

양자 인공지능 과제 관련 내용 정리 (일부)

https://youtu.be/W4M_U1Kj8Po

양자 컴퓨터 상에서 구현된 인공신경망 기술

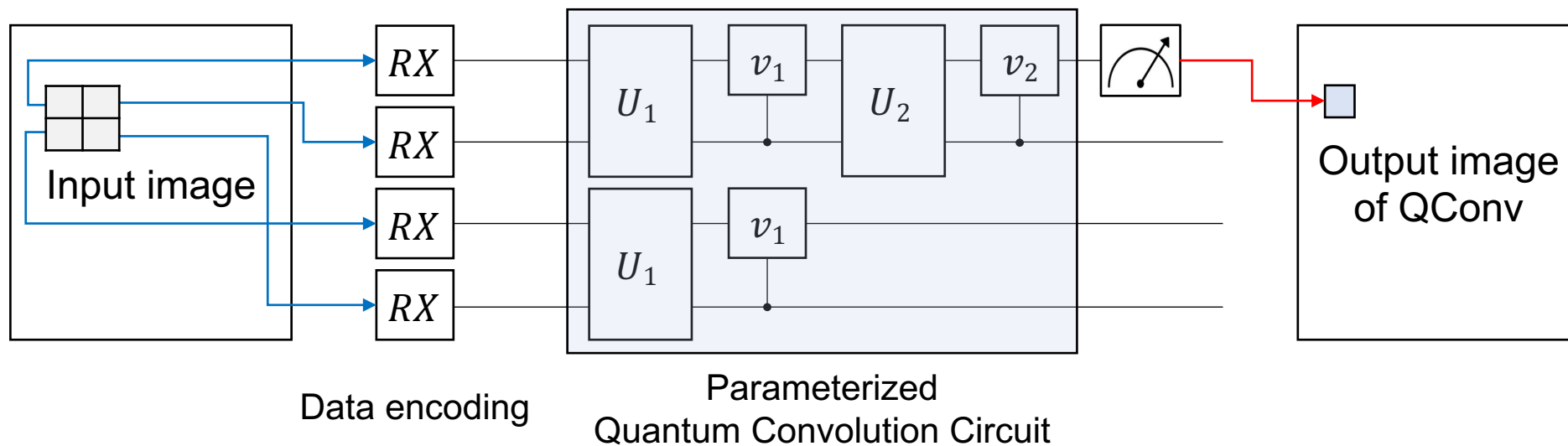
무료 양자 자원으로 수행 가능한 양자 신경망 수준

양자 컴퓨터 상에서 구현된 인공신경망 기술

양자 컴퓨터 상에서 구현된 인공지능망 기술

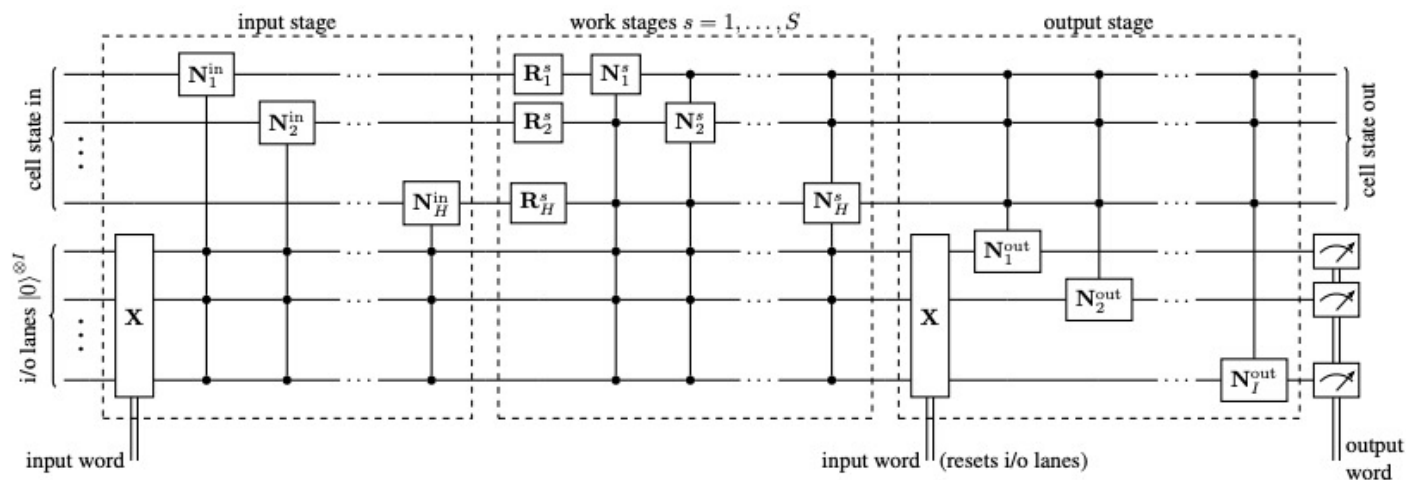
- 고전 신경망 중 일부는 양자 컴퓨터 상에서 동작 구현된 상태
 1. Quantum Convolutional neural networks (QCNN)
 2. Quantum Recurrent Neural Networks (QRNN)
 3. Quantum Generative Adversarial Networks (QGAN)
- 이외에도 Quantum Graph Neural Networks (QGNN), Quantum Graph Recurrent Neural Networks(QGRNN) 등이 연구되고 있음
- 또한, 고전 신경망에서 앙상블 등을 통해 네트워크를 융합해서 사용하는 것처럼 개별적인 양자 회로를 병렬로 동작시키는 방법들도 사용

- 고전 신경망에서 수행하는 **Convolution 및 Pooling 레이어를 양자회로로 구현**
→ 입력데이터와 필터(가중치)를 행렬곱 연산하므로 지역적, 공간적 특징 학습에 효과적
- Classical CNN과 비교하여 **더 적은 파라미터**로 비슷한 수준의 성능 달성 가능
- 보통 필터에 **qubit를 많이 할당하지 않고 2x2, 3x3 커널(필터) 정도 사용 (4-qubit, 9-qubit)**
→ convolution은 입력 데이터 전체에 수행하는 것이 아니라 필터 단위 → **적은 큐비트로 수행하기에 적합**
- 정해진 게이트 및 회로가 있는 것이 아니고 랜덤으로도 설정 가능



QRNN

- 고전 신경망의 한 종류인 RNN은 Sequence to sequence 모델의 기초
- 기존의 RNN 및 LSTM과 마찬가지로 네트워크에 제공된 입력에 대해 연속적으로 cell을 적용
→ cell : 이전 hidden layer의 출력을 입력으로 하며, sequence 학습을 위해 이를 기억하는 메모리 역할 (NN에서의 node)
- 입력 층 및 출력 층과 각 은닉 셀의 비선형 활성화 함수를 양자 회로로 구성**
- [QRNN]전까지는 실제로 실행 가능한 양자 순환 신경망(QRNN)은 제안되지 않음
→ 해당 논문에서는 **12-qubit**를 사용하여 MNIST 데이터셋에서 **0과 1만 분류하는 이진 분류기를 제안**
고전 RNN(20808개) 및 LSTM (21448개)에 비해 더 적은 1956개의 파라미터로 학습 가능



QGAN

- 고전 신경망의 **GAN**은 랜덤 분포로부터 실제와 유사한 데이터를 생성
 - Generator는 데이터의 확률 분포를 학습
 - 이를 위해 discriminator는 실제 데이터와 Generator가 생성한 데이터를 둘 다 입력 받은 후 real과 fake 데이터를 구분하도록 학습
 - Generator는 이러한 정보를 반영하여 구분할 수 없는 데이터를 생성
- QGAN는 이와 동일한 구조를 가지지만, **Generator(양자 신경망)** 및 **Discriminator(고전 또는 양자)**로 구성
→ 현재 Qiskit에서 제공 중인 QGAN*은 **discriminator가 고전 신경망**으로 구성



Quantum Neural Network

Classical Neural Network or Quantum Neural Network

*https://qiskit.org/documentation/machine-learning/locale/ko_KR/tutorials/04_qgans_for_loading_random_distributions.html

QGAN

- QGAN에서는 **현재의 양자 하드웨어를 통해 실제로 생성(학습) 능력을 가질 수 있는가**에 대한 문제가 발생
Discriminator는 양자 이진 분류기이므로 가능
→ Generator에 대해 제한된 양자 자원을 사용하여 학습하기 위한 연구 진행
- 현재, **간단한 MNIST 데이터 셋(0 또는 1 분류, 2x2 이미지)**에 대해서는
훈련에 필요한 **파라미터의 수를 줄일 수 있으며, 고전 GAN과 유사한 수준의 성능 달성 가능**

Huang, He-Liang, et al. "Experimental quantum generative adversarial networks for image generation." Physical Review Applied 16.2 (2021): 024051.

QGAN

- 크기가 큰 QGAN은 RSA등으로 암호화 된 데이터를 생성해낼 수 있을 것이라고 추측함
→ 그냥 결론에 한 줄 나온 이야기이고 이에 대한 설명은 딱히 없습니다..**
(근거는 쇼어 알고리즘으로 암호 해독을 할 수 있어서라고 적혀 있음)
- 제가 생각하기에
랜덤 분포 → G → 키에 대응되는 암호화 된 데이터를 생성 → D → 해당 키에 대한 암호문이 맞는지 판단
이를 통해 키에 맞는 암호문을 생성하도록 학습인 것 같습니다.

** Dallaire-Demers, Pierre-Luc, and Nathan Killoran. "Quantum generative adversarial networks." *Physical Review A* 98.1 (2018): 012324.

무료 양자 자원으로 수행 가능한
양자 신경망 수준

무료 양자 자원으로 수행 가능한 양자 신경망

- 각 플랫폼의 무료 Simulator 사용, 양자 인공 신경망 기반의 간단한 이진 분류 문제 기준

Platform	Quantum Circuit	Parameterized Quantum Circuit	
Qiskit	24-qubit	14-qubit (QSVM Quantum-only)	Entangled, shots = 1, repetition = 2
		27-qubit (Hybrid with Tensorflow)	Not entangled, shots = 100
		17-qubit (Hybrid with PennyLane)	Entangled, shots = 0
Amazon Braket	25-qubit	20-qubit (Hybrid with Scipy optimizer (Classical optimizer))	Entangled, shots = 0
Q# (Azure Quantum)	25-qubit (해당 큐비트 이상은 너무 많은 시간이 소요되어서 확인 불가)	Q#에서 제공하는 Microsoft.Quantum.Kata.QuantumClassification 라이브러리 필요. Microsoft.Quantum.Measurement와 같은 일반 회로 모듈은 문제 없으나, 해당 분류기는 문제 발생..	
		1-qubit (Qiskit circuit, Hybrid with Tensorflow)	Not entangled, shots = 100 (에러 없지만 2-qubit 부터 제대로 실행되는지 확인이 불가할 정도로 느림)

*Q#은 Azure Quantum notebook (Cloud) 사용하였고, 뭔가 비정상적으로 느린 것 같은데 왜 느린 지는 아직 확인하지 못했습니다.

무료 양자 자원으로 수행 가능한 양자 신경망

- Entangled circuit(큐비트 얽힘 존재하는 회로)일 경우

- 1. 얽힘 없는 회로보다 더 적은 큐비트 사용 가능

- 2. Shot을 적게 설정해도 되는 듯 → Shot이란 큐비트의 상태를 몇 번 측정할지를 말함

- (즉, shots= 100이면 100번 측정하여 00인 경우 5번, 01인 경우 40번, 10인 경우 30번, 11인 경우 25번이 됨)

- 3. 더 오랜 시간 소요

- 양자 신경망의 경우, 일반적인 양자회로 구현 시보다

- 1. 더 적은 큐비트 사용 가능

- 2. 회로 실행에 더 오랜 시간 소요

- 기댓값을 계산하고, 해당 값을 기반으로 내부 파라미터를 조정하는 과정을 반복적으로 수행하여야 하므로

- 동일한 작업을 하더라도 고전 신경망에 비해 더 적은 epoch로 학습 가능

- 블로흐 구면 위의 모든 값을 나타낼 수 있는 큐비트를 사용하므로 데이터의 표현 범위가 풍부하므로

- 간단한 이진 분류 문제에 대해 약 절반 정도의 epoch로 비슷한 수준의 성능 보임 (해커톤 때 확인했습니다)

- 더 적은 파라미터 수

무료 양자 자원으로 수행 가능한 양자 신경망

- 구현 시 유의 사항

1. 플랫폼마다 사용 불가능한 라이브러리 존재

2. 적절한 큐비트 사용 필요

- 1) 간단한 데이터에 의도적으로 많은 큐비트를 적용해보았을 때, loss 값이 매우 커짐 → 수 만배 정도

- 2) 큐비트 증가 → 수행 시간 증가 → **회로 반복 수행이 필요한 양자 신경망의 특성 상 최소한의 큐비트 사용 필요**

→ 동일 큐비트에 데이터 재업로딩 등의 큐비트 재사용에 대한 연구도 진행 중

3. 암호 분석의 경우

- 1) 키 비트를 예측한다면, 각 비트마다의 이진 분류가 되므로 많은 큐비트 필요, 소요 시간 증가

:10비트 데이터셋에 대한 패리티비트 예측에서 0,1분류가 아닌 [0,0] [0,1] 예측으로 바꾸었을 때 20시간이 걸려도 안 끝남

- 2) 그러나 0,1,...,63과 같이 십진수로 키를 찾을 경우, label의 수가 증가하여 키 복구 확률이 떨어질 것으로 생각....

감사합니다.