알고리즘을 실제로 구현했을 때 필요한 연산수를 나타냄
  - 현대 슈퍼컴퓨터의 성능을 나타내는 단위는 테라플롭스(TeraFlop/s, 초당 1조), 페타플롭스(PetaFlop/s, 초당 1천조), 엑사플롭스(ExaFlop/s, 초당 1백경) 등이 있음
- 부동소수점(Floating-Point)
  - 연산 한 개의 기준은 보통 덧셈이나 곱셈 한 개이나, 최근에는 덧셈과 곱셈을 하나의 연산으로 간주
  - 수치적인 연산이 대부분 덧셈과 곱셈이 동시에 이루어지므로, 이를 통합한 기능이 연산처리장치에 탑재 (예, FMA[3])
    * 예를 들어, 100차원 벡터 두 개를 요소별로 더하는 알고리즘의 연산수는 100flop
  - 유효숫자(precision)에 따라서 성능이 구분
    * 16-bit half precision, 32-bit single precision, 64-bit double precision

- [그림 1]은 삼성 갤럭시 S7에 탑재된 Mail-T880 MP12의 이론성능은 265.2 GigaFlop/s, 1996년 6월 세계에서 가장 빠른 슈퍼컴퓨터[4]의 성능은 220.4GigaFlop/s

3) Fused Multiply-Add, https://en.wikipedia.org/wiki/Multiply%E2%80%93accumulate_operation
4) 도쿄대학의 Hitachi SR2201/1024, 당 시대 슈퍼컴퓨터의 도입비용은 약 5천만 달러

자료 : top500.org, Performance Development https://www.top500.org/statistics/perfdevel/
Samsung Exynos, Wikipedia https://en.wikipedia.org/wiki/Exynos

[그림 1] 삼성 갤럭시S7에 탑재된 GPU의 이론 성능과 슈퍼컴퓨터의 비교

○ 중앙연산처리장치(Central Processing Unit, CPU)의 성능은 냉각비용 대비 연산속도(Clock speed)가 한계에 다다름에 따라 점차 멀티코어(Multi-core)화 됨

- 고성능 CPU[5]의 경우 22개의 코어와 약 1.5 TeraFlop/s의 이론성능 보유 [그림 2]

○ 그래픽연산처리장치(Graphical Processing Unit, GPU)는 그래픽 출력에 필요한 연산을 수행하는 장치였으나, 성능이 점차 향상되어 고성능 컴퓨팅의 필수 도구로 진화

- GPU는 수 천 개의 연산코어[6]를 탑재한 매니코어 시스템으로 최신 GPU의 경우 약 10 Teraflop/s의 이론성능 보유 [그림2]

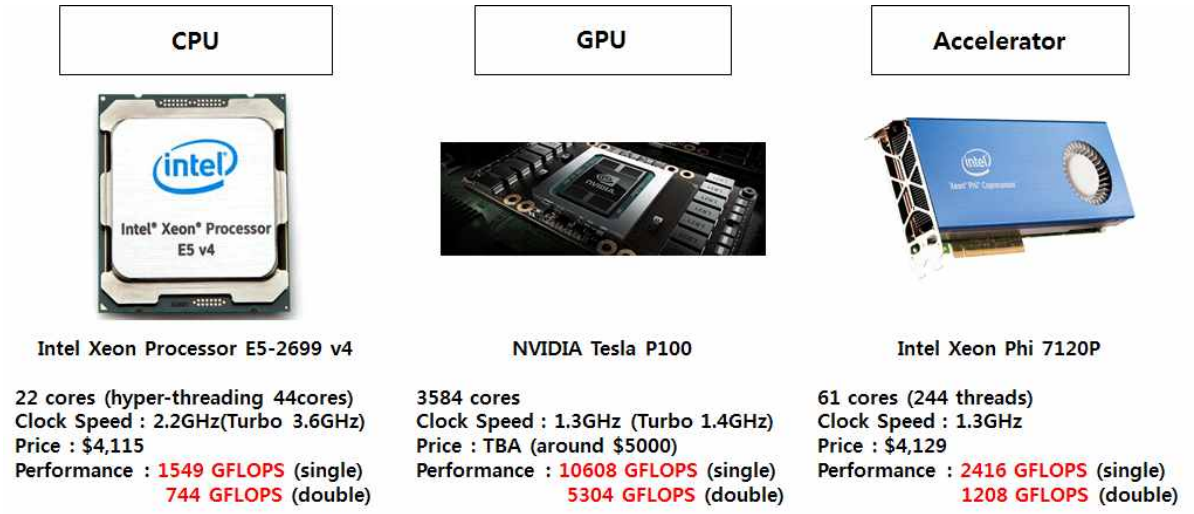
- 2008년 Nvidia가 C언어 기반의 GPU 프로그래밍 툴 CUDA(Compute Unified Device Architecture)를 공개함으로써 GPU 활용의 진입장벽을 낮춤

* GPU 프로그래밍은 SIMD[7] 기반의 대규모 병렬처리에 기반을 두고 있음

5) Intel Xeon Processor E5-2699 v4 (http://ark.intel.com/products/91317)
6) GPU 코어는 CPU의 물리적 코어와는 다르게 사칙논리연산유닛(Arithmetic Logic Unit)에 중점을 둔 RISC 모델
7) Single Instruction Multiple Data, 하나의 동일한 연산에 서로 다른 데이터를 처리하는 기법. 예를 들면, 100차원 벡터의 요소별 합 연산은 덧셈이라는 동일한 연산에 서로 다른 100개의 데이터를 처리

- AMD의 ATI 그래픽카드의 경우 병렬 컴퓨팅 표준 플랫폼 언어인 OpenCL (Open Compute Language) 활용

○ 그 밖에 Intel Xeon Phi, FPGA(Field Programmable Gate Array) 등 가속기(Accelerator)도 고성능 계산에 활용 중



자료 : Intel Xeon Processor E5-2699 v4, http://ark.intel.com/products/91317
Nvidia Tesla P100, http://www.nvidia.com/object/tesla-p100.html
Intel Xeon Phi 7120P, http://ark.intel.com/products/75799

[그림 2] 현대 계산자원의 성능

□ 최신 연산처리장치의 활용 측면에서의 이슈

○ 계산자원의 이론성능은 하드웨어적으로 측정된 값으로 실제 응용 프로그램에서는 달성하기 어려움

- 실행 환경, 알고리즘의 특성과 병렬화 가능성 등에 따라 성능 차이 존재
- 특히 메모리전송 대역폭과 계산 성능의 불균형으로 인해 같은 알고리즘이라도 다른 계산자원에서 현격한 성능 차이가 발생

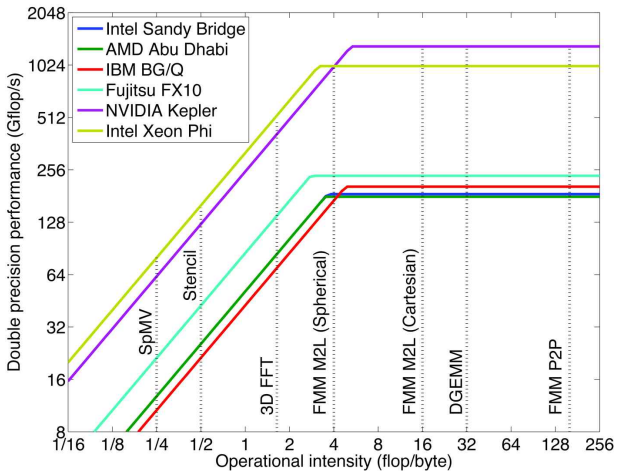
* 이 이슈는 폰노이만 연산처리장치 아키텍처 상에서 빈번히 발생하는 문제로 90년대 이전에는 메모리전송 대역폭이 계산 성능보다 우월했으나, 현대의 계산자원은 계산 성능이 비약적으로 향상되어 메모리전송 대역폭에 병목이 발생

○ 특히 계산자원의 멀티코어·매니코어화 기조에 따라 병렬컴퓨팅(Parallel Computing)이 필수 기반기술로 부상

함. 만약 계산자원이 10개의 컴퓨팅 코어를 보유하고 있다면 1개의 컴퓨팅 코어가 10개의 덧셈을 처리하는 방식.

- 병렬컴퓨팅 기술은 과거 클러스터 형태의 슈퍼컴퓨터를 활용하기 위한 것이었으나, 계산자원 자체가 병렬화 됨에 따라 현대 고성능컴퓨팅의 핵심 기술
- 알고리즘의 병렬화는 크게 데이터 기반(data-parallel)과 태스크 기반(task-parallel)으로 나뉘는데 GPU와 같은 매니코어 시스템에서는 대부분 데이터 기반을 주축

○ 루프라인(Roofline) 모델 - 알고리즘의 메모리전송량과 계산량의 비율로 달성 가능한 최대 성능(Roofline)을 표현



자료 : How will the fast multipole method fare in the exascale era? (2013.07)

[그림 3] 다양한 계산자원의 루프라인

- 연산강도(Operational Intensity)는 알고리즘의 총 연산수와 메모리전송량의 비율을 나타내며, 계산강도가 높을수록 전송량대비 계산량이 많은 것을 의미
- 앞서 기술했듯이 현대 계산자원은 메모리전송 대역폭보다 계산성능이 우월하므로 루프라인이 선형적으로 증가하는 부분은 메모리전송량이 계산량보다 많은 경우를 나타냄

* 메모리전송에 소요되는 시간동안 연산처리장치가 유휴상태(idle)에 처하기 때문

병렬 계산의 효율성

Q. 12개의 원소를 갖는 두 벡터의 합을 구하는 연산이 있다고 가정하자. 만약 4개의 코어를 갖는 연산처리장치를 활용한다면 단일 코어 대비 4배의 성능향상을 달성할 수 있는가?

A. 반드시 4배의 성능향상을 보장할 수 없다. 위 연산의 연산강도를 계산해보면 약 $1/16^{8)}$으로, [그림 3]에서 추정할 수 있듯이 메모리전송량에 의해 최대 성능을 달성하기 어렵다.

(2) 인공지능 사례 분석

□ 인공지능 연구에는 막대한 규모의 계산이 소요

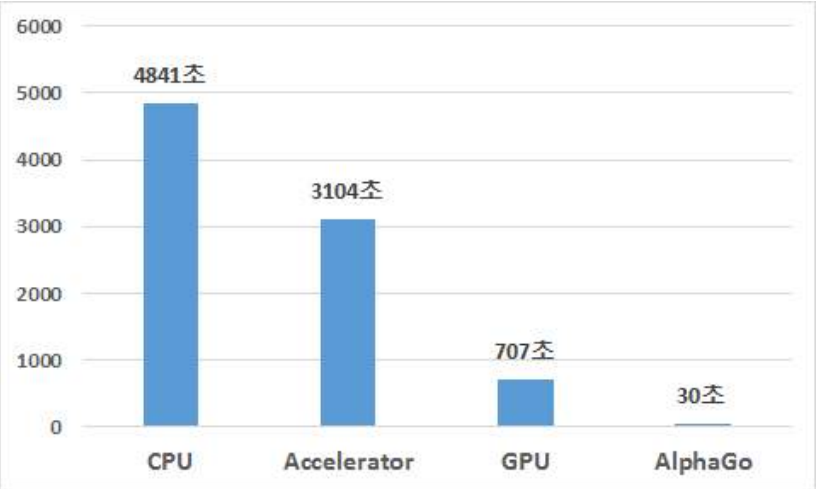
○ AlphaGo의 딥러닝은 한 번 연산하는데 약 30 GigaFlop이 필요[9)]

- AlphaGo의 컴퓨팅 환경은 바둑 게임의 초읽기 30초 동안 약 10만 개의 경우의 수를 탐색

- 30초 동안 계산해야 하는 연산수 6 PetaFlop (6천조 번 연산)

* 딥러닝 알고리즘에 국한된 연산수. 10만 개의 경우의 수를 탐색하기 위해 필요한 딥러닝 연산은 20만 번으로 총 연산수는 6 PetaFlop

○ [그림 2]의 연산처리장치를 기준으로 6 PetaFlop을 처리하는데 소요되는 시간은 다음 [그림 4]와 같음



[그림 4] 계산자원별 소비시간 비교

- [그림 4]에서 볼 수 있듯이 인공지능 연구에는 HPC환경이 필수

* 계산자원별 성능은 이론성능의 80%로 계산. (AlphaGo환경은 176개 GPU)

* 딥러닝 알고리즘은 연산강도가 높기 때문에 확장성(Scalability)이 보장됨

8) 위 문제를 수식으로 나타내면 $z = x + y$, 여기서 x, y, z는 12개의 원소를 갖는 벡터임. 12개의 원소를 갖는 실수형 벡터의 메모리 크기는 4byte×12 = 48byte. 계산에 필요한 벡터 x, y와 결과 값을 저장하는 벡터 z가 계산을 위해 연산처리장치의 레지스터에 전송되고, 결과 값 z를 다시 내려 받아야 하기 때문에 4번의 12차원 벡터 전송이 발생. 따라서 메모리전송량은 총 192byte인 반면 계산량은 12번이기 때문에 연산강도는 1/16

9) AlphaGo의 연산량에 관한 자세한 내용은 후술

□ 게임 인공지능 연구에 활용된 컴퓨팅 환경 [부록 참조]

○ 그동안의 대표적인 인공지능 연구 성과는 슈퍼컴퓨터급 성능을 갖는 컴퓨팅 환경으로부터 시작

- 개발 당시의 컴퓨팅 환경은 top500.org[10] 기준으로 약 2~300위 수준. <표 1> 참조

- CPU의 성능이 지속적으로 향상됨에 따라, Deep Blue와 같이 전용 체스칩에 의존하던 경향에서 CPU 중심의 클러스터로 변화

  * 이후 AlphaGo에 이르러 GPU와 같은 가속기가 도입되면서 방대한 계산에 적합한 컴퓨팅 환경 도입

○ 학습에 요구되는 데이터의 양이 증가할수록 고성능 컴퓨팅 환경이 필수적

- IBM Watson은 초당 500 Gigabyte에 해당하는 정보를 처리할 수 있음

  * 500 Gigabyte는 약 백 만권의 책에 해당하는 용량

- AlphaGo는 16만 개의 프로 바둑기사의 기보를 수순에 따라 약 3천만 개의 바둑판 상태로 재구성하여 학습함

  * 콘볼루션 신경망의 입력 값인 바둑판 상태는 48가지의 특징으로 추출. 최종적으로 수치화된 데이터의 용량은 약 1.85 Terabyte

<표 1> 인공지능 연구와 컴퓨팅 환경

	IBM Deep Blue	IBM Watson	Google AlphaGo[11]
연산처리장치	30 노드 POWER2SC (120Hz) VLSI 체스칩 480개	90 노드 POWER7(3.5GHz, 8코어, 32쓰레드)	약 35~40노드 CPU 1,920 코어 GPU 280 장
연산성능	11.38 GigaFlop/s (VLSI 제외)	80 TeraFlop/s	약 300 TeraFlop/s[12]
top500 순위	259위 (1997 6월)	192위 (2011 6월)	382위 수준 (2015년 11월)

10) 세계 슈퍼컴퓨터 성능을 1위부터 500위까지 공개하는 웹사이트
11) AlphaGo의 경우 구글 클라우드를 활용하여 연산을 수행했기 때문에, 정량적으로 정확한 계측이 불

□ AlphaGo의 인공지능 알고리즘 - 콘볼루션 신경망

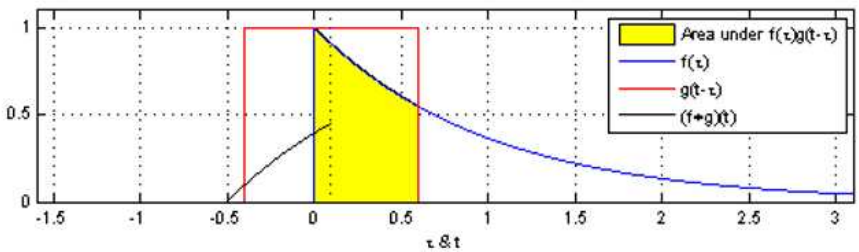
- AlphaGo에 적용된 딥러닝 기술은 알고리즘 측면에서 계산강도가 높은 **행렬곱 연산**이 대부분을 차지
- 행렬곱 연산은 **현대 HPC의 최대성능을 활용할 수 있는 것**으로 GPU를 비롯한 고성능 가속기의 효용성이 증대
- 또한 경험적인 지식이 중요한 딥러닝 기술은 다양한 시도가 매우 중요하므로, **계산을 수행하기 위한 인프라 확보가 제4차산업혁명을 대비하기 위해 반드시 선결돼야하는 과제**

○ AlphaGo의 인공신경망 구조는 13층의 콘볼루션 신경망

- 콘볼루션 신경망(Convolutional Neural Network, CNN)은 인공신경망 중에서도 이미지 분석에 특화된 방법론

  * Facebook의 얼굴인식 알고리즘은 9층의 콘볼루션 신경망으로 구성되고, 4백만 장의 얼굴 이미지를 학습하여 약 98%의 정확도 달성

- 콘볼루션의 수학적 의미는 두 함수의 합성을 표현한 것으로, 일반적으로는 임의의 두 함수가 겹치는 영역을 적분한 값을 나타냄 [그림 5]



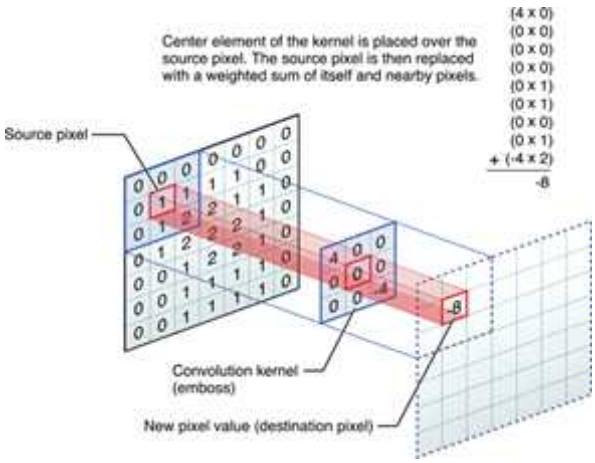
자료 : Convolution, https://en.wikipedia.org/wiki/Convolution

[그림 5] 콘볼루션의 수학적 표현

- 이미지에서의 콘볼루션은 이미지의 경계를 찾거나(edge detection), 흐리게 또는 선명하게 하는데 사용되는 필터를 의미함

가능함

12) GPU의 개당 성능을 1 Tflop/s로 추정했을 때의 전체적인 성능치



자료 : iOS Developer Library - vImage Programming Guide

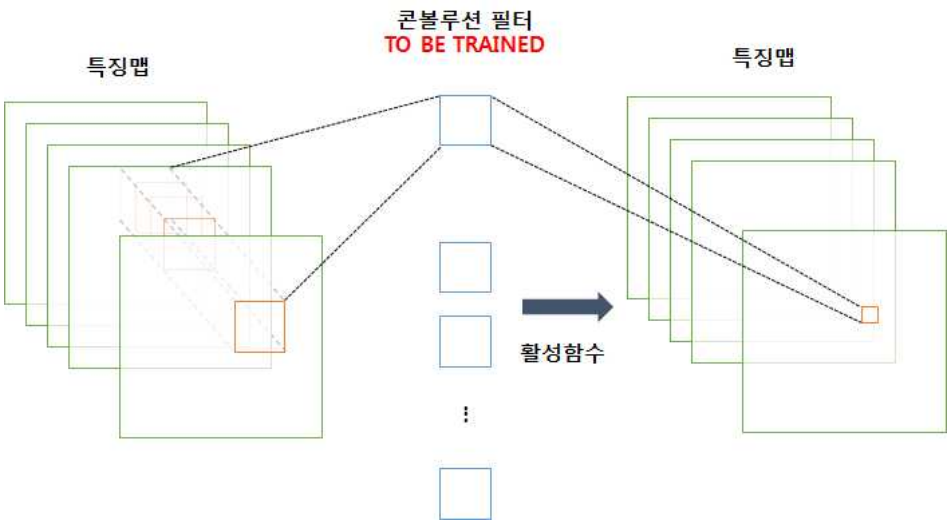
[그림 6] 콘볼루션의 과정

* [그림 6]은 이미지의 콘볼루션 과정을 표현하는데 3x3 행렬로 표현된 콘볼루션 필터가 이미지를 이동하며 각 픽셀 값을 곱해서 합하는 연산으로 계산됨

* 따라서 콘볼루션 필터의 역할은 이미지의 특징을 추출하는 것으로, 콘볼루션 필터 자체의 값에 따라 부각하고자 하는 관점이 달라진다고 볼 수 있음

- 콘볼루션 신경망에서의 콘볼루션 필터는 데이터를 통해 학습되고, 이것을 콘볼루션 층(Convolutional Layer)으로 지칭

* [그림 7]은 콘볼루션 층의 구조를 나타냄. 특징맵(feature map)은 이미지의 속성을 나타내는 것으로, 일반적인 이미지에서 RGB(Red, Green, Blue)의 세 가지 속성이 특징맵으로 사용됨



[그림 7] 콘볼루션 층의 개념도

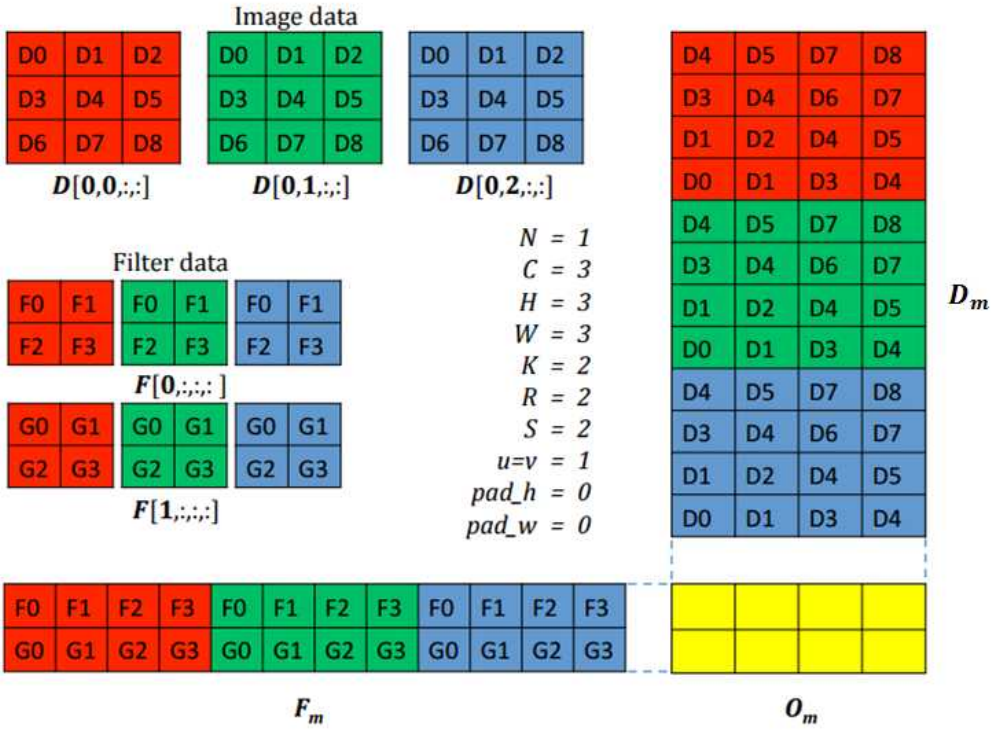
* 콘볼루션 필터로 계산된 값은 다음 층 특징맵의 원소 값으로 대응되고, 이 값을 보정하기 위해 활성함수를 적용

○ 콘볼루션 신경망에서 콘볼루션의 과정은 행렬곱 연산으로 구성

- 콘볼루션 필터와 계산을 위한 특징맵을 행렬에서 벡터 형태로 전개할 경우, 두 벡터의 내적[13]이 결과값으로 다음 층 특징맵의 원소로 대응

  * 콘볼루션 필터는 고정되어 있는 상태에서 해당하는 특징맵이 변화하므로, 다수의 독립적인 벡터 내적은 곧 행렬 곱으로 표현할 수 있음[14]

- [그림 8]는 콘볼루션 필터의 계산이 어떻게 행렬곱으로 대응되는지를 설명



자료 : cuDNN : Efficient Primitives for Deep Learning, https://arxiv.org/pdf/1410.0759.pdf

[그림 8] 콘볼루션의 계산 - 행렬곱

  * [그림 8]에서 이미지는 D, 콘볼루션 필터는 F라고 볼 때, D_m과 F_m은 행렬 데이터를 벡터화

  * 따라서 이미지 전체에 콘볼루션 필터를 적용한다는 것은 행렬 D_m과 F_m의 곱으로 표현됨

- 행렬곱 연산은 연산강도가 높은 대표적인 알고리즘으로 현대의 계산자원의 성능을 십분 활용할 수 있음 [부록 참조]

  * 행렬 곱은 [그림 11]의 Dense Linear Algebra(BLAS3)의 연산에 해당

13) dot product. 두 벡터의 원소의 곱을 모두 합한 값
14) 콘볼루션의 과정을 상세하게 소개한 강의자료 참고. http://cs231n.github.io/convolutional-networks/

AlphaGo의 계산량 분석

- AlphaGo의 계산의 대부분은 콘볼루션 신경망의 계산[15]
  - [그림 9]로 비유해서 표현하자면, 필터 행렬인 F_m의 크기는 192×9, 바둑판기보 행렬에 대응하는 D_m의 크기는 9×69312
  - AlphaGo의 첫 번째 콘볼루션 층은 19×19 바둑판을 48가지 관점에서 분류한 특징맵이 입력 값으로 사용[16]
    * 48가지 관점은 예를 들어 흑 돌의 위치, 백 돌의 위치, 빈 칸의 위치, 활로, 꼬부림 등 이진수로 표현가능
    * 첫 번째 콘볼루션 필터의 크기는 5×5이고 총 192개의 서로 다른 필터를 사용하여 다음 콘볼루션 층에 사용될 192개의 특징 맵을 생성
    * 따라서 48개의 19×19 입력 값에 192개의 5×5 필터와 콘볼루션 하는데 필요한 계산량은 약 **4.159 GigaFlop**[17]
  - 두 번째에서 열 세 번째 콘볼루션 층은 192개의 19×19의 정보가 입력 값이고 필터의 크기와 개수는 각 각 3×3과 192개
    * 첫 번째 콘볼루션 층과 동일하게 계산할 경우 총 **23.715 GigaFlop**[18]
  - 따라서 특정 바둑판 상태가 입력된 경우 착수 확률분포를 계산하거나 승산을 계산할 때 약 **30 GigaFlop의 계산량**이 필요[19]
    * NVIDIA의 계산전용 그래픽카드인 K40의 cuDNN 실측 성능은 약 1.2 TeraFlop/s[20]로 초당 약 40번의 콘볼루션 신경망 계산 가능
- AlphaGo의 콘볼루션 신경망은 정책 네트워크와 가치 네트워크 두 가지로 구성[21]
  - 정책 네트워크는 프로 바둑기사의 착수 선호도를 예측하는 것으로 특정 바둑판이 입력되면 빈 칸에 대한 착수 확률의 분포를 산출
    * 바둑 게임의 탐색에서 착수 가능한 모든 지점을 고려하지 않고 높은 착수 확률 분포를 갖는 지점을 기준으로 탐색 (게임 트리의 폭을 줄이는 과정)
  - 가치 네트워크는 현재 바둑판 상태의 승산을 근사
    * 예측한 승산이 정확할수록 게임 트리를 깊게 탐색해야하는 소요를 줄일 수 있음 (게임 트리의 깊이를 줄이는 과정)
  - 특정 바둑판 상태에서 성공적인 탐색을 위해서는 정책과 가치네트워크를 모두 계산해야 함
    * 앞서 소개한 콘볼루션 신경망 계산량에 따르면 176개의 GPU[22]를 활용하면 초당 7,040번의 콘볼루션 신경망의 계산이 가능하나 정책/가치 네트워크를 모두 계산해야 하므로 초당 3,520번의 착수를 계산할 수 있음
  - 따라서 **30초의 초읽기가 주어진다면, 176개의 GPU를 활용하여 약 100,000개의 바둑판 상태에 대하여 착수 확률분포와 승산을 계산**할 수 있음
    * **프로 바둑기사가 30초 초읽기 동안 약 100수 정도의 수읽기를 진행할 수 있다고 본다면 AlphaGo는 이에 대해 약 100배의 성능을 보유**
  - 정책과 가치 네트워크를 학습하는 과정은 위 계산량에 더하여 네트워크의 가중치를 조정하는 오류역전파법에 대한 계산을 고려해야 함

□ AlphaGo의 컴퓨팅 파워

○ AlphaGo의 컴퓨팅 환경은 다음 <표 2>와 같음

<표 2> AlphaGo의 컴퓨팅 환경

구 분	탐색 쓰레드	CPUs	GPUs
단일 (single)	40	48	8
분산 (distributed)	40	1202	176
인공신경망 학습	-	-	50

자료 : AlphaGo의 인공지능 알고리즘 분석, 소프트웨어정책연구소 (2016)

○ 인공신경망 학습에 소요된 전력량을 분석해보면 약 8,400 kWh가 필요

- 한 개의 GPU가 200Wh를 소모한다고 가정 했을 때, AlphaGo의 신경망 학습에 소요되는 시간은 총 5주[23]로 50개의 GPU가 총 8,400 kWh를 소모
- 이 계산을 가정집에서 계산한다면 누진세가 적용되어 약 360만원의 전기요금이 부과됨[24]

○ 인공신경망 학습은 hyper-parameter의 존재로 경험적으로 많이 계산해보고 결과를 유도하는 것이 일반적

- 5주에 해당하는 계산을 여러 번 수행하여 최상의 결과 활용

15) Mastering the game of Go with Deep neural networks and tree search, Nature (2016)
16) 자세한 신경망 구조는 다음 보고서 참고. AlphaGo의 인공지능 알고리즘 분석, 소프트웨어정책연구소 (2016)
17) 19×19 바둑판 * 48개 특징맵 * 5×5 필터 * 25 덧셈 (원소끼리 곱한 후 더하는 과정, 내적에서의 덧셈) * 192개 필터 * 2개 연산 (활성함수 계산) = 4.159 Gflop
18) 19×19 바둑판 * 192개 특징맵 * 3×3 필터 * 9 덧셈 (원소끼리 곱한 후 더하는 과정, 내적에서의 덧셈) * 192개 필터 * 2개 연산 (활성함수 계산) * 11층 = 23.715 Gflop
19) 콘볼루션 층만을 계산할 때는 이론적으로 27.874 Gflop이 필요하나, 활성 함수의 계산, fully connected layer등을 고려해 볼 때 도합 약 30 Gflop이 필요함
20) cuDNN : Efficient Primitives for Deep Learning, https://arxiv.org/pdf/1410.0759.pdf
21) 자세한 내용은 “AlphaGo의 인공지능 알고리즘 분석” 보고서 참고, https://spri.kr/post/14725
22) 판후이 2단과의 대국 당시 사용된 분산 AlphaGo의 GPU 개수
23) 정책네트워크 학습(3주), 정책네트워크 강화학습(1주), 가치네트워크 학습(1주)
24) 전기요금계산기, http://cyber.kepco.co.kr/ckepco/front/jsp/CY/J/A/CYJAPP000.jsp, 주택용(저압) 기준

3. 고성능컴퓨팅 환경의 국내·외 현황

(1) 해외 현황

□ HPC 분야는 세계 최대의 연산처리장치 생산국인 미국이 선도했으나, 최근 3년간 중국이 풍부한 자금력과 기술력을 앞세워 양강구도를 형성

○ HPC 기술력은 '슈퍼컴퓨터'로 대변되고, 매년 2회에 걸쳐 세계에서 가장 빠른 슈퍼컴퓨터를 1위에서 500위 까지 공개

- 매년 6월 유럽의 ISC(International Supercomputing Conference)와 11월 미국의 SC(Supercomputing Conference)에서 발표하고 결과는 홈페이지(top500.org)에 게시

- top500.org에서는 슈퍼컴퓨터 전용 수치해석 라이브러리인 HPL[25]을 활용하여 실측성능을 측정

* 이론 성능와 실측 성능을 모두 게시하고, 순위는 실측성능을 기준으로 함.

이론성능과 실측성능

- 이론성능은 하드웨어 스펙에 따라 달성 가능한 최대 성능을 지칭하는 반면, 실측성능은 LINPACK과 같은 수치라이브러리를 실제로 실행했을 때 얻을 수 있는 성능
- SW 아키텍처의 측면에서 이론성능과 실측성능의 차이를 좁히는 것이 기술력

○ 2016년 11월 순위를 기준으로, 세계 1위 슈퍼컴퓨터는 93.0 PetaFlop/s의 성능을 보인 중국의 "선웨이 타이후라이트"로 지난 6월에 이어 1위 자리를 지킴

- 선웨이 타이후라이트는 중국이 자체 기술력으로 개발한 SW20160 연산처리장치를 탑재

- 2위는 33.8 PetaFlop/s의 성능을 보이는 중국이 2013년 6월에 공개한 텐허2로, 이는 중국이 막대한 자금력을 바탕으로 기술까지 갖춘 HPC 강국으로 진입함을 보임

* 중국은 지난 3년간 top500 순위에서 1위를 고수

25) High Performance Linpack, 여기서 Linpack은 Linear Algebra Package의 약어로 과학기술 시뮬레이션의 가장 밑바탕이 되는 수치해석 알고리즘을 뜻함

중국의 자체개발 연산처리장치

- Sunway(약자로 SW)는 중국이 자체개발한 연산처리장치로 자세한 내용은 공개되지 않았으나 계산에 특화된 RISC[26]타입
  - 2006년부터 지속적으로 성능을 개선하여 듀얼코어(SW-2, 2008년), 16코어(SW1600, 2010년) 개발에 성공
- SW26010은 매니코어[27] 연산처리장치로 총 260개의 컴퓨팅 코어를 보유
  - SW26010의 이론성능은 3.06 TeraFlop/s로 선웨이 타이후라이트에 40,960개가 탑재

□ HPC의 글로벌 트랜드 - 엑사스케일(Exascale) 컴퓨팅

○ 슈퍼컴퓨터 성능은 [그림 9]과 같이 지수적으로(exponential) 발전

- 2016년 6월 기준 전 세계 고성능 컴퓨팅 성능의 합은 593.4 PetaFlop/s 수준으로 10년 전 보다 약 212배 빨라짐
- 세계 1위의 컴퓨팅 성능은 93.0 PetaFlop/s이며, 세계 최고수준의 컴퓨팅 성능과 500위에 해당하는 컴퓨팅 성능에는 약 8년 이상의 격차를 보임

자료 : Top 500. 재편집

[그림 9] 세계 HPC 성능 발전 추이

26) Reduced Instruction Set Computing, 범용 연산처리장치가 아닌 특수 목적에 의해서 고안된 연산처리장치로 명령어 집합이 기능에 최소화됨

27) 매니코어는 일반적으로 수백개 이상의 연산처리 코어를 탑재한 HW 아키텍처로 GPU에서 사용되는 방식. SIMD(Single Instruction Multiple Data)기반의 연산 수행

○ 그간의 발전 양상을 볼 때 향후 5년 내에 세계 최고 수준 컴퓨터는 엑사스케일(Exascale)급으로 현실화가 예상

- 미국을 비롯한 HPC 선진국은 <표 3>와 같이 엑사스케일 컴퓨팅 달성을 위해 체계적 R&D를 추진

- 특히 주목할 부분은 제4차산업혁명을 대비하기 위한 움직임으로 인공지능 분야에의 HPC 환경 지원이 반영

<표 3> 세계 HPC 관련 추진 현황

국 가	내 용
미 국	• 법·제도적인 체계 확보를 통한 기술개발 주도 - 테러 대응 다음으로 중요한 국가연구개발사업으로 지정 - 1991년 세계최초로 고성능 컴퓨팅법 제정 • 지속적이고 적극적인 투자로 HPC 시장 선도 - 2013년 NITRD[28] 프로그램은 예산의 1/3이상인 약 1.5조원을 슈퍼컴퓨팅 관련 분야에 투자 • Exascale Computing Project(ECP) 발족 - ECP에서도 **향후 응용분야로 데이터 분석과 기계 학습에 활용할 계획**
일 본	• 그간 세계 2위의 슈퍼컴퓨터 제조국으로 국가 경쟁력 강화를 위한 전략적 육성 추진 - 1980년대 후반부터 국가지원으로 슈퍼컴퓨터를 개발하고, 2011년 세계 1위 수준의 K컴퓨터를 개발 - 2014년 부터 약 1,300억 엔을 투자하여 엑사스케일을 목표로 한 Flagship2020 과제추진[29] • **2018년 1분기에 인공지능 전용 슈퍼컴퓨터 공개 예정** - ABCI(AI Bridging Cloud Infrastructure)는 일본의 인공지능 연구를 진흥하기 위해 개발 중인 33페타플롭스 급의 슈퍼컴퓨터 (195억 엔 투자)
중 국	• 국가 주도적인 집중투자로 인해 신흥강국으로 부상 - 2013년 약 4,300억 원을 투자하여 세계1위 슈퍼컴퓨터 Tianhe-2를 개발 - 현재는 **자체기술로 개발한 세계 1위 수준의 슈퍼컴퓨터**인 Sunway TaihuLight를 보유 - 연간 1조원 규모 이상을 슈퍼컴퓨터 관련 연구에 투자

28) Networking Information Technology R&D : 미국 연방정부가 연구자금을 지원하는 네트워킹 및 정보기술 연구개발

29) Flaghip2022로 변경, https://www.top500.org/news/japan-runs-into-detour-on-exascale-roadmap/

유 럽	• 유럽내 25개국이 참여하는 PRACE[30] 프로젝트로 컴퓨팅 환경에 대한 공동 활용체제 구축 • **인간의 뇌를 구현하는 프로젝트인 휴먼브레인 진행** - 2023년까지 약 11억 유로를 지원 • 영국은 슈퍼컴퓨팅 소프트웨어 개발 프로젝트에 약 3,500만 파운드(한화 약 497억 원)를 지원

출처 : 한국과학기술정보연구원, HPC기본 및 동향, 2016.

□ HPC의 수요는 과학기술 시뮬레이션에 특화된 컴퓨팅 기술에서 국가의 정책을 통해 공공 플랫폼으로 발전

○ 국가주도의 초고성능 컴퓨팅, 학술 및 연구목적의 공공 인프라, 기업과 산업체 연구소를 위한 분야, 개인 또는 엔터테인먼트 및 워크스테이션급 분야로 세분

Canada Compute

- 캐나다 정부에서 관리 및 지원하는 컴퓨팅 플랫폼으로 자국의 산·학·연의 R&D를 위한 계산자원을 지원
  - Canada Foundation for Innovation (CFI)로부터 재정적 지원을 받으며 현재 3500개가 넘는 리서치 그룹과 1만 명의 연구진을 보유
  - 성능은 20 PetaByte/s 이상의 처리 속도를 보이며, 현재까지 5518 저널, 58개 특허, 28개 투자, 23개 스핀오프(Spinoff)의 실적을 보임
- 고성능의 빅데이터, GPU 컴퓨팅과 스토리지, 테라바이트(Tera Bytes) 이상 데이터의 공유 및 전송을 지원
  - 특정 소프트웨어에 대한 확장성 제공
  - 글로버스 포탈, 유전자 데이터 베이스 제공
  - 각자의 클라우드 및 관리 서버 50GB 제공

  * Outwardfacing IP 주소 제공 및 사용자의 데스크탑을 기반으로 데이터 스토리지 및 백업 시스템 제공.

30) Partnership for Advanced computing in Europe : 유럽의 과학자와 기술자에게 최고급 슈퍼컴퓨팅 서비스를 제공

PRACE

- 유럽 내 여러 나라에 산재되어 있는 HPC 환경을 통합하여 범 유럽차원의 HPC 생태계 구성하고 고수준의 HPC 환경 및 서비스 제공을 목적하는 비영리 단체로 벨기에의 브뤼셀(Brussels)에서 설립
  - 시스템 및 운영을 위해 스페인, 이탈리아, 독일, 프랑스 등의 4개 주요국이 4억 유로를 부담하며, EU에서 6천 7백만 유로를 지원
- PetaFlop, ExaFlop 급의 슈퍼컴퓨팅을 지원하는 범유럽 규모의 Tier-0부터 국가내 규모의 Tier-1, 지역내 시스템인 Tier-2로 구성됨
  - 유럽 내 기업들의 연구를 지원하기 위해 연 2회 선정을 통해 Tier-0급의 리소스를 일정 기간 동안 무상으로 제공
- 오픈소스 프로젝트를 지원하고 참여국 내의 전문가를 활용
  - 프로젝트가 진행되는 동안 다수에 의한 검증 및 코드 재활용 측면의 오픈소스로 인한 장점을 가짐
- GPU를 활용한 최적화된 병렬처리를 제공하며 SW활용분야를 확장
- 중소기업의 경쟁력 향상을 위해 중소기업의 요구사항에 부합하는 HPC 시스템 서비스인 SHAPE[31]를 제공
  - 중소기업이 수행하려는 프로젝트에 적합한 HPC 환경 및 전문가 네트워킹 제공

31) SME(Small and medium enterprise) HPC Access Programme in Europe

(2) 국내 현황

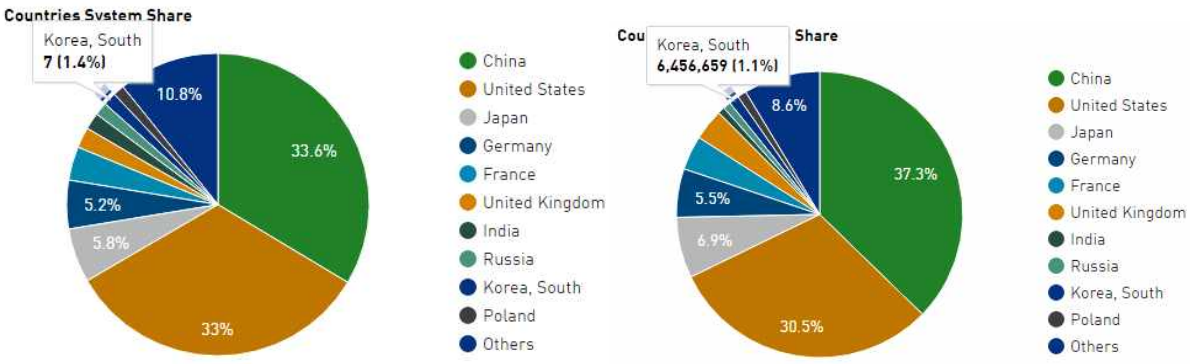
□ 국내 슈퍼컴퓨터 현황

○ 국내 슈퍼컴퓨터는 기상청이 보유한 미리와 누리 슈퍼컴퓨터가 각각 36위, 37위를 차지했고, 500위권에 7개의 슈퍼컴퓨터가 등재되었으며, 전 세계 컴퓨팅 파워 중 1.1% 차지 (2016년 6월 기준)

<표 4> 국내 HPC 장비 도입 현황

보유기관	명칭	제조사	도입년도	성능(TFlop/s)	순위
기상청	미리	Cray	2015	2,395	36
기상청	누리	Cray	2015	2,395	37
제조업체	-	HP	2015	824	296
기상청	우리	Cray	2014	345	395
대구경북과학기술원	아이렘	HP	2016	307	455
서비스제공업체	-	HP	2014	295	475
서비스제공업체	-	HP	2014	295	476
KISTI	타키온2	SUN	2009	274	500위 밖
삼성전자	SRD CAE	HP	2014	262	500위 밖

※자료 : Top500에서 추출 (2016년 6월 기준)



자료 : top500.org statistics https://www.top500.org/statistics/list/

[그림 10] 500위권에 등재된 슈퍼컴퓨터 대수(좌), 우리나라의 비중(우)

○ 국가 슈퍼컴퓨팅센터에서는 차세대 슈퍼컴퓨터 5호기를 도입할 예정이고, 그 성능은 약 30 PetaFlop/s으로 현재 500위권 슈퍼컴퓨터 중 2 ~ 3위에 해당하는 성능

○ 이와 동시에 세계 상위 10위권 수준의 슈퍼컴퓨터 기술 확보를 위해 '국가 초고성능컴퓨터 사업'을 추진하여 슈퍼컴퓨터 기술의 국산화를 위한 노력을 기울이고 있음

□ 국내에서는 지난 2011년 6월 세계 2번째로 슈퍼컴퓨팅 관련 육성법을 제정

○ '국가초고성능컴퓨터 활용 및 육성에 관한 법률'을 제정하고, 국내 HPC 산업을 육성하기 위한 계획 수립 및 전담기관(KISTI)을 지정하여 산업전반의 활용을 촉진

국가초고성능컴퓨터 활용 및 육성에 관한 법률

- IT 융합형 국가혁신체제 구축 및 국가초고성능 기본계획을 수립 (2011.6.)
  - (목적) 국가초고성능컴퓨터의 효율적인 구축과 체계적인 관리를 통하여 지속가능한 활용을 도모하고 과학기술의 발전 기반을 조성함으로써 국민의 삶의 질 향상과 국가경제 발전에 이바지함
- 전략과 목표
  - 신규 수요 창출을 통한 초고성능 컴퓨팅 활용 확대
  - 세계 Top10 수준의 초고성능컴퓨팅 서비스 기반 구축
  - 초고성능컴퓨팅 자체 개발 역량 확보 및 산업화 토대 마련
- 국가 초고성능 컴퓨팅 육성 기본계획으로 3대 전략-10대 전략과제를 수립하고 매년 시행계획을 수립

<표 5> 3대 전략 - 10대 전략 과제

3대 전략	10대 전략과제
초고성능컴퓨팅 활용 확대	과제 1. 국가초고성능컴퓨팅 활용 국가연구개발 활성화 과제 2. 초고성능컴퓨팅을 활용한 산업 혁신 강화 과제 3. 초고성능컴퓨팅 기반 공공·민간 응용 서비스 확대 과제 4. 초고성능컴퓨팅 이해 확산을 위한 국민 참여활동 확대
초고성능컴퓨팅 서비스 기반 강화	과제 5. 미래수요 대응 초고성능컴퓨팅 자원 확보 과제 6. 효율적인 국가 초고성능컴퓨팅 서비스 체계 구축 과제 7. 수요기반 초고성능컴퓨팅 전문인력 육성
초고성능컴퓨팅 기술개발·산업화 촉진	과제 8. 초고성능컴퓨팅 시스템 자체 개발 역량 확보 과제 9. 차세대 초고성능 컴퓨팅 원천 기술 R&D 확대 과제 10. 초고성능컴퓨팅 관련 산업 기반 육성

자료 : 제1차 국가초고성능컴퓨팅 육성 기본계획('13~'17)

4. 결 론

□ 학습기반 인공지능의 범용기술인 딥러닝은 HPC 환경의 성능을 십분 활용할 수 있는 핵심기술

○ 딥러닝은 연산강도가 높은 행렬곱 연산으로 세분화되어 HPC의 유연한 확장성을 보장

- 컴퓨팅 환경이 커질수록 계산의 효율이 증대하고 시간적 비용 절감이 가능
- 또한 행렬곱 연산의 병렬처리를 비롯한 딥러닝 알고리즘은 텐서플로우 등 인공지능 공개SW에 접목되어 HPC에 대한 진입장벽이 낮아짐

○ 제4차산업혁명의 핵심 동력인 인공지능 기술은 연구의 범위와 규모가 급속히 확장됨에 따라 이 수요를 충족할 수 있는 컴퓨팅 인프라 확보가 반드시 선결돼야 하는 과제

- 인공지능을 활용한 서비스나 제품은 빠른 출시와 사용자 경험이 중요하므로, 시의성 있는 대응을 위해 HPC 환경이 필수적

□ 현대 인공지능의 눈부신 성과와 글로벌 동향을 살펴보면 인공지능의 핵심인프라는 고성능컴퓨팅 환경

○ 그동안 인간과 대결하여 승리한 인공지능의 원천은 슈퍼컴퓨터급 계산 환경으로 인공지능 연구 분야의 혁신은 HPC 인프라 없이는 불가능

- 체스 그랜드 마스터를 꺾은 IBM의 Deep Blue, 퀴즈 우승자와 대결하여 승리한 IBM의 Watson, 세계 최정상 바둑기사를 꺾은 Deepmind의 AlphaGo는 당시 세계 2~300위 권의 슈퍼컴퓨터를 활용한 결과

○ 글로벌 HPC 동향을 살펴보면 제4차산업혁명의 기조에 따라 기계학습, 데이터 과학 분야의 환경 조성을 위한 노력이 반영

- 미국의 엑사스케일 프로젝트, 일본의 ABCI 슈퍼컴퓨터는 인공지능 기술을 선도하기위한 환경 조성
- 중국 역시 풍부한 자금력과 세계 1위의 슈퍼컴퓨팅 기술력을 통해 인공지능 연구를 주도하기 위한 공격적인 투자가 진행 중

[부 록]

(1) 인공지능과 GPU 컴퓨팅

□ 인공지능의 세 번째 황금기와 GPU

○ 1980년대 중반 소개된 인공신경망의 학습 방법인 오류역전파법(Error back-propagation method)은 신경망의 구조가 깊어질수록 계산량이 폭발적으로 증가

- 경험적으로 인공신경망의 구조를 최적화하므로 많은 시도를 통해 결과를 도출하는 것이 필수
- 따라서 인공신경망 자체를 학습하는 부분에도 시간이 많이 소요되지만, 다양한 모델을 시험해봐야 한다는 관점에서 계산비용이 기하급수적으로 증가

○ GPU는 이렇게 막대한 계산량을 극복할 수 있는 대안으로 부상

- GPU는 3D 게임이나 CAD와 같은 그래픽 작업에 최적화되어 발전해 왔는데 연산처리장치의 비약적인 발전으로 범용 목적(General Purpose)에 활용 가능해짐
- 최신 GPU의 연산성능은 동일 가격대비 CPU와 비교했을 때 약 10배의 성능 보유

□ GPU 컴퓨팅과 딥러닝 알고리즘

○ GPU 성능의 원천은 수천 개에 이르는 연산처리 코어

- 루프라인 모델[그림 3]은 알고리즘이 대량의 연산처리 코어를 동시에 활용할 수 있다는 것을 전제

  * 병렬처리가 불가능한 알고리즘의 경우 단일 코어에서 실행하는 환경과 동일하므로 활용도가 급격히 저하

- GPU의 성능을 십분 활용하기 위해서는 알고리즘의 병렬 처리가 선결과제

알고리즘의 병렬화

- 소스코드에서 반복문을 병렬화 : 반복문 내부에 데이터의 의존성(dependence)이 없는 경우 각각의 연산이 독립적이므로 병렬화 가능

```
// 벡터 덧셈 연산
for(j = 0 ; j < n ; j++)
    y[j] = alpha*x[j] + y[j];
```

→ 병렬화

```
// myid는 0부터 n까지 무작위로 정해짐
idx = myid;
y[idx] = alpha*x[idx] + y[idx];
```

만약 n개의 연산처리코어가 있다면, 각각의 코어에서 부여받은 myid번째 연산을 수행

- 병렬화가 불가능한 경우 : 반복문 내부에 데이터의 의존성 존재

```
// 피보나치 수열의 n번째 원소 계산
x[0] = 1; x[1] = 1;
for(j = 2 ; j < n ; j++)
    x[j] = x[j-1] + x[j-2];
```

○ GPU 컴퓨팅의 성공 분야와 한계

- GPU는 막대한 이론 성능을 보유하고 있으나 이것은 매우 제한적인 상황에서 달성 가능한 수치

  * 높은 연산강도, 고속 계산을 위한 정확도의 희생, GPU 아키텍처에 맞는 메모리 패턴, 병렬화 과정에 필요한 모수의 최적화 등 여러 가지 요인이 복합적으로 충족되어야 함

  * 따라서 GPU 컴퓨팅은 병렬화 과정뿐만 아니라 아키텍처에 기반한 최적화가 성능측면에서 매우 중요한 요소이기 때문에 진입장벽이 높음

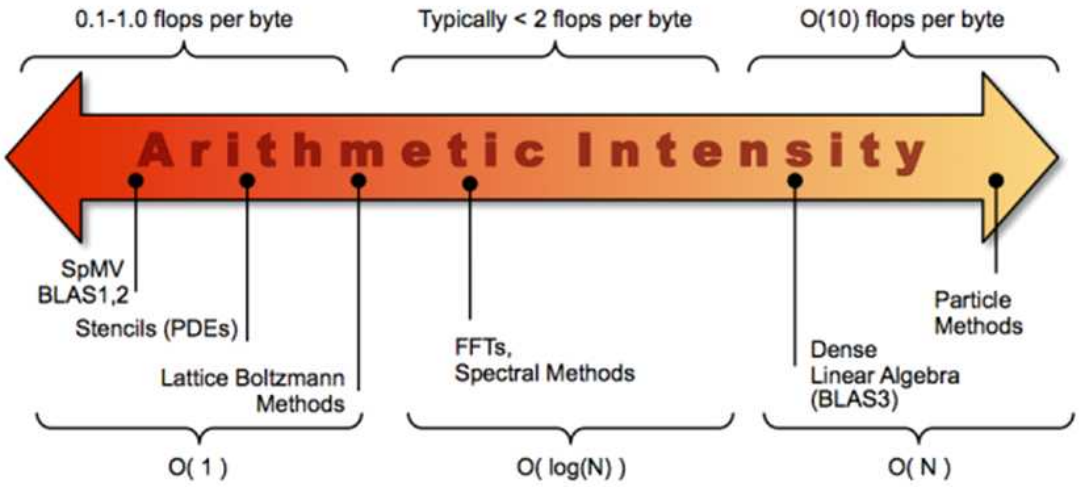
- GPU의 이론 성능을 달성할 수 있는 알고리즘은 벡터-행렬연산

  * 기본선형대수루틴(Basic Linear Algebra Subprogram, BLAS)은 벡터-행렬 연산을 계산복잡도에 따라 세 가지단계로 구분

  * BLAS 알고리즘은 대부분 반복적이고 독립적인 계산으로 이루어져 있기 때문에 병렬화 가능성이 높으므로 루프라인 모델에 의해 성능이 좌우됨

- [그림 11]는 다양한 알고리즘[32]에 대한 연산강도를 나타내며, 현대의 연산처리장치나 고성능 GPU에서는 연산강도가 높을수록 이론 성능에 가까워짐

32) SpMV - 희소행렬과 벡터곱 알고리즘, FFT - 고속 푸리에변환 등



자료 : Roofline Performance Model
https://crd.lbl.gov/departments/computer-science/PAR/research/roofline/

[그림 11] 다양한 알고리즘의 연산강도

행렬곱 연산의 병렬화

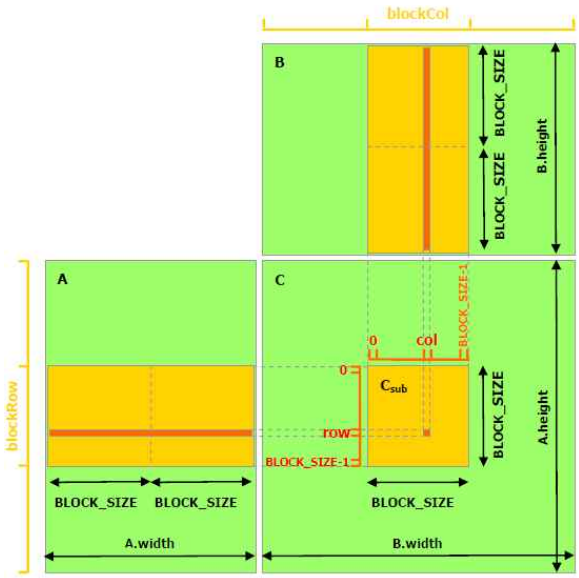
- 행렬곱 연산은 [그림 11]의 Dense Linear Algebra에 해당하는 것으로 연산강도가 행렬의 크기가 커질수록 높아짐
  - GPU를 비롯한 대부분의 계산자원에서 이론 성능을 십분 달성 가능
  - 특히 데이터의 재사용성이 높기 때문에 캐쉬 메모리의 활용 효율이 높음
  - 다양한 수치라이브러리의 가장 기반이 되는 알고리즘
    (이미지 처리, 행렬 분해, 딥러닝 등에 활용)
- 병렬화 과정은 다음 [그림 12]와 같은 개념으로 설명할 수 있음[33)]

```
// C = A*B
// 각 행렬은 n×n 차원 행렬
// 행렬 곱의 순차코드

for(j = 0 ; j < n ; j++)
    for(k = 0 ; k < n ; k++)
        for(l = 0 ; l < n ; l++)
            C[j][k] += A[j][l] * b[l][k]
```

↓ 병렬화

```
// 특정 block_size로 행렬을 나누어
// 한 쓰레드에서 해당하는 block의
// 한 원소를 계산
ridx = my_block_row_id;
cidx = my_block_col_id;
for(j = 0 ; j < n ; j++)
  Cvalue += A[ridx*n+j]*b[j*n+cidx];
C[ridx*n+cidx] = Cvalue;
```

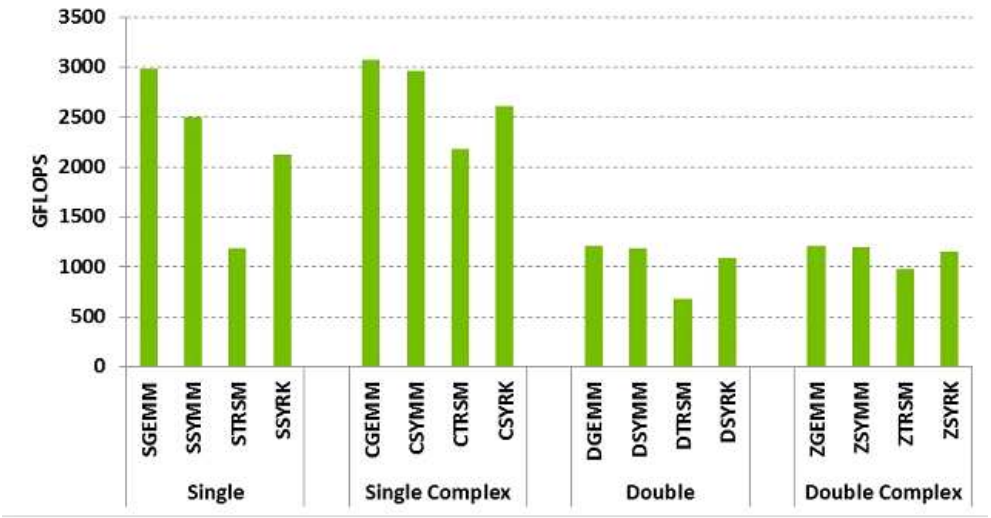


[그림 12] 행렬곱의 병렬화

○ 딥러닝의 학습과정은 병렬화 효율이 높은 BLAS 알고리즘으로 구성

- NVIDIA에서는 GPU 전용 BLAS인 cuBLAS[34]를 라이브러리 형태로 제공

  * 계산전용 그래픽카드인 NVIDIA K40m 모델에서 precision별 cuBLAS 연산의 성능 (행렬곱, 행렬해법 등) [그림 13]



자료 : cuBLAS Performance https://developer.nvidia.com/cublas

[그림 13] 유효숫자별 cuBLAS의 성능 실측치

- cuBLAS를 토대로 딥러닝 라이브러리 cuDNN(Cuda Deep Neural Network libraries)[35]는 기계학습 공개소프트웨어와 연동하여 사용 가능

  * TensorFlow, Caffe, Theano, Torch, CNTK 등 지원

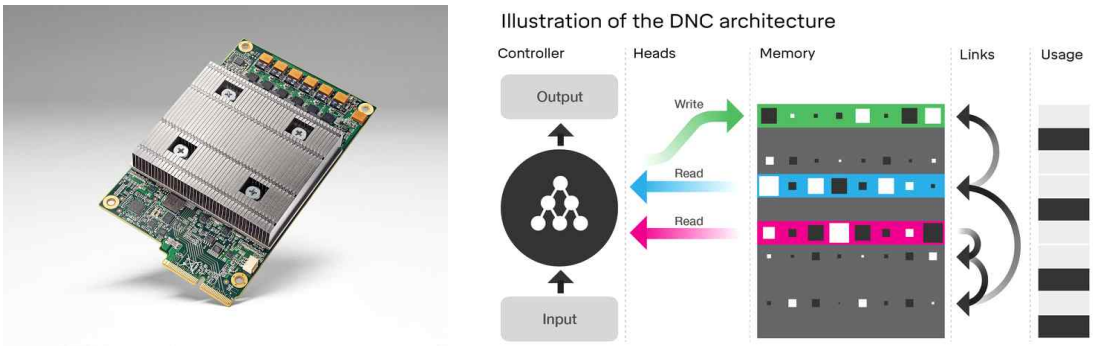
  * 다수의 GPU에서도 사용 가능한 환경 제공

33) 병렬화 과정에 대한 자세한 내용은 다음 문헌 참고
CUDA Toolkit Documentation, NVIDIA, https://docs.nvidia.com/cuda/cuda-c-programming-guide/
34) cuBLAS, NVIDIA, http://docs.nvidia.com/cuda/cublas/
35) cuDNN : Efficient Primitives for Deep Learning, https://arxiv.org/pdf/1410.0759.pdf

인공지능 전용 컴퓨팅 환경

- 지난 5월 구글은 기계학습에 특화된 연산처리장치 TPU(Tensorflow Processing Unit)을 공개[36)]
  - 계산에 필요한 유효숫자(precision)를 최소화하여 트랜지스터 당 계산의 효율을 극대화
  - 구글의 검색알고리즘인 RankBrain과 Street View에 활용
  - 구글의 기계학습 공개소프트웨어인 TensorFlow와 클라우드 서비스인 Cloud Machine Learning과 연계하여 서비스 예정

- AlphaGo 개발진 Deepmind는 미분가능한 신경 컴퓨터(Differentiable Neural Computer, DNC) 기술을 소개하여 기존 인공신경망의 새로운 학습 방법을 제안
  - DNC의 특징적인 개념은 기존 인공신경망의 기능에 정보 저장의 기능을 추가한 것으로, 정보의 조각들로부터 사실을 추론하는 사람의 뇌의 기능과 더욱 유사한 방향을 제시
  - 특히 메모리에 읽고 쓰는 작업만으로 신경망을 학습할 수 있다는 기술을 소개하여 대용량 데이터에 대한 효율적인 학습 체계를 제시



자료 : Differentiable neural computers, Deepmind (2016)

[그림 14] 구글 TPU(좌), DNC의 개념도(우)

36) Google supercharges machine learning tasks with TPU custom chip (2016. 05) https://cloudplatform.googleblog.com/2016/05/Google-supercharges-machine-learning-tasks-with-custom-chip.html

(2) 게임 인공지능 성공사례

□ (1997, IBM Deep Blue) 세계 체스 챔피언 개리 카스파로프와 대결하여 승리

○ Deep Blue는 1985년 카네기 멜론 대학의 ChipTest[37] 프로젝트로부터 시작했으며, 체스 게임을 수행하기 위한 컴퓨터를 디자인하는 것이 목표

○ 1996년 2월 챔피언과의 대결에서 4-2로 패배하였으나, 1997년 재대결에서 3½-2½로 승리 (비긴 경기는 ½로 환산)

○ Deep Blue의 체스 경기 전략은 brute force로 제한시간 안에 최대한 많은 수를 고려하여 최적의 착수를 결정

- 체스의 경우의 수는 약 35^{80} (트리로 생각한다면 너비가 35, 깊이가 80)
- 알고리즘 적인 측면보다 컴퓨팅 파워에 의존성이 큼. Deep Blue의 계산 성능은 1초에 2억 번의 수를 탐색할 수 있음

자료 : Deep Blue https://www-03.ibm.com/ibm/history/exhibits/vintage/vintage_4506VV1001.html
http://stanford.edu/~cpiech/cs221/apps/deepBlue.html

[그림 15] Deep Blue 컴퓨터 (좌) Deep Blue와 카스파로프의 대결 (우)

37) 1985년 VLSI(Very-Large-Scale Integration) 기술을 활용하여 체스 프로그램을 개발. 초당 5만 번의 수를 연산할 수 있는 능력 보유

□ (2011, IBM Watson) ‘퀴즈쇼 제퍼디!’에서 과거 우승자에게 승리

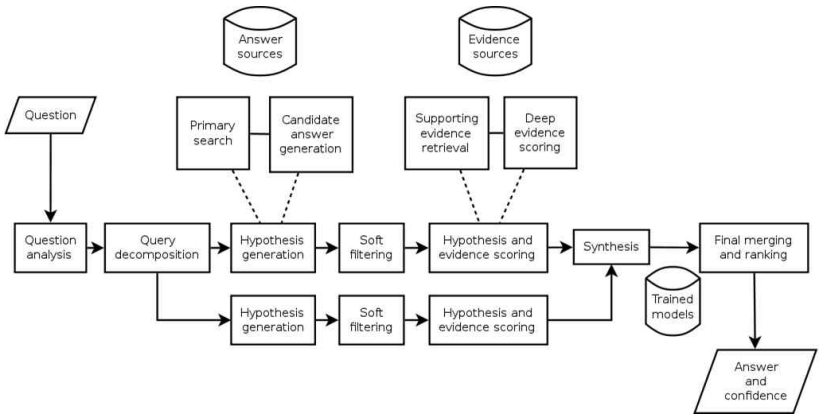
○ IBM Watson은 자연언어 처리를 통해 질문에 답하는 컴퓨터 시스템으로 ‘퀴즈쇼 제퍼디!’에서 우승하기 위한 인공지능을 만드는 것이 목표

○ 2011년 IBM Watson은 과거 우승자인 Brad Rutter, Ken Jennings와 대결하여 백만 달러를 가장 먼저 달성함으로써 우승을 차지함

- IBM Watson은 약 4 테라바이트에 해당하는 2억 건의 정보(위키피디아 포함)를 학습했고, 대결 도중에는 인터넷을 사용 안함

- Ken Jennings 30만 달러 (2위), Ban Rutter 20만 달러 (3위)

○ Watson은 자연어 처리(natural language processing), 정보 검색(information retrieval), 지식 표현(knowledge representation), 기계학습(machine learning) 기법을 활용하여 문제의 답을 유도 (약 100가지 이상의 기법을 사용함)



자료 : Watson (computer) https://en.wikipedia.org/wiki/Watson_(computer)

[그림 16] Watson의 SW 알고리즘 개요

자료 : IBM Watson, https://www-03.ibm.com/press/uk/en/presskit/36929.wss

[그림 17] IBM Watson 컴퓨터 (좌) 퀴즈쇼 제퍼디! 대결 (우)

□ (2016, Google AlphaGo) 세계 최정상의 바둑기사에게 4:1로 승리

○ AlphaGo는 바둑 인공지능 프로그램으로 공개 당시 프로수준의 바둑실력을 보유하고 있었으며, 지난 3월 세계 정상급 기사인 이세돌 9단에게 도전함

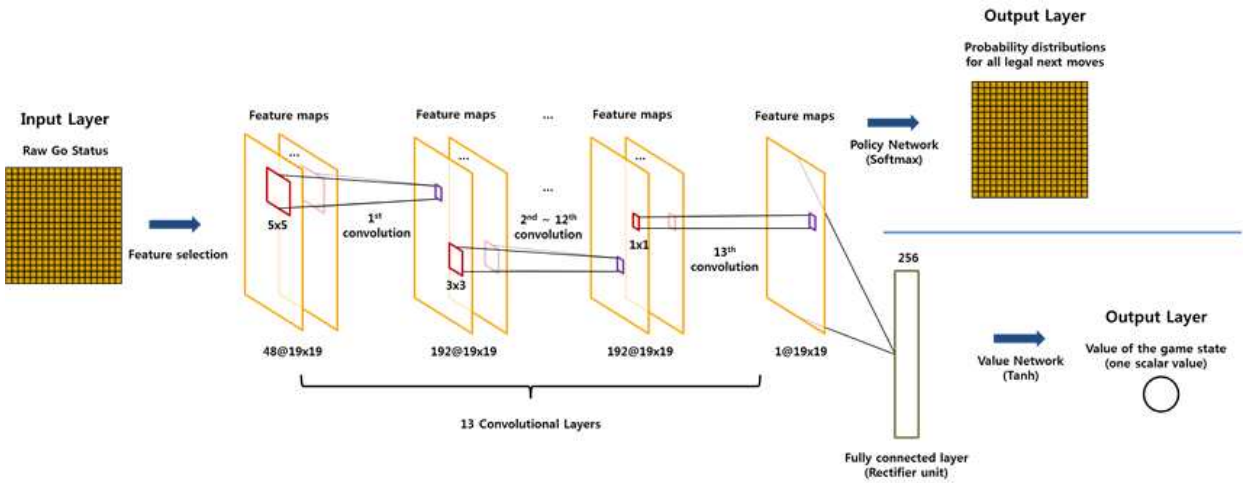
○ 그 결과 AlphaGo는 4:1이라는 성적으로 이세돌 9단에게 승리를 거둠으로써 인공지능의 대중적 관심을 일으킴

○ AlphaGo의 승리전략은 프로 바둑기사의 기보를 학습하여 무한대에 가까운 바둑의 경우의 수를 선택적으로 좁힘

- 바둑의 경우의 수는 약 150^{250}
- AlphaGo는 콘볼루션 신경망을 활용하여 프로 바둑기사의 기보 16만 개를 학습
- 또한 강화 학습 기법을 활용하여 자체 대국을 통해 정확도를 향상시킴
- 착수 전략은 몬테카를로 트리 탐색(Monte Calro Tree Search, MCTS)기법을 활용하여 제한시간 안에 최적의 수를 결정함

○ AlphaGo는 구글 클라우드를 활용하여 기보를 학습했고, 실제 대국에서도 미국 소재의 클라우드 컴퓨터에 접속하여 경기를 수행

- 매니코어 시스템인 GPU를 다수 활용
- 구글 클라우드에서 사용된 구체적인 계산자원 모델은 공개하지 않음



자료 : AlphaGo의 인공지능 알고리즘 분석, 소프트웨어정책연구소 (2016)

[그림 18] AlphaGo의 인공신경망 구조

[참고문헌]

1. 국외문헌

Silver, D. et al., “Mastering the game of Go with Deep neural networks and tree search,” Nature vol 529, pp. 484-489, 28 Jan 2016.

cuDNN : Efficient Primitives for Deep Learning, https://arxiv.org/pdf/1410.0759.pdf

Barba, Lorena A., and Rio Yokota. “How will the fast multipole method fare in the exascale era.“ SIAM News 46.6 (2013): 1-3.

2. 국내문헌

AlphaGo의 인공지능 알고리즘 분석, 소프트웨어정책연구소 (2016)

3. 기타(신문기사 등)

Convolutional Neural Network, http://cs231n.github.io/convolutional-networks/

cuBLAS, NVIDIA, http://docs.nvidia.com/cuda/cublas/

CUDA Toolkit Documentation, NVIDIA, https://docs.nvidia.com/cuda/cuda-c-programming-guide/

Deep Blue https://www-03.ibm.com/ibm/history/exhibits/vintage/vintage_4506VV1001.html

Exascale Computing Project, https://exascaleproject.org/exascale-computing-project/

Fused Multiply-Add, https://en.wikipedia.org/wiki/Multiply%E2%80%93accumulate_operation

Top500, http://www.top500.org

Intel Xeon Processor E5-2699 v4, http://ark.intel.com/products/91317

Intel Xeon Phi 7120P, http://ark.intel.com/products/75799

Japan Runs into Detour on Exascale Roadmap https://www.top500.org/news/japan-runs-into-detour-on-exascale-roadmap/

Nvidia Tesla P100, http://www.nvidia.com/object/tesla-p100.html

Roofline Performance Model https://crd.lbl.gov/departments/computer-science/PAR/research/roofline/

Samsung Exynos, Wikipedia https://en.wikipedia.org/wiki/Exynos

Watson (computer), Wikipedia, https://en.wikipedia.org/wiki/Watson_(computer)

주 의

1. 이 보고서는 소프트웨어정책연구소에서 수행한 연구보고서입니다.
2. 이 보고서의 내용을 발표할 때에는 반드시 소프트웨어정책연구소에서 수행한 연구결과임을 밝혀야 합니다.