HyeongChan Kim

https://github.com/kozistr, http://kozistr.tech/about

EDUCATION

Korea University of Technology and Education (KOREATECH) Mar 2016 –

CHALLENGES & AWARDS

6th place, **NAVER NLP Challenge**, SRL Task, 2018.
4th / 13th place, **NAVER A.I Hackathon**, 2018.
Final Round (Digital Forensic), **A.I R&D Challenge**, 2018.
2nd place (Demon), **Boot2Root** CTF, 2018.
2nd place (Demon), **WhiteHat League 1**, 2017.
3nd place (SeoulWesterns), **Harekaze CTF**, 2017.
9th place (3rd price, A book as award), **TF-KR MNIST Challenge**, 2017.

Kaggle Challenges :: Competition Expert

LB Top 2% Cornell Birdcall Identification (24 / 1395), 2020.
LB Top 9% ALAKSA2 Image Steganalysis (93 / 1095), 2020.
LB Top 4% Tweet Sentiment Extraction (84 / 2227), 2020.
LB Top 4% Flower Classification with TPUs (27 / 848), 2020.
LB Top 4% Bengali.AI Handwritten Grapheme Classification (67 / 2059), 2020.
LB Top 3%, Kannada MNIST Challenge (28 / 1214), 2019.

CTFs & Conferences

Staff, Challenge Maker, **HackingCamp 17**, 2018.
Staff, **Belluminar CTF**, 2017.
Challenge Maker, **KID CTF**, 2017.
Challenge Maker, **KOX CTF**, 2017.
Staff, Challenge Maker, **HackingCamp 16**, 2017.
Staff, Challenge Maker, **CodeGate OpenCTF**, 2017.
Staff, Challenge Maker, **HackingCamp 15**, 2017.
Conference Staff**, POC,** 2016.

PUBLICATIONS

[2] Kim et al, CNN ARCHITECTURE PREDICTING MOVIE RATING FROM AUDIENCE'S REVIEWS WRITTEN IN KOREAN. Jan. 2020.

[1] zer0day, Windows Anti-Debugging Techniques (CodeEngn Archive) Sep. 2016.

INDUSTRY EXPERIENCE

Watcha, Seoul, South Korea Jun 2020 – Present

Machine Learning Researcher

- Developed a pipeline to recognize all movie & tv actors from the poster & still-cut images.
  - Utilized SOTA face detector & recognizer.
  - Optimized pre/post processing routines to consider the latency.
- Developed a novel sequential recommendation architecture to recommend what content to watch next.
  - Achieved SOTA performance compared to pervious SOTA architectures (SASRec, BERT4Rec).
- Developed Image Super Resolution model to upscale movie & tv posters, still-cuts.
  - Optimized the codes for fast inference time & memory-efficiency on cpu.
  - In internal evaluation (qualitative evaluation by the designers), it catches details better & handles higher resolution & takes a little time.
- Working as a full-time.

Rainist, Seoul, South Korea Nov 2019 – Jun 2020

Machine Learning Engineer

- Developed the card & bank account transaction category classification models, designed light weight purpose for the low latency. (now on service)
  - In A/B (online) test result, improved **about 25 ~ 30%p *accuracy**, **about 0.6%p 1+retention**.
- Developed the machine learning model serving RESTful API server (utilizing k8s + open source project).
  - **zero failure rate** (zero 40x 50x error)
- Developed the classification model, forecasting possibility of loan overdue.
  - baseline deep learning model
- Worked as a full-time.

% *accuracy : how many people don't update/change their transactions' category.

VoyagerX, Seoul, South Korea Jan 2019 – Sep 2019

Machine Learning Engineer

- Developed speaker verification, diarization models & logic to recognize the arbitrary speakers recorded from the noisy (real-world) environment.

- Developed a hair image semantic segmentation / image in-paint / i2i domain transfer model to swap hair domain naturally.
- Worked as an intern.

ELCID, Pangyo, Korea Jun 2016 - Aug 2016
Penetration Test
- Penetrated some products related to network firewall and anti-virus.
- Worked as a part-time job.

OUT SOURCING

Korea University Course Information Web Parsing, ITL July 2017 – Mar 2018
AWS CloudTrail logger analyzer / formatter, ELCID Sep 2019 – Oct 2019

RESEARCH EXPERIENCE

Heterogeneous Parallel Computing Lab, Cheonan, Korea Sep 2018 - Dec 2018
Undergraduate Research
- Wrote a paper about improved TextCNN model for predicting movie rate.

TALKS

NAVER NLP Workshop 2018, Pangyo, Korea Dec 2018
- SRL Task, challenging without any domain knowledge

PROJECTS

Generative

Awesome Generative Adversarial Networks (Stars 550+) July 2017 –
Implemented lots of Generative Adversarial Networks in tensorflow 1.x. & 2.x Novelties of this project are trying to implement lots of GANs which some of them are not released or in tensorflow 1.x & 2.x based on the paper with some tweaks.

gan-metrics (Stars 3+) Mar 2020 –
Implemented lots of metrics for evaluating GAN in pytorch.

I2I Translation

Improved Content Disentanglement (Stars 3+) Sep 2019
Re-implement / tune 'Content Disentanglement' paper in pytorch.

Image Inpainting

Improved Edge-Connect (Stars 8+) Oct 2019
Re-implement / tune 'Edge-Connect' paper in pytorch.

Style Transfer

Neural Image Style Transfer Mar 2018
Implemented a neural image style transfer.

Segmentation	**Awesome Segmentation (Stars 65+)**	Aug 2018
	Implemented lots of image semantic segmentation and ordered the papers.	
Optimizer	**AdaBound Optimizer (Stars 40+)**	Jan 2019
	Implemented AdaBound Optimizer (Luo et al. 2019) w/ some tweaks in tensorflow.	
	RAdam Optimizer (Stars 4+)	Sep 2019
	Implemented RAdam Optimizer (Liu et al. 2019) w/ some tweaks in tensorflow.	
Super Resolution	**Deep Residual Channel Attention Network (Stars 38+)**	Sep 2018
	Implemented a RCAN model in tensorflow.	
	Enhanced Super Resolution GAN (Stars 25+)	Jun 2019
	Implemented an ESRGAN model in tensorflow.	
	Natural and Realistic SISR w/ Explicit NMD (Stars 5+)	Apr 2020
	Implemented a NatSR model in pytorch.	
NLP	**Improved TextCNN (Stars 4+)**	Dec 2018
	Implemented an improved TextCNN model (Kim et al. 2020)	
	Text Tagging	Dec 2018
	Implemented a text category classifier in tensorflow.	
R.L	**Rosseta Stone (Stars 490+)**	Sep 2018-
	Hearthstone simulator using C++ w/ some R.L. I contributed to this project by implementing `feature extractor` and `neural network' in libtorch++.	
Speech Synthesis	**Tacotron**	Jan 2019
	Implemented a google tacotron speech synthesis in tensorflow.	
Open Source Contributions	**syzkaller**	Apr 2018
	New Generation of Linux Kernel Fuzzer :: Minor contribution #575	

simpletransformers Apr 2020

Transformers made simple w/ training, evaluating and prediction possible w/ one line each :: Minor contribution #290