포트폴리오

24년 가을학기 과정 지원

지원자 : 박명규

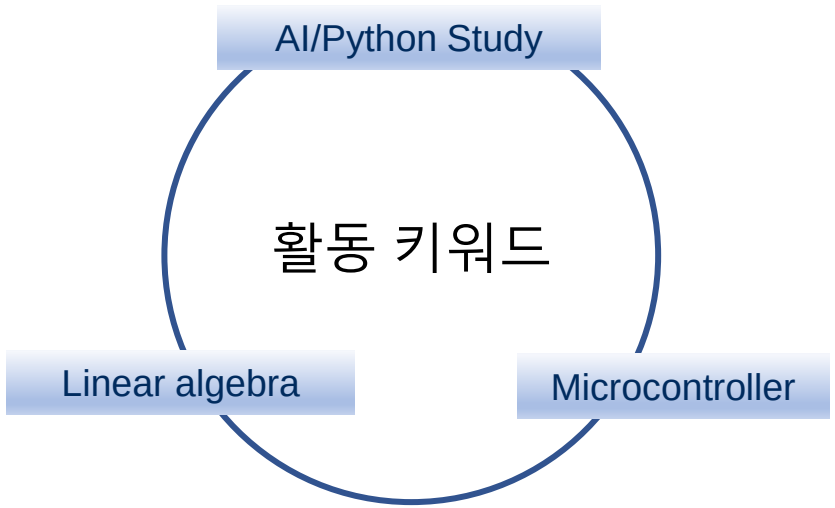
목차

SMART (2018 ~ 2021)

SMART란?

- 동양미래대학교 로봇 전공 동아리로 황일규 교수님을 지도교수님으로 두고 있습니다.
- 주로 마이크로 컨트롤러를 사용하여 기구와 결합된 인공지능 서비스를 제작하며 경진대회 수상을 목표 활동합니다.

1-1. SMART (2018 ~ 2021)



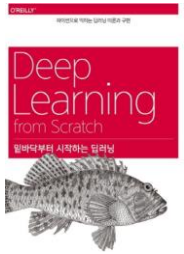
대회 준비

- 기간 : 2021/03 ~ 2021/09
- 목표 : 다양한 사용자의 맞춤형 키오스크 개발
- 결과 : 기술창업 아이디어 경진대회 대상

선형대수 스터디

- 기간 : 2020/09 ~ 2021/03
- 내용 : 인공지능을 위한 선형대수 공부
- 결과 : Numpy 라이브러리 이해도 증가

AI/Python 스터디

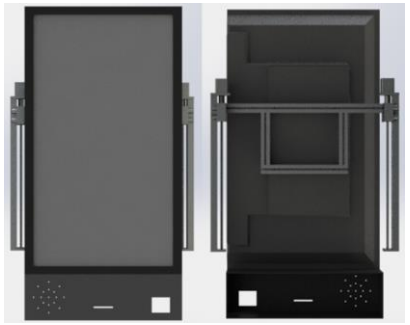


- 기간 : 2020/09 ~ 2021/03
- 목표 : CNN 이론, Python 및 Numpy 공부
- 결과 : 심층 신경망에 내부 구조에 대한 이해

1-1. SMART (2018 ~ 2021)

대회 준비

▣ 유니버셜 키오스크



○ 주제 선정 이유

- 디지털 취약 계층과 키오스크 높이로 인해 사용하지 못하는 어린이, 휠체어 사용자를 위한 키오스크 제작

▣ 역할

- 센서를 통해 사용자의 키를 판단한 후 모터를 조정하여 키오스크 높이 조절 기능 구현
- 종이에 주문을 적어 제출해도 주문이 가능하도록 OCR 기능 및 문장 파싱 기능 구현
- Solidworks를 통한 제품 디자인
- [결과] : 기술창업 아이디어 경진대회 대상

Stack

python

TensorFlow

Firebase

Pseudo Lab (2021 ~ 2022)

Pseudo Lab이란?

- 2020년에 결성된 비영리 커뮤니티로 기초 이론 공부, 논문 리뷰, 연구개발 프로젝트 등의 다양한 활동들을 진행합니다.
- 후원사로는 NIPA 정보통신산업진흥원, Google Cloud, Superb AI, MakinaRocks 등이 있습니다.

1-2. Pseudo Lab (2021 ~ 2022)

GAN 프로젝트

- 기간 : 2021/08 ~ 2021/12
- 목표 : GAN 논문 스터디 및 대회 준비
- 결과 : Kaggle 10% 진입, AI X ART 특별상

경량화 프로젝트

- 기간 : 2022/03 ~ 2022/06
- 내용 : depth estimation 모델 경량화 논문 스터디
- 결과 : MAI Workshop 참가

경량화 스터디

- 기간 : 2022/09 ~ 2022/12
- 목표 : NLP, CV 등 다양한 분야의 경량화 기술 적용
- 결과 : TFLite 사용법에 익숙해졌으며 경량화 분야의 전반적인 기술들을 알게되었습니다.

1-2. Pseudo Lab (2021 ~ 2022)

GAN 스터디

▣ 반 고흐 전시전

▣ 역할 (Kaggle)

- 모델 선정 (pix2pix), style 정보만 반영하는 것이 중요해 선택했습니다.
- 데이터 셋이 적은 문제를 해결하기 위해 2개의 판별자를 도입했습니다.
- 모델 평가 코드 자동화 했습니다.
- [결과] : Kaggle 10%, 동메달

Kaggle result	
Initial score	43.71
final score	34.82

▣ 역할 (AI X ART)

- 주제 선정 (그림 속 옛 명인들의 실제 얼굴)
- 동양인 얼굴 데이터 크롤링
- [결과]: 2021 AI X ART 특별상

1-2. Pseudo Lab (2021 ~ 2022)

경량화 프로젝트

▣ Mobile AI Crew (MAI Workshop)

Mobile AI Crew

↙ 백링크 1개

딥러닝 모델 경량화 기술을 스터디하고, Depth estimation network에 적용하여 효과도를 분석, 이론과 실무 지식을 동시에 공부하는 것이 우리 크루의 목적입니다.
또한 MAI workshop의 challenge에 참가하여 국내 데이터 사이언스와 머신러닝 커뮤니티에 기여하고자 합니다.

▣ 역할

- 데이터 전처리
- 논문 리뷰 및 실험 결과 발표
- 평가지표 코드 구현

▣ 핵심 아이디어

$$\text{Final Score} = \frac{2^{-20 \cdot \text{si-RMSE}}}{C \cdot \text{runtime}}$$

- Final Score에서 runtime이 비중이 크기 때문에 다른 성능 지표에서 불이익이 생기더라도 inference 속도를 빠르게 하는 것에 중점을 뒀습니다.
- Pruning 후 model compression을 사용하여 연산량을 큰 폭으로 줄입니다.

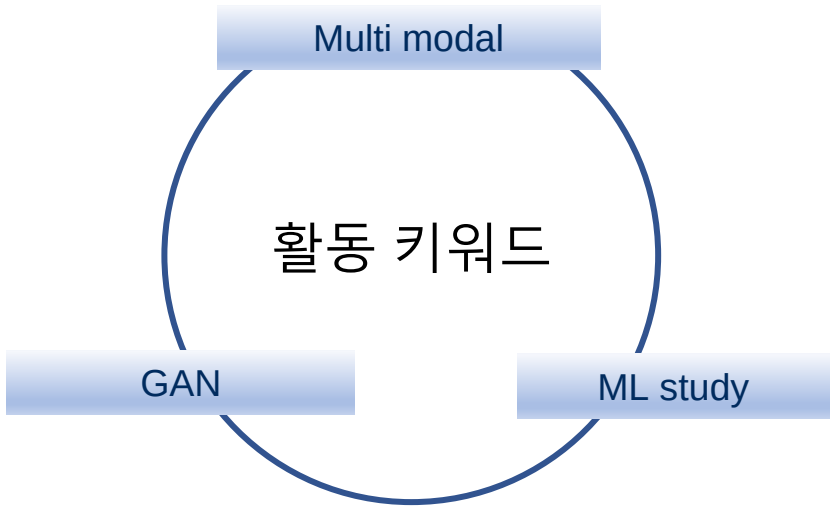
BOAZ (2022 ~ 2023)

BOAZ란?

BOAZ

- BOAZ는 국내 최초 대학생 연합 빅데이터 동아리로 연세대 이원석 교수님을 지도 교수님으로 두고 있습니다.
- 매주 정기세션을 통해 데이터 분석 / 데이터시각화 / 데이터 엔지니어링에 대해 공부합니다.
- 매년 1회 오프라인 컨퍼런스를 통해 6개월 동안 제작한 결과물을 발표합니다.

1-3. BOAZ (2021 ~ 2022)



ML Study

branch	세션 내용
week_0	pandas, numpy, matplotlib, EDA 기초
week_1	분류 및 회귀, 언더피팅과 오버피팅
week_2	SVM, 결정나무
week_3	앙상블, 랜덤포레스트, 부스팅, 스태킹
week_4	머신러닝 리뷰, 딥러닝기초, CNN
week_5	RNN, LSTM
week_6	GAN, 강화학습

- 기간 : 2022/06 ~ 2022/08
- 목표 : 7주동안 ML의 기본 지식 공부
- 결과 : 정형 데이터에 사용하는 선형/비선형 모델 숙지

Mini Project

- 기간 : 2022/03 ~ 2022/06
- 내용 : GAN을 통한 이미지 업스케일링 서비스
- 결과 : Super resolution 서비스 구축

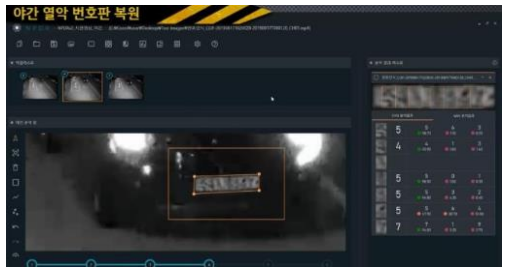
BOAZ Conference

- 기간 : 2023/01 ~ 2023/07
- 목표 : Multimodal을 사용한 FakeDetection 서비스 구축
- 결과 : 기존 NLP만을 사용하는 방법보다 더 높은 평가 지표를 얻었습니다.

1-3. BOAZ (2021 ~ 2022)

Mini Project

▣ Super resolution with SRGAN



- 문제: 노후화나 비용적 문제로 저해상도 이미지를 쉽게 접할 수 있게 됐습니다.

GAN을 통해 노후화된 설비 문제나
환경등의 이유로 발생한 저해상도 문제를
해결할 수 있습니다.

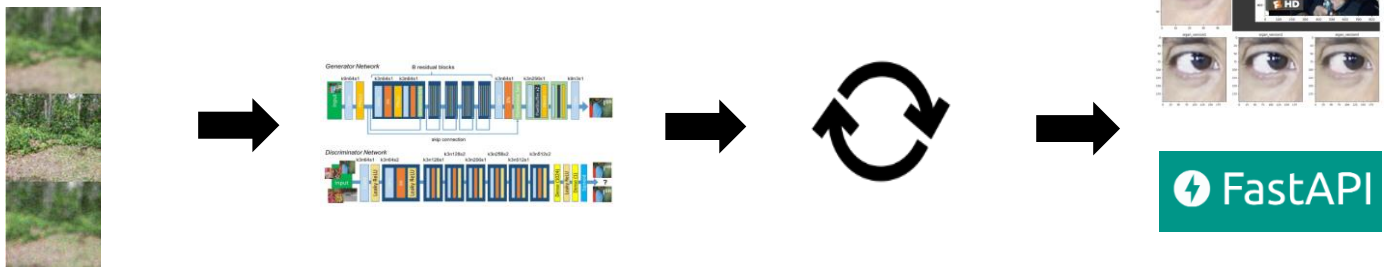
▣ 방법

○ 데이터 셋

Kaggle Super Resolution Dataset

고해상도 이미지를 블러 처리하여 저해상도로
만들어서 학습을 진행했습니다.

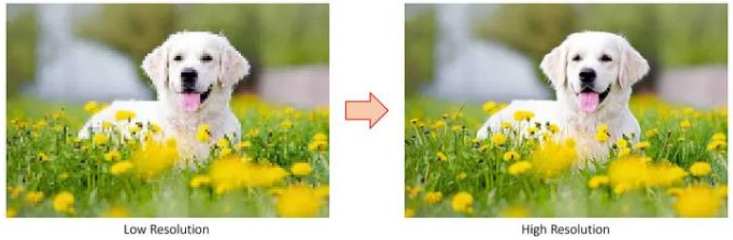
○ 모델 개발 방법 및 순서



1-3. BOAZ (2021 ~ 2022)

Mini roject

▣ Project result

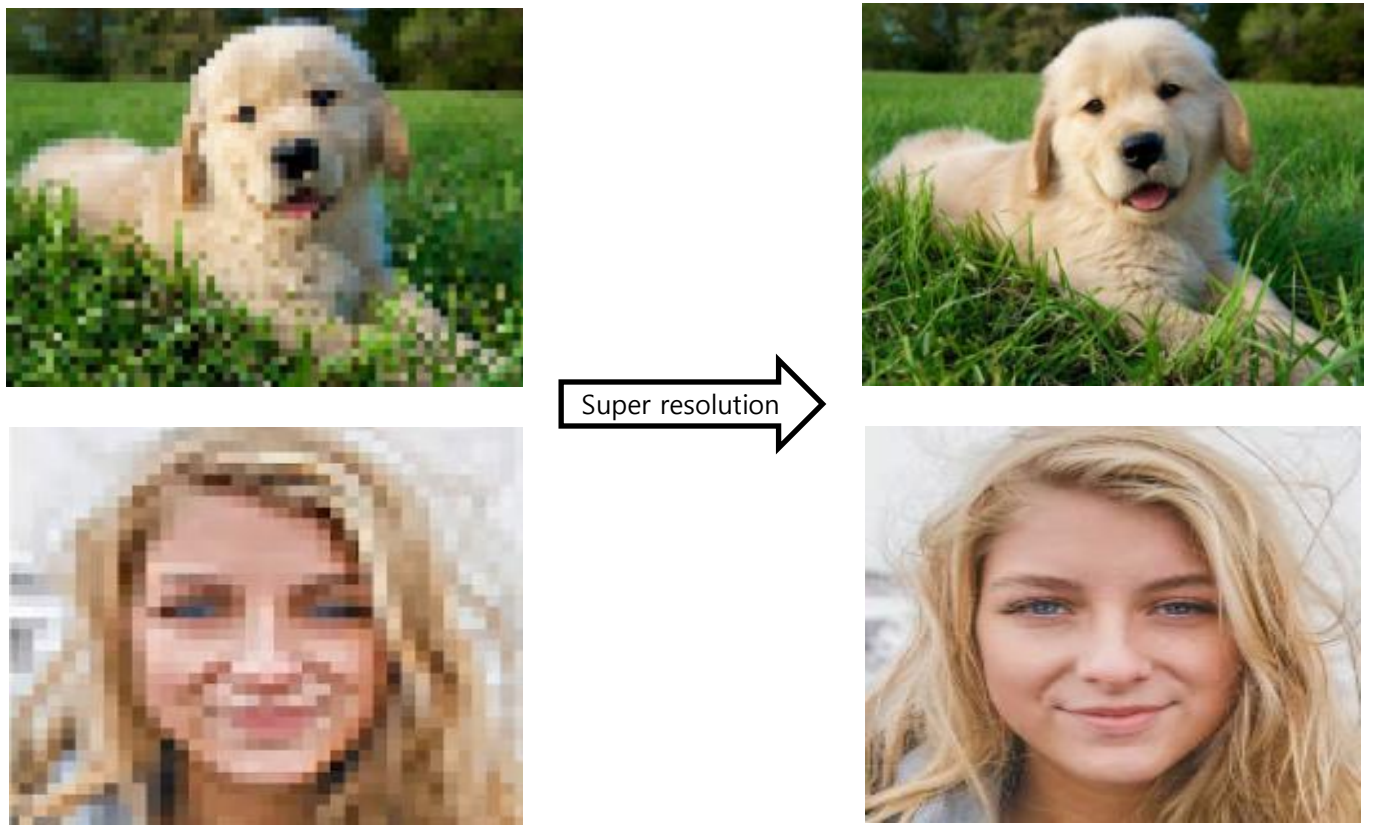


SSIM	PSNR
0.8223	27.20

$$SSIM(x, y) = [l(x, y)]^{\alpha} \cdot [c(x, y)]^{\beta} \cdot [s(x, y)]^{\gamma}$$

$$\begin{aligned} PSNR &= 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \\ &= 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \\ &= 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE) \end{aligned}$$

결과



1-3. BOAZ (2021 ~ 2022)

BOAZ Conference

▣ Multi-modal Fakenews Detection

- 문제1: 실제 사람들이 가짜뉴스인지 아닌지를 맞추는 정답률은 평균 58.3%
- 문제2: 점점 정교화된 가짜뉴스들이 늘어나는 추세

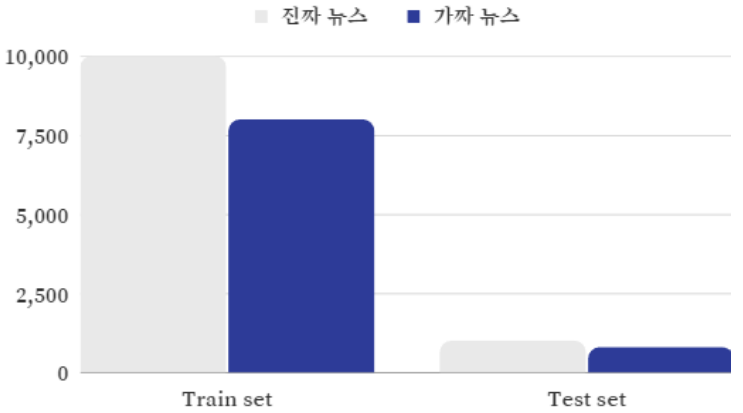
Muliti-modal을 활용한 높은 성능의 Fakenews Detection 서비스를 제공하여 해결하자!

▣ 방법

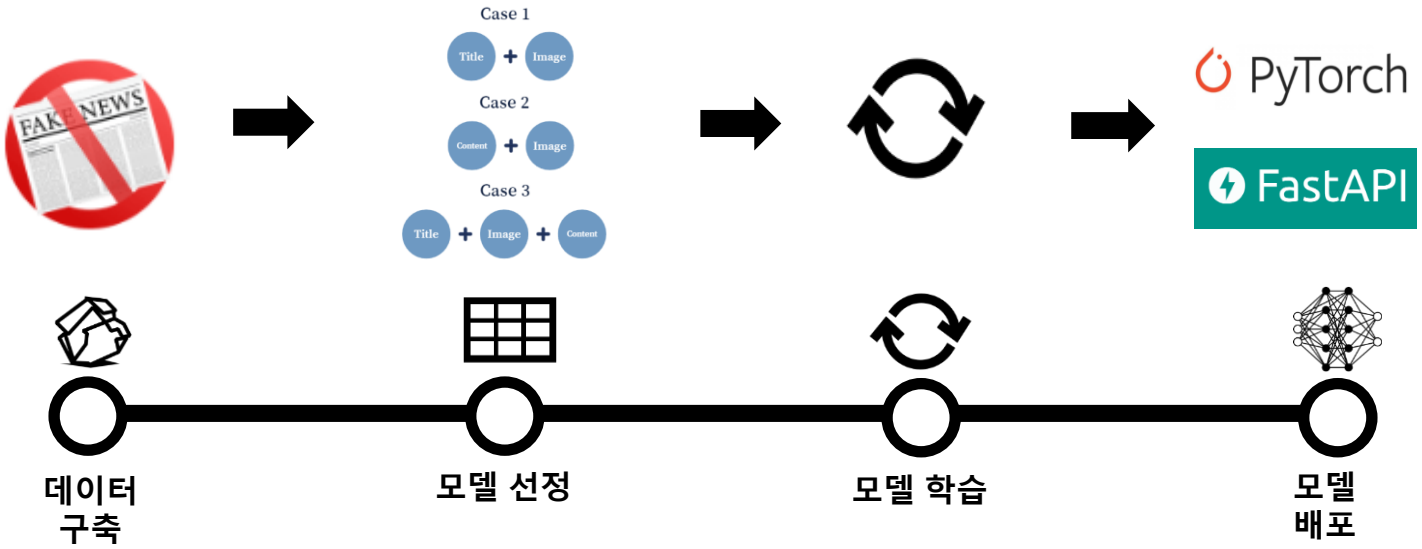
○ 데이터 셋

Fact Check 기사를 크롤링 했습니다.

뉴스의 본문, 제목, 본문 내 이미지를 크롤링하여 데이터 셋을 구축했습니다.



○ 모델 개발 방법 및 순서



1-3. BOAZ (2021 ~ 2022)

BOAZ Conference

▣ Project result

Sentence-BERT + ResNet152
뉴스 제목 + 이미지

Longformer + ResNet152
뉴스 본문 + 이미지

Sentence-BERT + T5 + ResNet152
뉴스 제목 + 본문 + 이미지

고전적인 Fast text를 사용한 detection과 Multi modal을 적용한 3가지 모델을 비교, ResNet과 Longformer의 조합이 가장 좋은 성능을 보였습니다.

Experiment

- batch size : 256
- learning rate : 1e-4(0.0001)
- epoch : 10
- dropout : 0.1
- optimizer : Adam

평가 지표

- F1-score
- AUROC
- Accuracy

	Models	AUROC	Accuracy	F1-score
Uni-modal	Fasttext (Baseline)	0.7205	-	-
	SBERT	0.7323 (+0.0118)	0.6912	0.6905
Multi-Modal	ResNet + SBERT	0.8186 (+0.0981)	0.7423	0.7330
	ResNet + SBERT + T5	0.8464 (+0.1259)	0.7558	0.7572
	ResNet + Longformer	**0.8542 (+0.1337)**	**0.7759**	**0.7740**

Undergraduate Internship

(2022 ~ 2024)

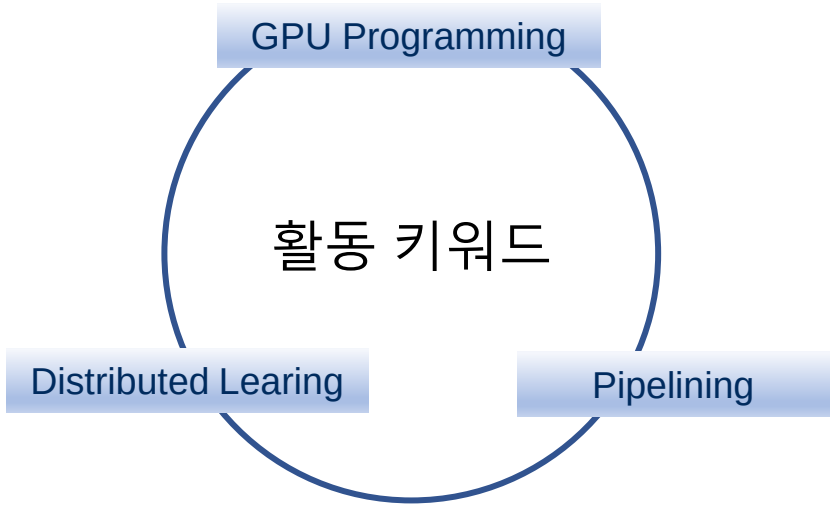
Undergraduate Internship

Computer Systems Laboratory

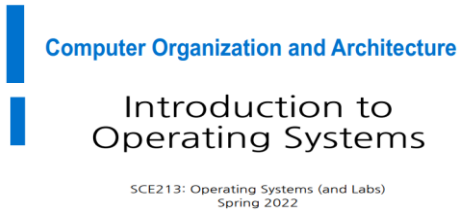
College of Computing and Informatics

- 아주대학교 Computer System Lab에서 인턴 활동을 하였으며 저의 지도교수님은 안정섭 교수님이였습니다.
- System for ML을 주제로 Throughput을 최대화하기 위한 Scheduling, GPU 전체 통합 메모리 관리에 대해 공부했습니다.

1-4. Internship (2022 ~ 2024)



CS Study



- 목표 : OS, Computer architecture 이해
- 결과 : 이론 학습 후 C를 사용하여 Toy assembler, scheculer 등을 구현

CS149

- 목표 : Parallel Computing의 이해
- 결과 : 과제를 통해 CUDA, C++ Multi-Threading, OpenMP 숙달

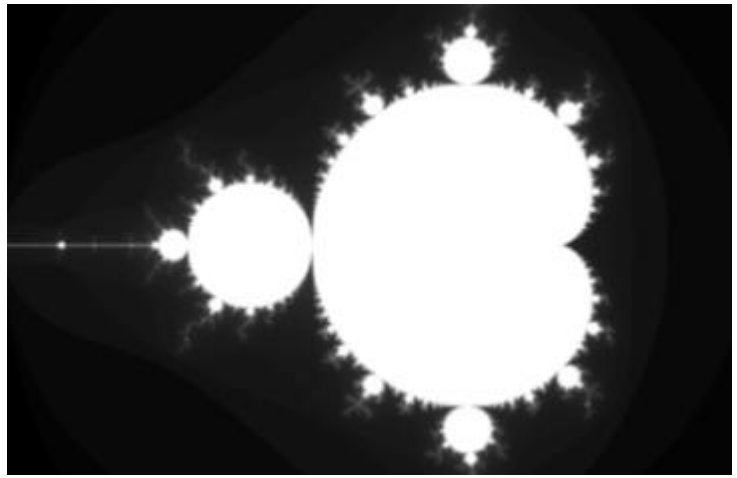
Paper review

- 목표 : Distributed Learning, LLM serving system 이해
- 결과 : 학습을 가속화 시키기 위한 방법들, 효과적인 LLM 메모리 할당 등에 대해 공부할 수 있었습니다.

1-4. Internship (2022 ~ 2024)

CS149

▣ Parallel Programming(1)



$$M = \{c : Z_{n+1} = Z_n^2 + C, \ \ lim_{n\to\infty}|Z_n| < \infty\}$$

자신을 제곱해서 더하는 형태로 이루어지는 연산을 Multi-thread로 구현하는 과제입니다.

▣ 내용

Multi-threading Part

- sleep lock, spin lock, async을 사용하여 Multi threading으로 Mandelbrot을 구하는 과제였습니다.

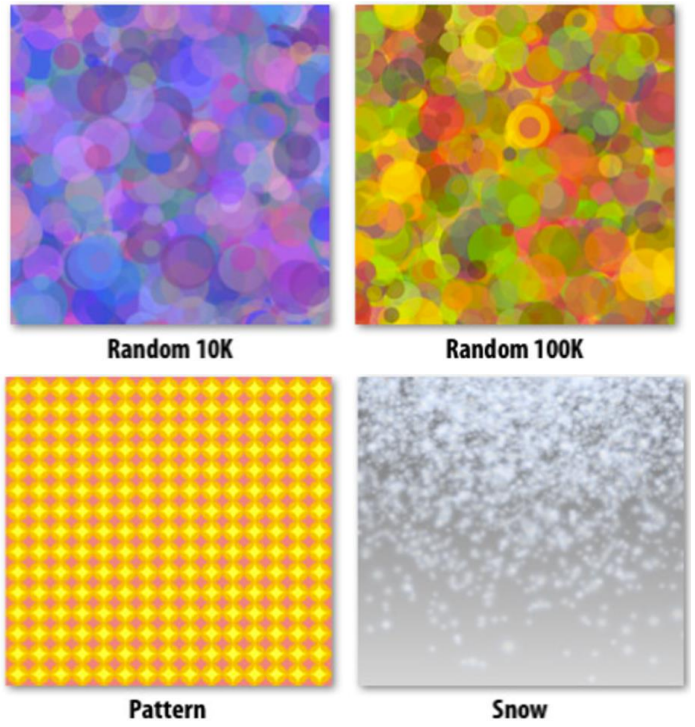
Result

```
Executing test: super_super_light...
Reference binary: ./runtasks_ref_linux
Results for: super_super_light
                                      STUDENT   REFERENCE   PERF?
[Serial]                              10.516    10.463      1.01  (OK)
[Parallel + Always Spawn]             109.062   108.403     1.01  (OK)
[Parallel + Thread Pool + Spin]       27.855    30.08       0.93  (OK)
[Parallel + Thread Pool + Sleep]      55.711    51.658      1.08  (OK)
================================================================================
Executing test: super_light...
Reference binary: ./runtasks_ref_linux
Results for: super_light
                                      STUDENT   REFERENCE   PERF?
[Serial]                              57.611    71.818      0.80  (OK)
[Parallel + Always Spawn]             112.716   119.109     0.95  (OK)
[Parallel + Thread Pool + Spin]       31.894    37.182      0.86  (OK)
[Parallel + Thread Pool + Sleep]      60.674    55.095      1.10  (OK)
================================================================================
Executing test: mandelbrot_chunked...
Reference binary: ./runtasks_ref_linux
Results for: mandelbrot_chunked
                                      STUDENT   REFERENCE   PERF?
[Serial]                              346.443   346.131     1.00  (OK)
[Parallel + Always Spawn]             47.064    47.345      0.99  (OK)
[Parallel + Thread Pool + Spin]       53.888    52.89       1.02  (OK)
[Parallel + Thread Pool + Sleep]      46.968    47.264      0.99  (OK)
================================================================================
Overall performance results
[Serial]                              : All passed Perf
[Parallel + Always Spawn]             : All passed Perf
[Parallel + Thread Pool + Spin]       : All passed Perf
[Parallel + Thread Pool + Sleep]      : All passed Perf
```

1-4. Internship (2022 ~ 2024)

CS149

▣ Parallel Programming(2)



CUDA를 사용하여 rendering을 진행하는 과제입니다.

▣ 내용

CUDA Part

- 이번 과제에서는 CUDA skeleton 코드내에서 틀린 곳을 고치고 rendering 기능을 구현하는 것이였습니다.
- Skeleton 코드에서는 이미지를 업데이트 하는 도중에 Atomicity를 지키지 않았기 때문에 해당 부분을 고쳤습니다.

Result

```
Scene : fireworks
Correctness passed!

Scene : snow
Correctness passed!

Scene : snowsingle
Correctness passed!
Your time : 19.2704
Ref Time: 9.4320

------------
Score table:
------------
-----------------------------------------------------------------
| Scene Name      | Ref Time (T_ref) | Your Time (T)   | Score    |
-----------------------------------------------------------------
| rgb             | 93.4623          | 0.6072          | 12       |
| rand10k         | 46.9117          | 4.2586          | 12       |
| rand100k        | 808.1904         | 39.5977         | 12       |
| pattern         | 2.6946           | 0.7216          | 12       |
| snowsingle      | 9.4320           | 19.2704         | 7        |
| biglittle       | 777.1138         | 27.8397         | 12       |
-----------------------------------------------------------------
|                                  | Total score:    | 67/72    |
-----------------------------------------------------------------
```

1-4. Internship (2022 ~ 2024)

CS149

▣ Parallel Programming(3)

Assignment 5: Big Graph Processing in OpenMP

Due: Fri Dec 8th, 11:59PM PT (No late submission allowed)

84 points total

If you complete this assignment, you will receive up to 10 bonus points on one of the regular programming assignments (PA1-PA4). Note that programming assignment averages are not capped, so this is essentially "extra credit" on the course.

OpenMP를 사용해서 3GB Big graph를 processing하는 과제입니다.

▣ 내용

OpenMP Part

- Top-down, Bottom-up, Hybrid를 사용하여 BFS 기능을 구현합니다.
- 기존 while loop에서 serial하게 움직이는 문제로 속도 저하가 발생했었는데 프로파일링을 통해 찾아 고쳤습니다.
- 아래 결과는 제 코드의 speed up과 Reference speed up이 비슷한 것을 보여줍니다.

Result

```
Your Code: Timing Summary
Threads  Top Down          Bottom Up          Hybrid
   1:    10.15 (1.00x)      9.61 (1.00x)      4.97 (1.00x)
   2:    6.73 (1.51x)       6.72 (1.43x)      3.18 (1.56x)
   4:    6.37 (1.59x)       5.81 (1.65x)      2.46 (2.02x)
---------------------------------------------------------
Reference: Timing Summary
Threads  Top Down          Bottom Up          Hybrid
   1:    9.72 (1.00x)       11.02 (1.00x)     5.92 (1.00x)
   2:    6.84 (1.42x)       8.10 (1.36x)      3.65 (1.62x)
   4:    5.70 (1.70x)       5.98 (1.84x)      3.06 (1.93x)
---------------------------------------------------------
Correctness:

Speedup vs. Reference:
Threads        Top Down          Bottom Up          Hybrid
   1:             0.96               1.15               1.19
   2:             1.02               1.20               1.15
   4:             0.90               1.03               1.24
```

1-4. Internship (2022 ~ 2024)

Paper review

▣ Paper review

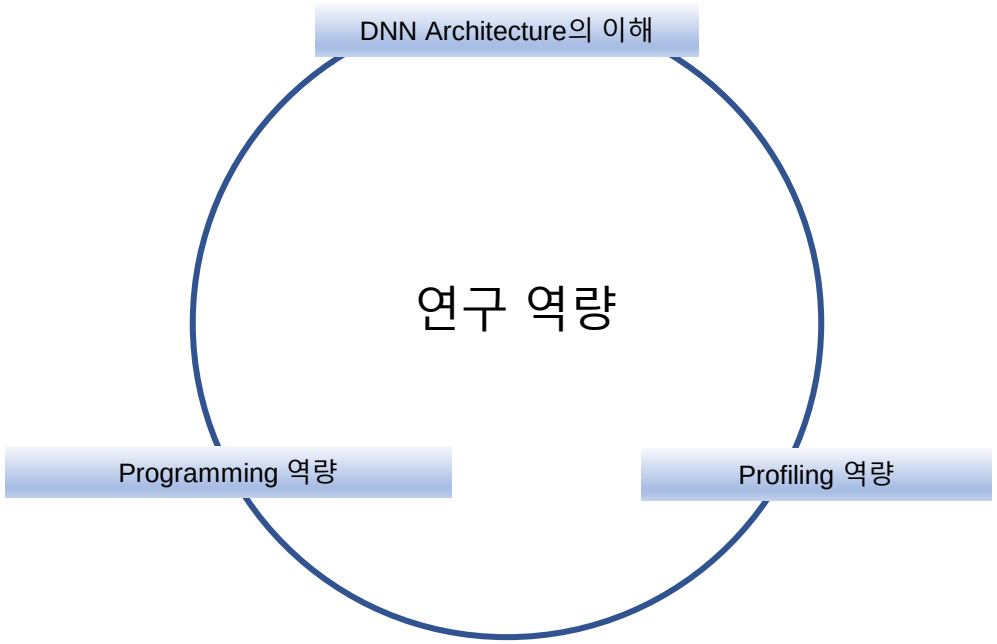
주로 Parallelism, Distributed Learning, LLM serving system에 대한 논문을 읽고 발표했습니다.

▣ 내용

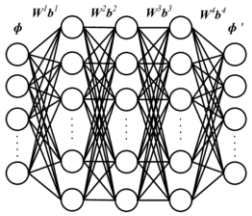
List

- 병렬 딥러닝 기법에 관환 연구동향분석
- PipeDream: Fast and Efficient Pipeline Parallel DNN Training
- GPipe: Efficient Training of Giant Neural Networks using Pipeline Parallelism
- Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism
- TurboTransformers: An Efficient GPU Serving System For Transformer Models
- ZeRO-Offload: Democratizing Billion-Scale Model Training
- DeepSpeed Inference: Enabling Efficient Inference of Transformer Models at Unprecedented Scale
- Orca: A Distributed Serving System for Transformer-Based Generative Models
- FlexGen: High-Throughput Generative Inference of Large Language Models with a Single GPU
- Efficient Memory Management for Large Language Model Serving with PagedAttention

2. 결론



◦ DNN Architecture의 이해

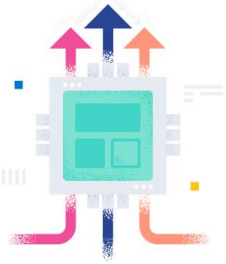


- Distributed Learning을 적용하기 위해서는 Model Architecture의 이해가 필요합니다. 저는 여러 프로젝트를 통해 다양한 Model Architecture 대해 공부했습니다.

◦ Programming 역량

- 딥러닝/머신러닝 프레임워크 활용 (Tensor flow, Pytorch)
- 병렬 프로그래밍(CUDA, OpenMP, C++ Multi-threading)

◦ Profiling 역량



- CS149 과제 중 시간 조건을 완수하지 못했을 경우 지연 부분을 찾아 해결했습니다.
- 프로젝트들을 진행할 때 사용하는 디바이스에 적합하게 모델을 선택하여 진행했습니다.

THANK YOU