양자 인공지능을 이용한 암호분석

- 3차 발표 -

김현지, 임세진, 서화정

https://youtu.be/f9OrryIIcEE

양자컴퓨터와 양자인공지능

양자인공지능 관련 최신 개발 환경

양자인공지능을 통한 암호분석

Development Roadmap | Executed by IBM ✓ On target ↻

IBM **Quantum**

	2019 ✓	2020 ✓	2021 ✓	2022	2023	2024	2025	Beyond 2026
	Run quantum circuits on the IBM cloud	Demonstrate and prototype quantum algorithms and applications	Run quantum programs 100x faster with Qiskit Runtime	Bring dynamic circuits to Qiskit Runtime to unlock more computations	Enhancing applications with elastic computing and parallelization of Qiskit Runtime	Improve accuracy of Qiskit Runtime with scalable error mitigation	Scale quantum applications with circuit knitting toolbox controlling Qiskit Runtime	Increase accuracy and speed of quantum workflows with integration of error correction into Qiskit Runtime
Model Developers					Prototype quantum software applications →		Quantum software applications	
							Machine learning | Natural science | Optimization	
Algorithm Developers		Quantum algorithm and application modules		✓	Quantum Serverless			
		hine learning | Natural science | Optimization				Intelligent orchestration	Circuit Knitting Toolbox	Circuit libraries
Kernel Developers	Circuits ✓		Qiskit Runtime ✓					
				Dynamic circuits ↻	Threaded primitives	Error suppression and mitigation		Error correction
System Modularity	Falcon ✓ 27 qubits	Hummingbird ✓ 65 qubits	Eagle ✓ 127 qubits	Osprey ↻ 433 qubits	Condor 1,121 qubits	Flamingo 1,386+ qubits	Kookaburra 4,158+ qubits	Scaling to 10K-100K qubits with classical and quantum communication
					Heron 133 qubits x p	Crossbill 408 qubits		

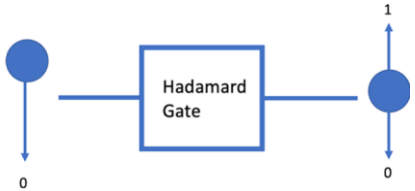
2021년 실제로 달성

IBM의 올해 양자 프로세서 개발 목표

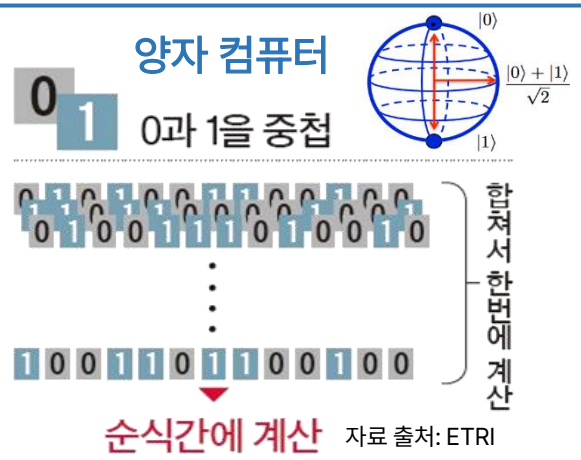
2025년부터는 실용적인 모델들이 개발되기 시작

공개키 해킹 가능한 레벨로 예상됨

양자컴퓨터의 특징 및 AI 분야에서의 활용

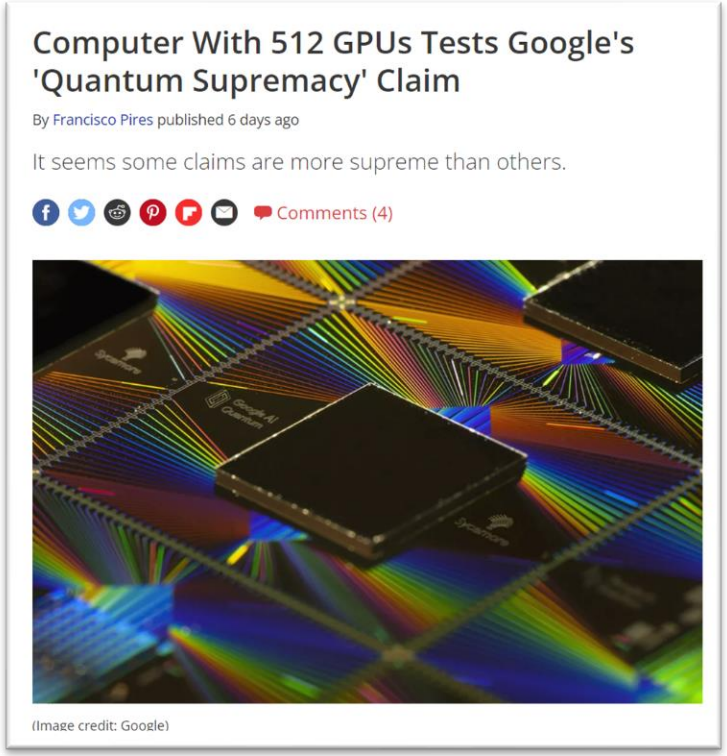


- **양자 중첩**
  - 인자는 **여러 상태를 확률적**으로 가지고 있음
    → 측정 시 단 **하나의 상태**로 결정

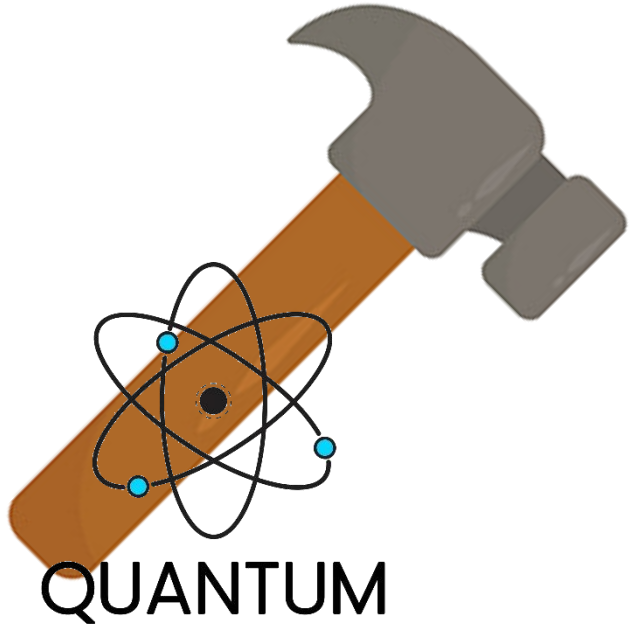


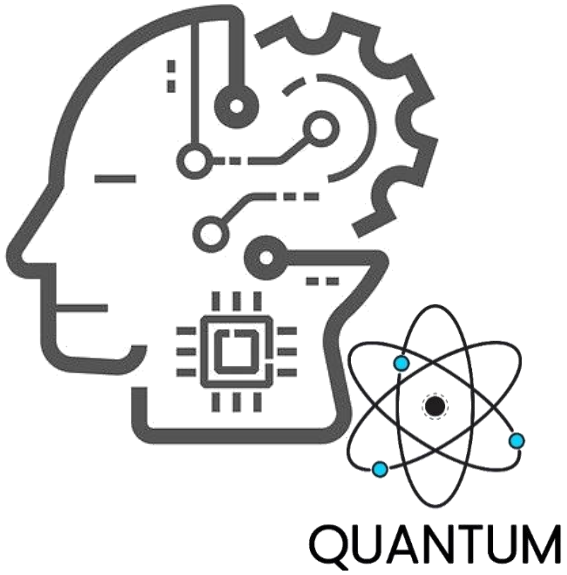
- **양자컴퓨터 활용은 AI 기술에서 IT 기업들에 의해 활발히 진행 중**
- **Quantum AI Lab**
  - NASA, Universities Space Research Association, Google에서 운영 중
  - **양자컴퓨터 AI**를 활용하여 **컴퓨터 공학적 문제점**을 해결하는 것을 목표로 함

년도	활동 내역
2013	Google에서 Quantum AI Lab 발표. 연구실 초기에는 D-Wave Two를 사용
2014	D-Wave Two와 전통 컴퓨터 간의 연산 성능 비교 분석
2019	**양자 우월성 달성 논문 발표.** 자체 개발 양자 프로세서인 **시커모어 사용**



양자컴퓨터를 통한 암호 해킹





양자 컴퓨터 상에서 주로 연구된 공격:
Grover를 통한 암호 해킹 – 전수조사

2021년도 한성대에서 세계 최초로 시도된 공격:
Quantum AI를 통한 암호 해킹 – 취약한 패턴 분석

과제의 목표 및 내용

최종 목표 : 양자 컴퓨터 상에서 양자 인공지능을 통해 암호 분석

전통적인 인공지능을 이용한 **암호 분석의 한계점 확인** › **양자 컴퓨터**와 **양자 인공지능의 특징 분석** **최신 연구 결과 파악** › 양자 컴퓨터 개발 **플랫폼 성능 비교 분석** **양자 인공지능을 통한 암호 분석**

양자 인공지능 알고리즘	Quantum Support Vector Machine (QSVM) Quantum Neural Network
개발 플랫폼	IBM ProjectQ / IBM Qiskit / Microsoft Q#
암호 알고리즘	고전암호 (Caesar, Vigenère) S-DES, S-AES S-SIMON, S-SPECK

»»

Quantum Neural Network (Hybrid)
Pennylane
S-DES **(분석 성공)** S-AES **(한계점 확인)**

현재 양자 인공지능의 시간 및 자원의 한계로 인해 대상 암호 알고리즘 및 방법론 변경

QSVM* → Hybrid

안정적이고 실현 가능한 암호 분석을 위해 hybrid 방식으로 변경

분석 가능한 S-DES에 집중

더 많은 실험 진행

한계점 및 방향성 파악

해당 결과를 기반으로 **다른 암호로 확장**할 경우의 한계점 및 방향성 파악

*QSVM의 한계점 : 양자 자원 부족, 많은 데이터 사용 어려움, 매우 긴 학습 시간

고전 신경망 vs 양자 신경망

- **고전 신경망과 달리 양자 신경망은 큐비트 회전각을 매개변수로 사용한다는 차이점을 가짐**
- 양자 신경망에서는 **행렬 곱 (양자 게이트)을 통해 큐비트의 상태를 변경**
  - 주로 사용되는 양자 게이트는 **회전 게이트 (Rx, Ry, Rz)** → 회전 각이 필요하며, 이를 매개변수라고 함
- **하이브리드 방식**
  - 고전 신경망 및 고전 최적화 알고리즘과 **특정 복잡한 작업을 위해 양자 컴퓨터를 사용**
  - 고전 신경망의 최적화 함수 및 손실 함수 사용 가능
  - **인공지능 프레임워크 (Tensorflow, Pytorch)**와 **양자 프레임워크 (Qiskit, Cirq)를 결합**하여 사용
- **NISQ (Noisy Intermediate-Scale Quantum era; 중간 규모의 양자 컴퓨터)**
  - **오류 정정이 어려워서 계산 오류가 많이 발생 →** Quantum만 사용하는 양자 신경망은 현실적으로 어려움
  - NISQ 시대의 프로세서는 기존의 보조 프로세서와 함께 작동해야 효과적
    → 현재는 고전 신경망과 양자 신경망의 **하이브리드 방식이 성능 면에서 안정적**

회전 게이트

$$\hat{R}_x(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}$$

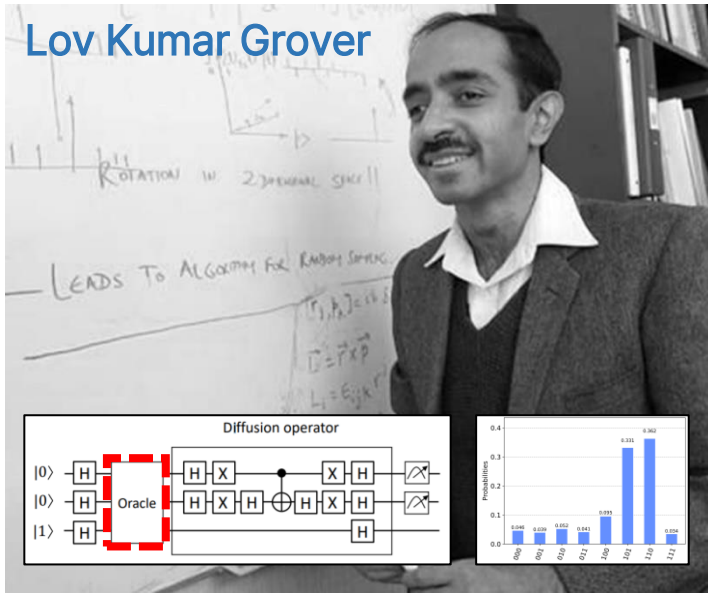
$$\hat{R}_y(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}$$

$$\hat{R}_z(\theta) = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix}$$

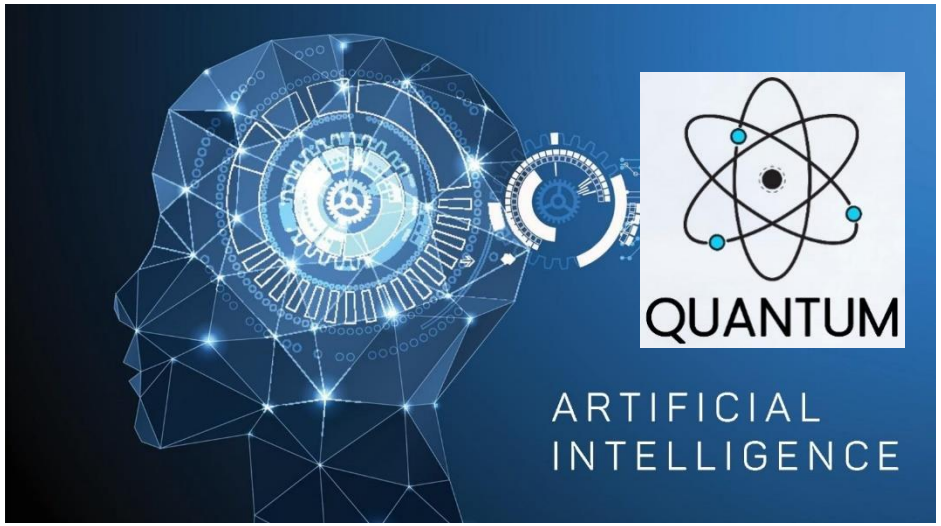
특징	고전 신경망	양자 신경망
프레임워크	Tensorflow, pytorch	Qiskit, Tensorflow Quantum, cirq, Pennylane
주 연산	행렬 곱	행렬 곱 (양자 게이트)
매개변수	가중치, 바이어스 (편향)	θ (큐비트 회전각)
활성화 함수	Relu, Swish 등의 비선형 함수	비선형 연산 사용 (양자 게이트)
최적화 함수	Adam, RMSProp	고전 신경망의 최적화 함수 (하이브리드), SPSA, COBYLA, SLSQP (양자 only)

Grover vs Quantum AI

- **Grover's algorithm을 통한 블록 암호 키 검색**
  - Grover Oracle을 공격하고자 하는 암호 알고리즘 구현
  - 주어진 평문-암호문 (*M-C*)쌍과 알려지지 않은 키 *K*
  - Oracle $f(K)\begin{cases}1, if\ Enc(M,K) = C \\ 0, otherwise\end{cases}$



- **Quantum AI를 통한 블록 암호 키 패턴 분석**
  - Classic AI 기반 블록 암호 키 패턴 분석을 **양자 컴퓨터 상에서 수행**
  - Grover는 블록 암호에 대한 확률론적 전수조사라면 **Quantum AI는 패턴 분석을 통한 확률론적 공격**
  - **Classic AI가 가진 연산 비효율성 해결**
    - 10개의 bit (10) 혹은 qubit (1024)로 한번에 표현 가능한 정보의 양의 차이로 연산 속도 및 저장 공간의 차이



Grover vs Quantum AI

- 양자 컴퓨터를 활용한 암호에 대한 공격 방법으로 Grover와 Quantum AI를 사용하는 것으로 크게 나눌 수 있음

양자 자원이 적은 Quantum AI 가 더 빠르게 실현될 가능성이 높음
양자 자원을 달성하더라도 qubit 오류 정정에 더 큰 영향을 받는 **Grover의 사용 시기는 더 늦을 것으로 예상**

Grover	Quantum AI
• 모든 key에 대한 전수조사이며 확실한 공격 • 많은 양질의 양자 자원이 필요 • Qubit에 오류가 있을 시 공격이 어려움	• 암호의 취약한 패턴을 분석하는 확률론적 공격 • 적은 qubit 수와 작은 양자 회로 필요 • Qubit에 오류가 있을 시 정확도 손실은 있으나 일정 수준 이상의 정확도를 확보할 경우 공격 가능

양자 인공지능 연구 동향

양자 신경망 연구 동향

- **Quantum만 사용하는 경우**, MNIST(이진 또는 다중 분류) 데이터 셋을 사용하여 성능을 평가하는 정도
- **대부분 Hybrid를 위한 라이브러리인 Pennylane과 IBM 시뮬레이터 및 하드웨어 사용**
  → 실제 양자 컴퓨터의 노이즈를 고려한 연구도 다수 존재

		Qubit	Device	Description
Quantum-only	[1]	8~14	Classical simulator	이진 분류, 다중 분류, 99%의 정확도
	[2]	8	Ibmqx4 (quantum hardware)	이진 분류, 99.86%의 정확도
	[3]	6	12-qubit superconducting quantum processor	2x2 이미지 데이터 사용, 데이터 생성 모델
Hybrid	[4]	4 or 9	Pennylane-Qiskit (simulator)	IBM hardware의 **노이즈를 수집하여 시뮬레이션**, 9-qubit에서 성능 저하
	[5]	4	Pennylane (simulator)	양자 레이어를 추가함으로써 레이어 개수 감소 (20 → 9)
	[6]	5	Pennylane (sim), ibmq_lima 등 (IBM hardware)	Pennylane의 여러 양자 회로에 대한 실험, **실제 양자 하드웨어 사용**

[1] Bausch, Johannes. "Recurrent quantum neural networks." Advances in neural information processing systems 33 (2020): 1368-1379.
[2] Edward Grant, Marcello Benedetti, Shuxiang Cao, Andrew Hallam, Joshua Lockhart, Vid Stojevic, Andrew G. Green, and Simone Severini. 2018. Hierarchical quantum classifiers. npj Quantum Inf. 4, 1 (2018), 1–8
[3] Huang, He-Liang, et al. "Experimental quantum generative adversarial networks for image generation." Physical Review Applied 16.2 (2021): 024051.
[4]Yang, Chao-Han Huck, et al. "Decentralizing feature extraction with quantum convolutional neural network for automatic speech recognition." *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021
[5] Houssein, Essam H., et al. "Hybrid quantum convolutional neural networks model for COVID-19 prediction using chest X-Ray images." *arXiv preprint arXiv:2102.06535* (2021).
[6] Suryotrisongko, Hatma, and Yasuo Musashi. "Evaluating hybrid quantum-classical deep learning for cybersecurity botnet DGA detection." *Procedia Computer Science* 197 (2022): 223-229.

양자 신경망 연구 동향

양자 신경망 연구 동향

Hybrid

NISQ에서 더 안정적인
hybrid 방식을 주로 사용
(Pennylane 라이브러리)

Various model

다양한 모델 개발 중
(QCNN, QRNN, QGAN etc.)

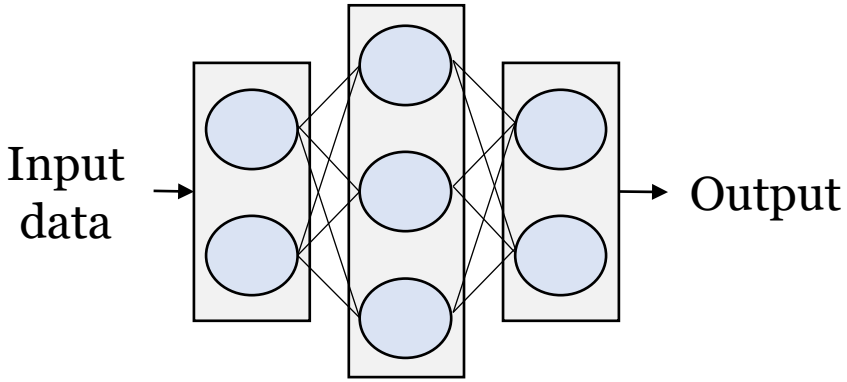
Quantum resource limitation

현재, **양자 자원 부족**으로 인해
많은 qubit 사용 어려움
(간단한 분류 작업 주로 수행)

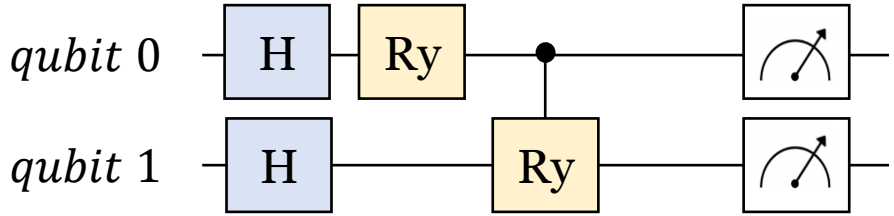
양자 신경망

양자 신경망

- **양자 역학적 현상 (얽힘, 중첩)을 활용한 인공지능**
- **고전 신경망의 학습 과정을 양자 회로로 구성한 것**
  - 양자 시뮬레이터 및 하드웨어로 실행 가능
- **양자 회로만 사용 또는 고전 신경망과 결합 가능 (하이브리드 방식)**
  - **하이브리드 방식**
    - 전체 신경망의 일부분을 양자 신경망으로 구성
      → **양자 회로를 레이어로 사용**
    - 고전 신경망은 고전 컴퓨터 (GPU)에서 연산하고, 양자 신경망은 양자 컴퓨터 (QPU)에서 연산
- 고전 신경망과 같이 입력 데이터에 대해 신경망이 동작하고 예측을 수행



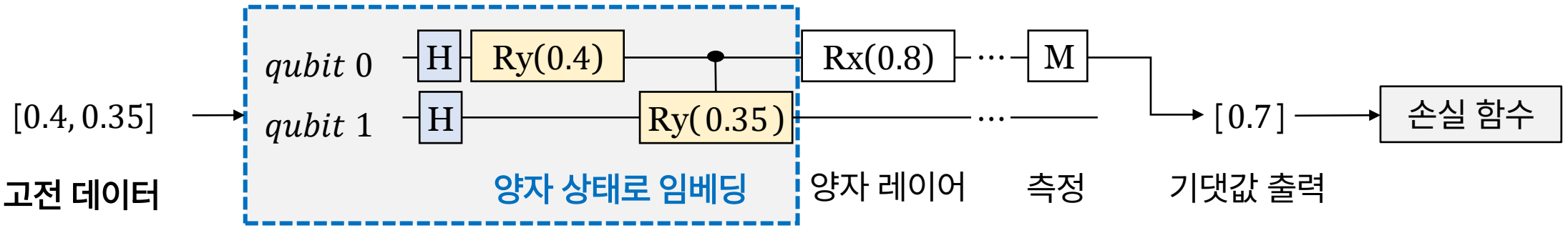
고전 신경망



양자 신경망

양자 신경망 - 하이브리드 양자 신경망

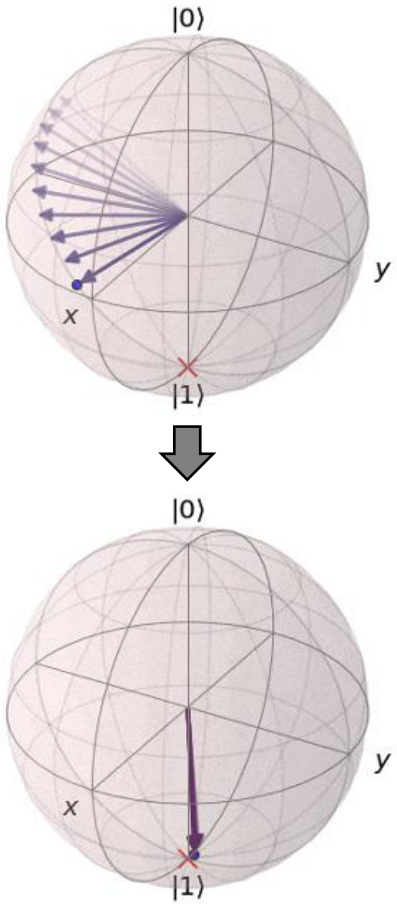
- **데이터 준비**
  - 학습에 사용하고자 하는 고전 데이터 준비
- **데이터 임베딩**
  - 고전 데이터를 양자 신경망으로 학습할 수 없으므로 데이터 임베딩 필요 → **고전 데이터를 양자 상태로 변환**
  - 해당 과정은 입력 데이터를 회전각으로 사용하므로 **parameterized circuit이 아님**
  - 입력 데이터가 큐비트의 상태에 영향을 주도록 하며, 대표적으로 다음과 같은 임베딩 방식이 있음
  - **각 임베딩, IQP* 임베딩**: n개의 특징을 갖는 고전 데이터를 n개의 큐비트에 대한 **회전 각으로 사용**
    (많은 qubit, 적은 depth)
  - **진폭 임베딩**: 2^n개의 특징을 갖는 고전 데이터를 n개의 큐비트에 대한 **진폭 벡터로 사용**
    (적은 qubit, 높은 depth)

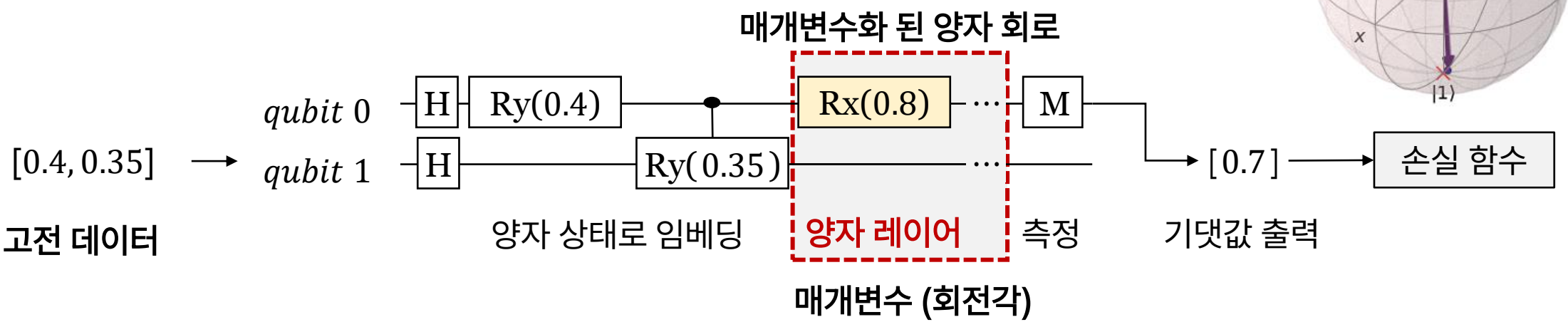


*IQP : Instantaneous quantum polynomial

양자 신경망 - 하이브리드 양자 신경망

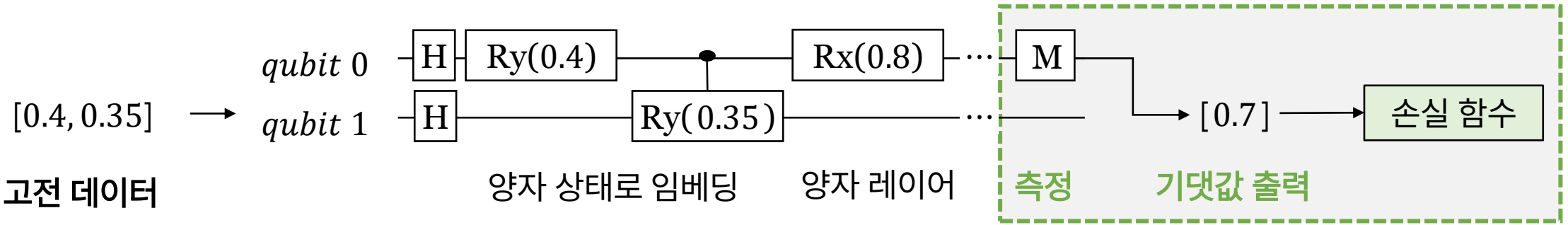
- **양자 레이어**
  - 양자 상태로 인코딩 된 후, **회전 게이트를 적용**
    - 우측 큐비트와 같이, **목표 지점에 도달하기 위해 큐비트 상태를 변화시켜가며 학습**
    - 큐비트의 상태는 **양자 게이트를 사용하여 변화**시킬 수 있음 (행렬 곱)
  - 정해진 회로 구성은 없으며, 실험을 통해 적절한 회로 구성 필요





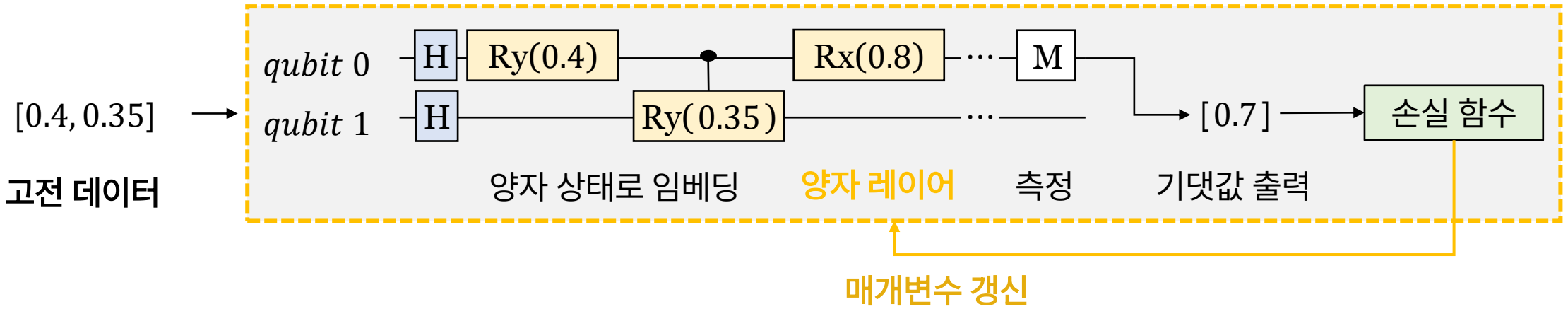
양자 신경망 - 하이브리드 양자 신경망

- **측정**
  - 0 또는 1 중 해당 값이 나올 **확률 (기댓값) 계산**
- **Shots만큼 회로를 실행 및 측정**
  - Shots=1,000이라면, 1,000번 회로 실행 및 측정이 수행
  - (큐비트 상태가 0이 나온 횟수) + (큐비트 상태가 1이 나온 횟수) = 1000
  - 더 많이 나온 상태가 최종 값으로 결정되며, 1,000번 중 900번이라면 **0.9의 확률로 1의 값**을 가지는 것



양자 신경망 - 하이브리드 양자 신경망

- **매개변수 갱신**
  - 측정을 통해 얻은 값을 기반으로 기댓값을 구한 후, 손실 함수에 입력
  - **매개변수를 조정 (큐비트 상태 변경)한 후 회로를 재실행**
    → 전체 과정 반복



양자 신경망을 위한 양자 컴퓨터 환경 및 SDK 분석

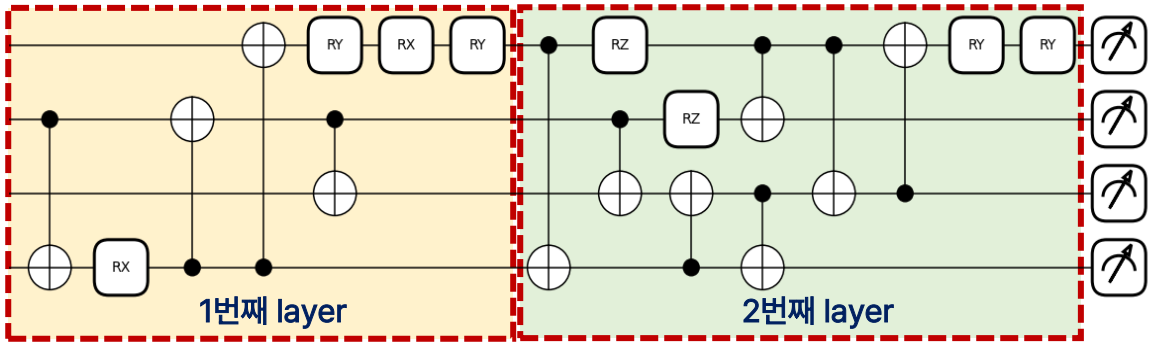
양자 인공지능을 위한 SDK 분석

- **양자 인공지능을 활용한 암호 분석에 필요한 조건**
  - 양자 회로 및 회전 게이트 (Rx, Ry, Rz) 동작 가능
  - 고전 인공지능 프레임워크 및 관련 라이브러리 지원
  - 하이브리드 신경망 구성 가능
  - 10-qubit 이상 지원: 하이브리드 신경망 사용 시 많은 qubit을 요구하지는 않음

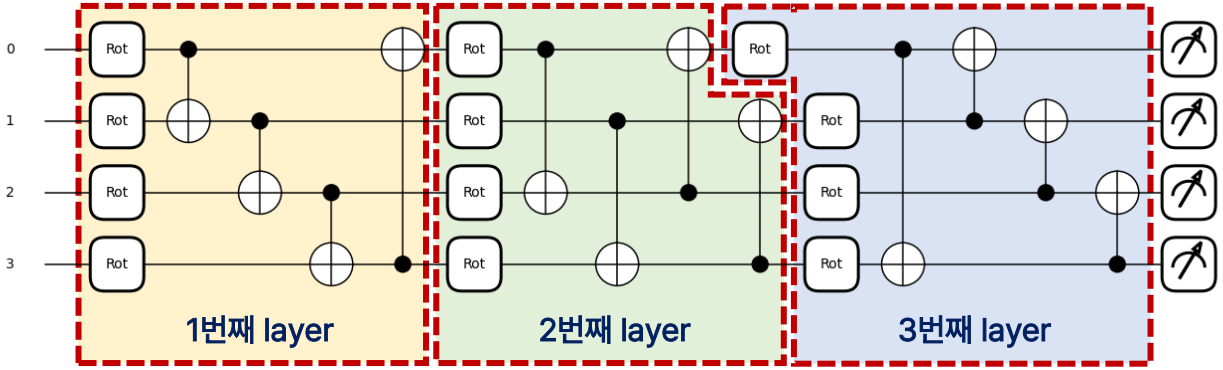
	D-wave Leap	IBM Qiskit	AWS Braket	Microsoft Azure Quantum	Pennylane
Circuit	X (annealing)	O	O	O	O
Rotation gate	X	O	O	O	O
인공지능에 사용 가능한 qubit (지원 qubit)	X (5760)	**27** (32)	**20** (25)	버그로 인해 동작하지 않음 (25)	**16** (28)
Library	X	O	X	O	O
Hybrid	X	Pytorch	Python	Python, C#	Tensorflow-keras, Pytorch
벤치마킹결과	양자 회로 실행 불가	MSE 사용 위해 multi-qubit 사용 시 버그 존재	관련 라이브러리 X, 거의 Pennylane과 연동하여 사용	라이브러리 문제, 회로 실행 불안정	모든 요소 만족, TF, Torch 지원, 정상적 학습 가능

Pennylane - Parameterized quantum circuit

- **Parameterized quantum circuit** (매개변수를 가질 수 있는 회전 게이트가 있는 양자 회로)
  - 회전 게이트 (Rx, Ry, Rz), 얽힘 게이트 $CNOT$ 사용
  - 1-layer당 qubit 수만큼의 회전 게이트 적용
  - 각 회로에는 얽힘 게이트의 차이가 있음
  - Random circuit (얽힘 비율 설정 가능), Strongly entangling circuit (풍부한 얽힘 및 회전 가능)
  - 회로 구성에는 정해진 규칙이 있는 것이 아니지만 Pennylane에서는 효과적인 회로를 제공



↑ **Random circuit** : 얽힘 비율=0.6 (전체 19개의 게이트 중 11개의 얽힘 게이트)



↑ **Strongly entangling circuit :** layer 3개인 경우

양자 인공지능을 활용한 S-DES 암호 분석

고전 신경망을 통한 암호 분석 (1차 발표 결과물)

- **암호화는 다음과 같은 성질을 가짐**

  - 단일 비트가 변경될 경우 **대부분 혹은 모든 비트에 영향** [혼돈]

  - 평문 1비트를 변경할 경우 통계학적으로 **암호문의 절반이 변경** [확산]

- 평문의 1번째 비트가 전체 암호문에 영향을 줄 수 있기 때문에 사용할 데이터는 temporal locality를 갖기 어려우며,

  전역적인 정보를 반영해야 함

  → temporal locality를 갖는 데이터의 학습에 효과적인 Convolution 및 Recurrent NN 계열이 아닌

  전역적인 정보를 고려하기 좋은 linear layer 기반의 MLP가 적절함

*Temporal : time 정보 가짐
*Locality : 인접 feature는 비슷한 정보를 가짐

[혼돈] C. E. Shannon, "Communication theory of secrecy systems," in *The Bell System Technical Journal*, vol. 28, no. 4, pp. 656-715, Oct. 1949, doi: 10.1002/j.1538-7305.1949.tb00928.x.
[확산] Stallings, William (2014). Cryptography and Network Security (6th ed.). Upper Saddle River, N.J.: Prentice Hall. pp. 67–68. ISBN 978-0133354690.

인공 신경망을 활용한 암호 분석 (1차 발표 결과물)

- **MLP (10-bit key, 8-bit plaintext and ciphertext)**

MLP	1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}	6^{th}	7^{th}	8^{th}	9^{th}	10^{th}	Params	Description
Previous Work	0.64	0.74	0.71	0.58	0.64	0.8	0.54	0.6	0.85	0.8	805,930	• 과적합 • 많은 파라미터
Residual	0.72	0.77	0.75	0.6	0.76	0.8	0.59	0.68	0.85	0.83	53,802	• 과적합 해결 • 파라미터 감소
Gated Linear Units (Best case)	0.72	0.79	0.77	0.62	0.75	0.81	0.59	0.66	0.87	0.85	55,092	• Residual에 비해 안정적 • 빠른 수렴

BAP가 낮음 (안전 비트)　　BAP가 높음 (취약 비트)

모든 비트의 BAP가 0.5 초과해야 공격 성공 (랜덤 확률이 아닌 비트를 예측)

- **비트 별 안전성**
  - **4번째, 7번째 비트**는 **안전** (60% 미만)
  - **6번째, 9번째, 10번째 비트**는 암호 분석에 **취약** (80%이상)
- Linear Neural Network (MLP) 구조에 최신 딥러닝 기술을 활용하여 암호 분석 수행
- 기본적인 MLP 기반의 이전 연구[So]에 비해 **평균적으로 5.3% 더 높은 정확도 달성, 매개변수 수는 93.16% 더 감소**

*Bit Accuracy Probability (BAP): 각 비트에 대한 예측 정확도

*Google Colaboratory (cloud service): Nvidia GPU (Tesla T4, 50GB RAM), 학습 소요 시간 : 약 5~8시간

인공 신경망을 활용한 암호 분석 (1차 발표 결과물)

- **MLP 구조의 네트워크가 CNN과 Transformer encoder에 비해 가장 좋은 성능을 보임**
  → data의 locality 보다는 **global information이 더 중요**함을 알 수 있음
  ***순서 그리고 지역적 정보 보다는 전역적인 정보를 고려할 경우 더 높은 성공률 달성**
- **정보 손실을 최소화** 하기 위해 각 레이어의 unit 수를 줄이지 않은 MLP를 사용
- **Residual connection** 적용 시 **parameter 수 현저히 감소**
- **Gated Linear Unit** 적용 시 **더 빠른 수렴 속도 및 안정적 학습 (과적합 감소)**

<한계점>

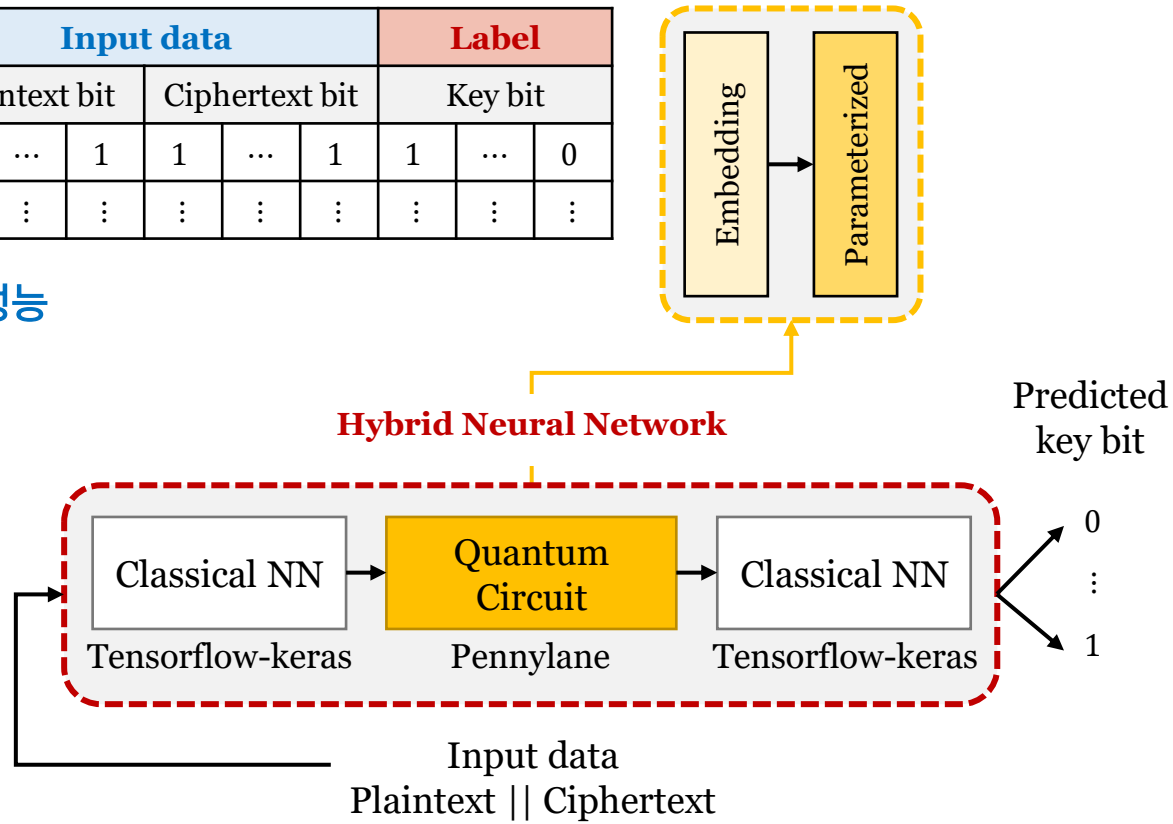
- 데이터가 복잡해질수록 네트워크는 커져야 함 (**network capacity 증가**)
- 실험 환경, 네트워크 및 데이터셋의 크기로 인해 **학습에 굉장히 많은 시간 소요**
- 키 공간이 커질수록 **전체 key bit에 대한** 암호 분석이 어려움 → 56-bit **DES에 대한 공격 확장은 불가능**할 것으로 보임

전통적인 인공지능을 이용한 암호분석의 한계점 확인

하이브리드 신경망을 이용한 알려진 평문 공격

- Dataset
  - Input data : (plaintext, ciphertext) bit
  - Label : key bit
- Hybrid network
  - Only QNN에 비해 학습 시간 절감, quantum-only 보다 더 안정적인 성능
- Library : Tensorflow-keras + Pennylane 사용
  - 양자 레이어와 고전 레이어를 결합하여 hybrid network 구성
  - 양자 레이어: 임베딩 레이어 + 파라미터화 된 양자 레이어
- 모든 비트에 대한 BAP (비트별 정확도)가 0.5를 초과하면 공격 성공
- Quantum circuit
  - Embedding
    - 얽힘 (Amplitude layer), 중첩 + 얽힘 (IQP)
  - Random circuit, Strongly entangling circuit
- Device
  - Pennylane의 default.qubit (default simulator), default.mixed (noise simulator)
  - Shots = 1000

Input data						Label		
Plaintext bit			Ciphertext bit			Key bit		
0	⋯	1	1	⋯	1	1	⋯	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮



암호 분석을 위한 하이브리드 신경망의 세부 사항 (2차 발표 결과물)

	Quantum (Random)	Quantum (Strongly)	Description
Quantum circuit	Amplitude + Random	Amplitude + Strongly	Qubit 절약을 위해 amplitude embedding 다양한 얽힘 (random) 더 많은 회전 게이트와 강한 얽힘 (strongly)
# of qubit	4	4	2-qubit은 충분한 성능 X 8-qubit은 실행 시간이 많이 소요
# of quantum layer	10	10	5-bit key까지는 5개의 quantum layer 가능 그 이상은 **10~15개** 적용 필요 (**1 epoch에 약 2~3만 초**) 20개는 학습 소요 시간 매우 증가 (**1 epoch에 약 4~5만 초**)
Architecture of classical hidden layer	128, 128, 128, 32	128, 128, 128, 64	실험 통해 적절하게 설정
# of circuit	2	4	Classical hidden layer의 구조와 # of qubit에 의해 결정 **2~4개가 적당**한 것으로 생각 더 늘릴 경우 학습 소요 시간 매우 증가
# of parameters	43,956	44,276	Quantum circuit의 parameter는 매우 적으므로 classical layer가 많을 수록 크게 증가 Classical NN의 # of parameter : 55092
# of data (train, val, test)	28500, 20000, 1500	19950, 14000, 1050	양자 신경망 학습 시간으로 인해 1차 결과물보다 적은 데이터 사용
Description	더 적은 데이터로 학습이 가능하며, 더 풍부한 회전 및 얽힘이 가능하여 classical 보다 더 높은 정확도를 얻은 Strongly entangling circuit 선택		

암호 분석을 위한 하이브리드 신경망의 세부 사항 (2차 발표 결과물)

	Quantum (Random)	Quantum (Strongly)	Description
			Qubit 절약을 위해 amplitude embedding

암호 분석을 위한 양자 신경망 세부 사항

4-qubit

성능 및 학습 시간 고려하여 선택

10~15 quantum layer

성능 및 학습 시간 고려하여 선택

Strongly Entangling circuit

Random circuit에 비해
더 높은 BAP 달성하여 선택

Using less data

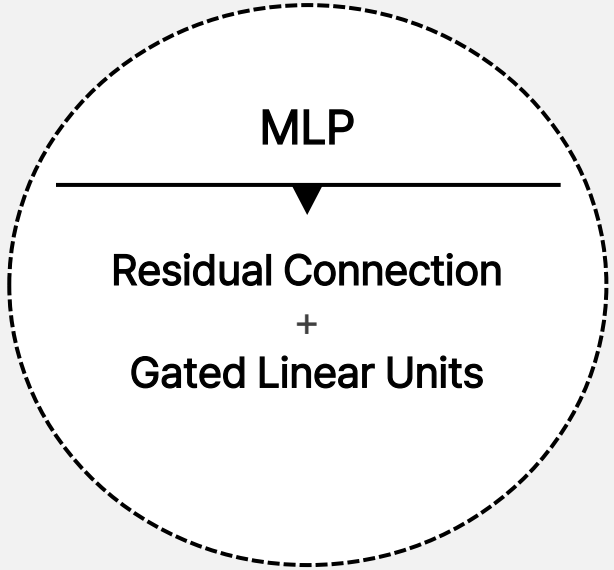
학습 시간 고려하여
전체 데이터셋 중 일부분만 사용
(19950 (학습), 14000 (검증), 1050 (테스트))

# of data (train, val, test)	28500, 20000, 1500	19950, 14000, 1050	양자 신경망 학습 시간으로 인해 1차 결과물보다 적은 데이터 사용
Description	더 적은 데이터로 학습이 가능하며, 더 풍부한 회전 및 얽힘이 가능하여 classical 보다 더 높은 정확도를 얻은 Strongly entangling circuit 선택		

Overview (3차 발표)

Classical neural network

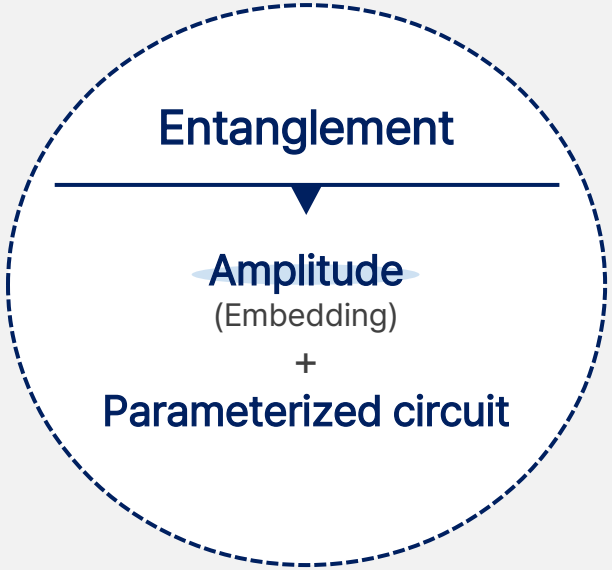
1차



MLP (Multi Layer Perceptron)
전역적인 정보를 고려할 수 있는 MLP 구조 사용
과적합 방지 및 파라미터 감소를 위한 최신 기법 적용

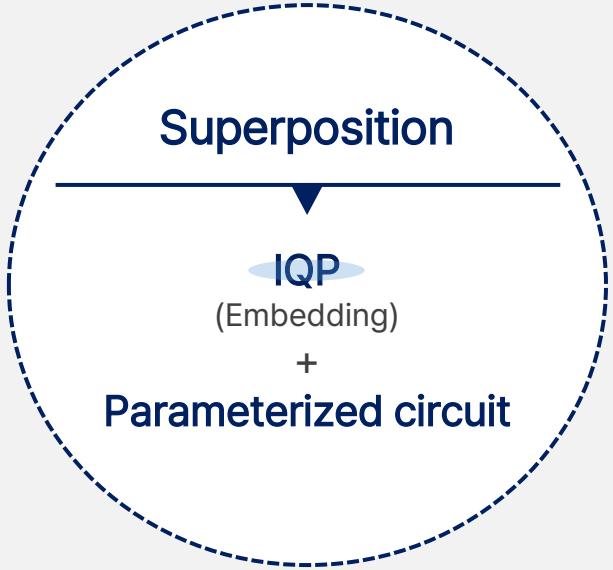
Quantum neural network using entanglement and superposition

2차



얽힘 (Entanglement)
얽힘을 통해 고전 신경망의
가중치 연결과 유사한 역할 수행*

3차



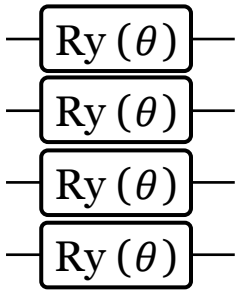
중첩 + 얽힘 (Superposition + Entanglement)
얽힘을 통한 가중치 연결 가능
여러 가중치를 중첩 상태로 병렬 연산 가능

* https://file.scirp.org/Html/19-7502330_60707.htm

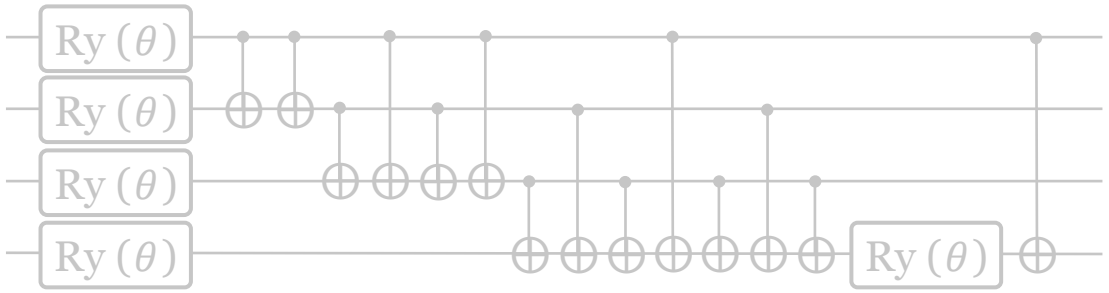
Embedding (Entanglement vs Superposition+Entanglement)

- Embedding 방식

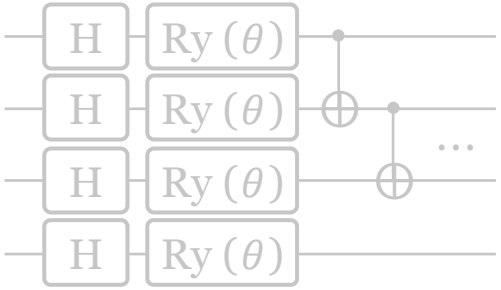
Angle embedding



Amplitude embedding



IQP embedding



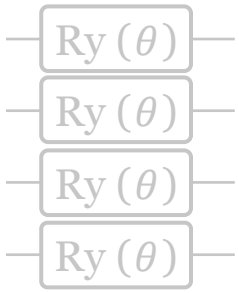
- n개의 데이터에 대해 n개의 qubit 필요
- 입력 데이터를 회전 게이트 (Rx, Ry, Rz)의 회전 각 (θ)으로 사용
- 많은 qubit, 적은 depth

간단하고 비용이 적지만 얽힘이 없으며 많은 데이터를 반영하기 어려움

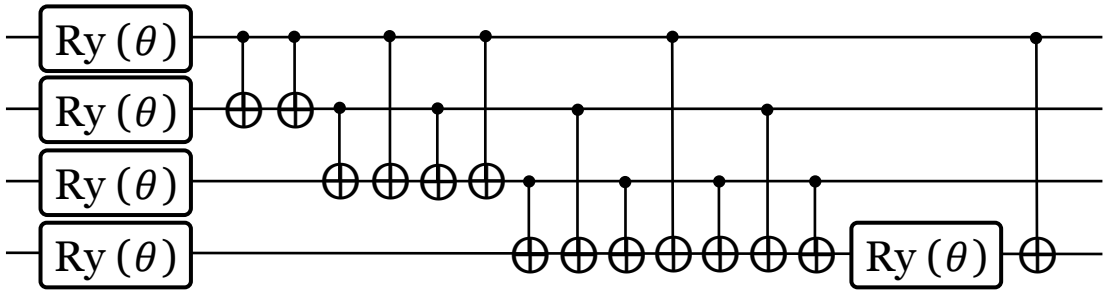
Embedding (Entanglement vs Superposition+Entanglement)

- Embedding 방식

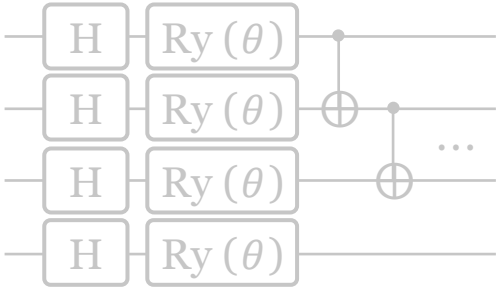
Angle embedding



Amplitude embedding



IQP embedding



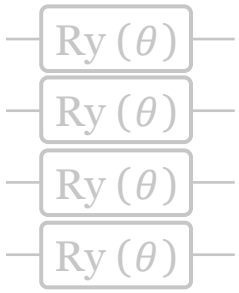
- 2^n개의 데이터에 대해 n개의 qubit 필요
  - 입력 데이터를 큐비트의 진폭벡터로 사용
  - ex) $x = (0.1, -0.7, 1.0)$이라는 데이터 벡터가 있다면 정규화 시킨 후 $(x_{norm} = 0.081, -0.571, 0.816, 0.000)$, 2-qubit의 양자 상태 $(0.081\,|00\rangle - 0.571|01\rangle + 0.816|10\rangle + 0.000|11\rangle)$로 변경
- 회전 게이트 및 얽힘을 위한 $CNOT$ 게이트 사용
- 적은 qubit, 높은 depth

많은 qubit를 동작시키기 어려운 현재의 양자 신경망에 적합한 임베딩 방식

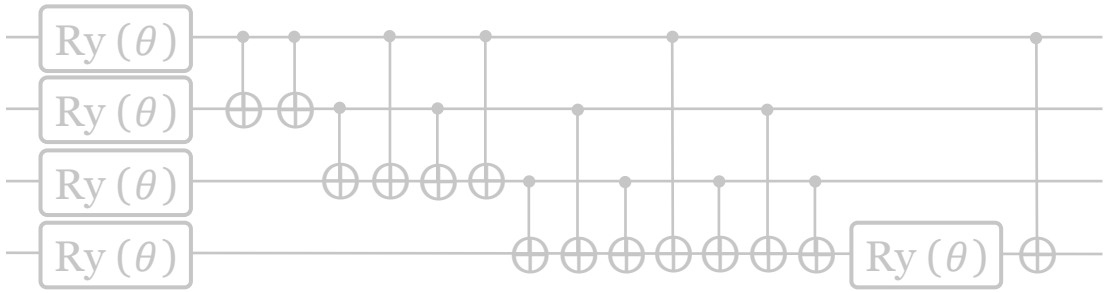
Embedding (Entanglement vs Superposition+Entanglement)

- Embedding 방식

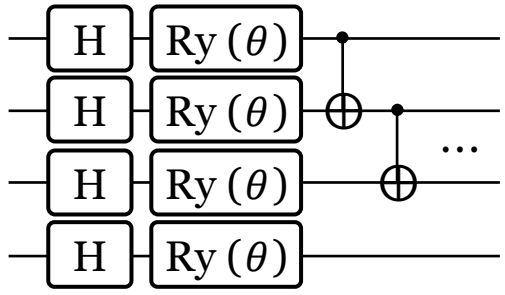
Angle embedding



Amplitude embedding



IQP embedding



- **해당 임베딩에 고전 데이터 사용 시, 입력 데이터에 대한 중첩이 아닌 가중치를 중첩시켜 여러 가중치에 대해 한번에 학습 가능**
  - **n개의 데이터**에 대해 **n개의 qubit** 필요 (입력 데이터를 회전 게이트의 회전 각으로 사용)
- 즉, amplitude 임베딩에 비해서 동일 qubit으로 **반영할 수 있는 데이터 특징 (feature)가 적음**
  → **더 많은 feature를 반영하고 싶다면 qubit의 수를 늘려야 함**
  - 그러나 현재의 양자신경망에서 동작 가능한 **최대 qubit 수는 16개** (임베딩 방식과 상관 없음)
  - **그럼 16개의 qubit을 사용한다면?** 그러나, 중첩이 없는 경우에 비해 **2배**의 시간이 더 소요
- 처음부터 중첩 상태의 양자 데이터 생성 시, 데이터에 대한 중첩 가능할 것 (2^n개의 데이터에 대해 n개의 qubit 필요하지만 **하이브리드에 사용 불가**)
- **많은 qubit, 조절 가능한 depth**

중첩을 활용한 고전 데이터 임베딩이 가능하지만, 현재의 양자 컴퓨터에서는 많은 feature 반영은 어려움

Epoch에 따른 BAP 분석 (S-DES)

안전 비트	$0.5 < BAP < 0.6$
일반 비트	$0.6 < BAP < 0.8$
취약 비트	$0.8 < BAP$

Classical

# of data : Tr (19950), Val (14000), Ts (1050)

Epoch \ Bit	1	2	3	4	5	6	7	8	9	10	Avg.
15	59	59	65	53	54	80	54	56	80	72	63.2
20	60	60	65	51	54	81	51	56	82	72	63.2
25	62	60	65	53	55	81	54	56	83	70	63.9
30	61	60	67	56	54	80	52	59	83	74	64.6
100	60	66	69	57	56	81	53	60	86	80	66.8

Quantum의 자원 및 시간의 한계로 인해 1차 결과보다 데이터 수를 줄인 후 비교

- 모든 epoch에 대해, 안전 비트 (4, 7), 취약 비트 (6, 9)
- 100 epoch :
  10번째 비트가 취약 비트로 검출
  8번째 비트가 안전 비트에서 제외
- 1, 2번째 비트 :
  15 epoch에서는 안전 비트였으나, 학습이 진행됨에 따라 탈락

Quantum Amplitude (Entanglement)

Epoch \ Bit	1	2	3	4	5	6	7	8	9	10	Avg.
20	53	52	65	52	54	80	53	56	83	71	61.9
25	61	61	65	55	52	80	54	56	82	74	64.0
30	64	71	71	56	51	80	56	60	85	80	67.4

양자 자원 및 시간의 한계로 인해 30 epoch까지 실행 가능

- 모든 epoch에 대해, 안전 비트 (4, 5, 7), 취약 비트 (6, 9)
- 30 epoch :
  10번째 비트가 취약 비트로 검출, 8번째 비트가 안전 비트에서 제외
  Classical 보다 적은 epoch에서 달성
- 동일 epoch에서 평균 BAP가 Classical에 비해 2.8% 높으며, 최대 성능에서 0.6% 향상
- 20~30 epoch 구간에서 classical에 비해 BAP 상승 폭이 큼
  Quantum (2.1%, 3.4%), Classical (0.7%, 0.7%)

Quantum IQP (Superposition + Entanglement)

Epoch \ Bit	1	2	3	4	5	6	7	8	9	10	Avg.
20	53	63	55	56	48	79	54	52	83	54	59.7
25	52	55	59	53	51	79	51	52	82	57	59.1
30	58	65	60	55	54	79	51	57	84	75	63.8

- 모든 epoch에 대해, 안전 비트 (1, 4, 5, 7, 8), 취약 비트 (9)
- 임베딩 가능한 feature가 적으므로 다른 방법들에 비해 BAP가 낮음
  데이터가 아닌 가중치가 중첩 → 반영 가능한 feature 수가 적음
  안전 비트가 많고 취약 비트가 적게 검출
  5번째 비트는 20 epoch에서는 분석 실패

Epoch에 따른 BAP 분석 (S-DES)

안전 비트	$0.5 < BAP < 0.6$
일반 비트	$0.6 < BAP < 0.8$
취약 비트	$0.8 < BAP$

Classical

# of data : Tr (19950), Val (14000), Ts (1050)

Epoch \ Bit	1	2	3	4	5	6	7	8	9	10	Avg.
15	59	59	65	53	54	80	54	56	80	72	63.2
20	60	60	65	51	54	81	51	56	82	72	63.2
25	62	60	65	53	55	81	54	56	83	70	63.9
30	61	60	67	56	54	80	52	59	83	74	64.6
100	60	66	69	57	56	81	53	60	86	80	66.8

Quantum Amplitude (Entanglement)

Epoch \ Bit	1	2	3	4	5	6	7	8	9	10	Avg.
20	53	52	65	52	54	80	53	56	83	71	61.9
25	61	61	65	55	52	80	54	56	82	74	64.0
30	64	71	71	56	51	80	56	60	85	80	67.4

- 모든 epoch에 대해, 안전 비트 (4, 5, 7), 취약 비트 (6, 9)
- 30 epoch :
  10번째 비트가 취약 비트로 검출, 8번째 비트가 안전 비트에서 제외
  Classical 보다 적은 epoch에서 달성
- 동일 epoch에서 평균 BAP가 Classical에 비해 2.8% 높으며, 최대 성능에서 0.6% 향상
- 20~30 epoch 구간에서 classical에 비해 BAP 상승 폭이 큼
  Quantum (2.1%, 3.4%), Classical (0.7%, 0.7%)

Epoch에 따른 BAP 분석 (Classical, Quantum 공통)

- 모든 epoch에 대해, 안전 비트 (4, 7), 취약 비트 (9)
  - 모든 방법에서 안전 및 취약 비트에 대해 비슷한 경향을 보임
  - 9번째 비트는 양자 자원 제한으로 인해 적은 feature만 반영할 수 있는 IQP (중첩+얽힘)에서도 높은 BAP를 달성했으므로 매우 취약한 비트
- Epoch에 따라 BAP가 전반적으로 상승
  - 학습이 진행됨에 따라 예측 확률이 높아짐
    → 안전 비트 수 감소, 취약비트 수 증가
- 가장 적합한 방법 : Quantum Amplitude 방식 (얽힘만 사용)
  - 동일한 30 epoch에서 안전 비트의 수가 가장 적고 취약비트의 수가 가장 많음
  - 평균 BAP가 가장 높음

데이터가 아닌 가중치가 중첩 → 반영 가능한 feature 수가 적음
안전 비트가 많고 취약 비트가 적게 검출
5번째 비트는 20 epoch에서는 분석 실패

Classical vs Quantum

- **Avg., Epoch, Params를 모두 고려하여 비교한 결과**

*동일한 데이터 수 사용

Method \ Bit	1	2	3	4	5	6	7	8	9	10	Avg.	Epoch	Params
Classical	60	66	69	57	56	81	53	60	86	80	66.8	100	55092
Amplitude	64	71	71	56	51	80	56	60	85	80	**67.4**	**30**	44276
IQP	58	65	60	55	54	79	51	57	84	75	63.8	30	38084

- **각 요소에 대한 순위 비교**

	1st	2nd	3rd
Avg. (Average BAP)	Amplitude	Classical	IQP
안전 비트 수 / 취약 비트 수	Classical (3/3) Amplitude (3/3)		IQP (5/1)
Epoch	Amplitude, IQP		Classical
Params (The number of parameters)	IQP	Amplitude	Classical

- Classical
  - Avg.는 2등이지만, **많은 epoch과 params가 필요**
- **Amplitude (가장 적합)**
  - Avg.가 가장 높으며, 적은 epoch과 params가 필요
  - **70 epoch & 파라미터 20% 감소, BAP 0.6% 향상 (양자 이점 달성)**
- IQP
  - Epoch과 params는 적지만 학습이 덜 된 상태라서 Avg.가 낮음
    → **현재 양자 컴퓨터로 충분한 학습이 불가능**하므로 부적합한 방식
- **얽힘 (Amplitude) vs 중첩+얽힘 (IQP)**
  - 현재는 양자컴퓨터 자원의 한계로 인해 **중첩이 사용될 경우 더 적은 feature만을 반영할 수 있어서 성능이 좋지 않음**
  - 향후, **더 많은 qubit 사용 가능 + 회로 실행 가속화**가 충분히 가능하다면 **중첩이 적용된 양자 신경망이 더 큰 이점을 가질 것으로 예상**

Classical vs Quantum

- **Avg., Epoch, Params를 모두 고려하여 비교한 결과**

*동일한 데이터 수 사용

Method \ Bit	1	2	3	4	5	6	7	8	9				
Classical	60	66	69	57	56	81	53	60	86				
Amplitude	64	71	71	56	51	80	56	60	85				
IQP	58	65	60	55	54	79	51	57	84	75	63.8	30	38084

Qubit는 0, 1 뿐만 아니라
블로흐 구면 위의 값들을 표현 가능*하므로
범위가 넓어서 더 정교한 가중치 표현 가능

양자 신경망이 고전 신경망보다 더 좋은 성능을 달성 (Qubit의 얽힘과 풍부한 표현 범위로 인한 이점)

현재는 얽힘만 사용한 양자 신경망이 암호 분석에 가장 효과적이며, 중첩을 충분히 활용하기 위해서는 더 큰 규모의 양자 컴퓨터 필요

(Average BAP)	Amplitude	Classical	IQP
안전 비트 수 / 취약 비트 수	Classical (3/3) Amplitude (3/3)		IQP (5/1)
Epoch	Amplitude, IQP		Classical
Params (The number of parameters)	IQP	Amplitude	Classical

- 70 epoch & 파라미터 20% 감소, BAP 0.6% 향상 (양자 이점 달성)
- **IQP**
  - Epoch과 params는 적지만 학습이 덜 된 상태라서 Avg.가 낮음
    → **현재 양자 컴퓨터로 충분한 학습이 불가능**하므로 부적합한 방식
- **얽힘 (Amplitude) vs 중첩+얽힘 (IQP)**
  - 현재는 양자컴퓨터 자원의 한계로 인해 **중첩이 사용될 경우 더 적은 feature만을 반영할 수 있어서 성능이 좋지 않음**
  - 향후, **더 많은 qubit 사용 가능 + 회로 실행 가속화**가 충분히 가능하다면 **중첩이 적용된 양자 신경망이 더 큰 이점을 가질 것으로 예상**

양자 컴퓨터의 노이즈를 고려한 암호 분석

- Pennylane에서 제공하는 default.mixed 시뮬레이터 사용
  - 노이즈 모델을 시뮬레이션하기 위해 노이즈 채널을 적용

	1	2	3	4	5	6	7	8	9	10	Avg.	Epoch
Amplitude												

S-AES 암호 분석

- 암호 분석은 입력 데이터의 차원도 크고 (평문 크기x2), 분석을 위해 수많은 데이터가 필요
- 그러나 S-DES에 대한 분석에도 많은 시간이 소요
  - 현재 자원으로는 전체 데이터셋 학습에 어려움이 있어 35000개의 데이터만 사용
  - 1-epoch에 7~8시간 (Amplitude), 14시간 (IQP) 소요 (30 epoch에 9~10일 소요)
- S-AES에 대한 분석을 위해 키 공간 및 데이터의 수를 줄여서 실험
  - 4-bit 키 공간, 35000개의 데이터 사용 시, 1-epoch에 7~8시간 소요 (데이터 수 부족으로 인해 일부 키 비트 분석 실패)
  - 키 공간을 늘릴 경우 더 많은 데이터를 사용해야 함
  - 11-bit 키 공간에 대한 분석을 위해서는 최소 140만개의 데이터가 필요* → 1-epoch에 최소 40일 소요**
- Classical의 결과에서 알 수 있듯이 S-AES를 위해서는 S-DES에 비해 더 많은 파라미터가 필요
  - 따라서 Quantum에서도 더 많은 파라미터가 필요
  - 이에 맞게 양자 레이어, qubit, 양자 회로 수를 늘릴 경우, 1-epoch에 $40 \times \alpha$일 소요**

*1차 결과물 참고
**2차 결과물의 소요 시간 분석 참고

현재의 양자 신경망으로는 S-AES 이상의 암호 분석은 어려움
(충분한 데이터 셋 사용 불가, 많은 qubit 사용 어려움, 매우 큰 학습 소요 시간)

Quantum AI를 활용한 암호 분석 방향성 제시

양자 인공지능 개발의 한계점

관련 라이브러리 업데이트가 활발
- 개발 환경이 불안정
- 에러에 관한 정보들이 많지 않음

양자 자원 활용 제약 및 실험의 어려움
- 사용 가능한 qubit의 수가 적음
- 학습 소요 시간이 매우 커서 충분한 실험 불가능

양자 하이브리드 신경망 기반의 암호 분석 (S-DES)

암호 분석 결과 (Classical, Quantum 공통)
- 취약 비트 : 6, 9, 10 (9가 가장 취약)
- 안전 비트 : 4, 7
- 학습이 진행될수록 취약 비트가 많이 검출됨

Quantum Advantage
- 동일 epoch에 대해 2.8% 더 높은 BAP 달성, 파라미터 20% 감소
- 최대 BAP에 대해 70 epoch & 파라미터 20% 감소, BAP 0.6% 향상
- 즉, 더 적은 epoch으로 더 많은 취약 비트 검출 가능

현재 양자 컴퓨터의 한계점으로 인한 어려움
- 축소된 데이터 셋 사용, 중첩 성질 활용 어려움, 적은 큐비트 사용

S-AES 이상의 암호에 대한 분석은 현실적으로 불가능
- 위와 같은 한계점들로 인해 충분한 데이터 셋 및 qubit 활용 불가

현재의 양자 컴퓨터를 활용한 양자 인공지능 기반의 암호 분석은 시기상조

향후, 더 안정적이고 큰 규모의 양자 컴퓨터가 개발된다면 S-DES 이상의 암호에 대한 분석이 가능할 것으로 예상

Q & A