Junha Hyung

1997.03.18

010-4308-2008

sharpeeee@kaist.ac.kr

junathesharpone@gmail.com

https://github.com/junhahyung

Senior at KAIST, Electrical Engineering & Computer Science. (graduation: 2021.08)

I have interest in machine learning and system programming, along with math and other domains of engineering.

Studies

GPA 3.92/4.3

KAIST courses https://github.com/junhahyung/grades

Non KAIST courses(online)

- Coursera machine learning - Andrew Ng (ML)
- Fast ai pytorch (DL)
- Stanford CS231n - Fei Fei Li (Computer Vision)
- Stanford CS224 - Chris Manning (NLP)
- Deepmind Reinforcement Learning - David Silver (RL)
- MIT Linear algebra - Gilbert Strang / currently studying

Awards

- KAIST EE 2019 fall Dean's list
- KAIST EE 우수학생 자격으로 CES2020 참가
- KAIST Leadership & Volunteer award ("리더십.봉사우수자" 수상)

Career Summary

2016.03 ~ enrolled in KAIST (graduation : 2021.02)

2018.02 ~ 2018.03 Individual research @ semiconductor system lab(KAIST, prof. Hoi-Jun Yoo)

2018.06 ~ 2018.08 Individual research @ ALIN-LAB(KAIST, prof. Jinwoo Shin)

2018.09 ~ 2019.02 Software/Machine Learning Engineer @ Kakao Corp, Reco-tech team(추천기술팀)

2019.06 ~ 2019.08 Research exchange student, studying NLP @ Okazaki lab, Tokyo Institute of Technology(prof. Okazaki)

2019.06 ~ 2020.06 Enrolling in SKT AI Fellowship program, working on Korean Sentiment Analysis project

2019.10 ~ 2019.01 Machine Learning developer for Hyundai RoadSense project (with Makinteract laboratory, KAIST)

2020.03 ~ 2020.06 Machine Learning researcher @ Visualcamp [https://visual.camp]

Skill Set

Programming Language

- Python, C/C++, Scala, Verilog, javascript, flutter

Frameworks

- Tensorflow, Pytorch, nodeJS, mongDB, AWS

Etc

- Fluent in Korean/English

Experience

2018.08 - 2019.02

Kakao Corp, Reco-tech team

Software/ML engineer(33 % contribution)

Improving Daum Cafe CTR (다음카페채널 CTR 고도화 프로젝트) - python/C++

- Implementing various MAB(multi-armed bandit) algorithms for Daum Cafe recommendation system
- Testing and implementing many different embedding & recommendation algorithms
- Implementing auto encoder for efficient recommendation
- Implementing real time-large scale CTR(click through rate) predictor system using FTRFL

2019.06 - 2020.06

SKT AI Fellowship - https://github.com/amy-hyunji/korean_multi_label_SA

Project participant(50 % contribution)

Developing Korean Sentiment Analysis system - python/tensorflow, Flask

- Data collection
- Implementing data augmentation & data cleansing algorithms
- Improving upon Bert(NLP model) implementing and using many different algorithms including adapter(by Google)
- Research & implementation on semi supervised learning & few shot learning
- Implementing web API using Flask

2019.10 - 2020.01

Roadsense Project

(100 % contribution on software part)

Developing real-time, vision-free road type classifier for electric scooter - python/pyQT, Keras

- Data collection
- Implementing GUI for data collection & inference
- Implementing deep learning classifier(convolutional neural nets)

2020.03 - 2020.06

Visualcamp Project

(with four people, equal contribution)

Research on normalization, refining gaze prediction model - python/Tensorflow

- Improving data normalization model
- Automizing data cleansing & normalization process
- 2D facial landmark generation using boundary heatmap - regression model
- Refining gaze prediction model using new embedding methods

Individual Projects

RISCV cpu - *https://github.com/amy-hyunji/Computer-Architecture*

Part of KAIST EE321(computer architecture) project, implening modern RISCV cpu. Implemented in verilog.

Text search engine - *https://github.com/junhahyung/text-search-engine*

Part of KAIST EE488 project, implening text search engine that uses inverted indexing and boolean query. Implemented in C.

Tcmalloc - *https://github.com/junhahyung/tcmalloc*

Part of KAIST EE488 project, implening tcmalloc, efficient memory management system for multi-threaded environment. Implemented in C.

High performance web server using I/O multiplexing and thread pool - *https://github.com/junhahyung/iomult*

Part of KAIST EE488 project, high performance web server that can deal with many concurrent client requests, up to 100,000. Uses I/O multiplexing and thread pool. Implemented in C

Simple static router - `https://github.com/junhahyung/simple-router`

Simple static router that runs with mininet.

Simple web proxy - `https://github.com/junhahyung/simple-proxy`

Simple web proxy implemented in C..

Etc. - `to be added`

Many small machine learning projects, algorithm projects, pintos projects

Awards

- KAIST EE 2019 fall Dean's list
- KAIST EE 우수학생 자격으로 CES2020 참가
- KAIST Leadership & Volunteer award ("리더십.봉사우수자" 수상)