

MOHAMED NAGY

[in mohamed-naji](#) | [+20 102757 2897](#) | n4jidx@gmail.com | [mohamednaji7](#) | [K mohamedahmednaji](#)

Skills and Tools

- PyTorch | tensorflow2 | Pandas | NumPy | Prompt Engineering | OOP | NLP | CV | LLM | OpenCV | RL | stable-baselines3 | gym (Gymnasium) | CVXPY |
- Fair Skills: scikit-learn | Decision Tree | C++ | Arduino
- Linux, Windows | Jupyter Notebook, Amazon Sagemaker, Colab, Kaggle | Comet ML | VS code, Cursor

Education

B.S.: Communication and Electronics Engineering

Faculty of Engineering
Alexandria University

Alexandria, Egypt

Sep 2018 - June 2024

Learning: books, courses, projects,...

- **Graduation Project:** 5G PDSCH Simulator and Precoding with Reinforcement Learning [slides](#) (ML @ slide 99)
 - Overview: Analyzed the 5G PDSCH system, optimizing the precoding process using DDPG, an RL algorithm. Covered channel coding/decoding, four I/O types, and modulation techniques.
 - Approach: Solved the precoding issue for MISO and multi-user MISO scenarios under power constraints at varying SNR levels, given Rayleigh-distributed channel state information and the number of transmitting antennas.
 - Key Solution: To address phase ambiguity causing state-action confusion in complex vectors, we set the phase of the first antenna to zero and adjusted the other antennas' phases accordingly.
 - Results compared to MRT solution: SU-MISO 98.31% reward, MU-MISO 100.43% for low SNR, and MU-MISO 87.3% for high SNR.
- **Book:** Dive into Deep Learning [🔗](#)
 - **Project: RNN in NLP** [Github](#), [Kaggle NB](#)
 - Used Comet to do Bayesian optimization, manage code, log training results, visualize progress, and monitor hardware usage. Utilized 50% of 16GB GPU memory.
 - Optimized gradient clipping, sequence length, and hidden layer size. Observed overfitting with large hidden layers. Achieved PPL of 5.08 on SageMaker Studio Lab. [Comet](#)
 - Experimented with ReLU, Tanh, and varying RNN layers. Results showed overfitting, loss divergence, and early plateauing due to gradient vanishing. [Comet](#)
 - **Exercise: NLP** - preprocessing, and n-gram text analysis [Colab](#)
 - **Project: Modern Convolutional Neural Networks** [Kaggle NB](#)
 - Fashion-MNIST (70,000 Images) 10 categories: utilizing GPU P100, and 10 epochs for each architecture.
 - LeNet-5: Late loss dropping due to gradient vanishing, small architecture, 1 minute training.
 - Batch Normalization: Improved gradient flow and earlier loss drop.
 - AlexNet: High computational load, with a training time of 7 minutes, due to the FCN head.
 - VGG, NiN, and GoogleNet: It took 9, 8, and 6 minutes, respectively. Larger architectures are better for large datasets.
 - **Exercise: Neural Network to Fit Synthetic Linear Regression Data** [Colab](#)
 - **Project: RNN to Fit Synthetic (Sinusoidal) Sequence Data**
 - [Colab](#): given a sequence of length 4 predicting [linear regression], 1 step is accurate, but autoregressive predicting +8 steps is very noisy because the low error of one-step prediction accumulates.
 - [Colab](#): Retraining the RNN model by inserting the prediction in the input sequence improves the model's robustness to its previous errors and boosts performance for 6-step prediction.

- **Computer Vision Project:** Face Recognition Model Using Siamese Network 🔗
 - **Colab:** Collected personal data via OpenCV as a verification set, downloading the labeled faces in the Wild dataset 🔗.
 - **Kaggle:** incorporating a resizing layer to reduce the fully connected layer size, achieving the best performance on the facial recognition validation set. Leveraging image augmentation techniques will improve model robustness and accuracy.
- **Specialization: Machine Learning Specialization: Coursera, Mar2023**
- **Courses :**
 - **Introduction to Deep Learning** — STAT 157, UC | MIT 6.S191
 - **Structuring ML Projects | Neural Networks and Deep Learning | ChatGPT Prompt Engineering for Developers** — DeepLearning.AI
 - **Intermediate Machine Learning | Intro to Machine Learning | Pandas** — Kaggle
 - **Fundamentals of Reinforcement Learning** — Coursera
 - **Algorithmic Toolbox | Python Data Structures** — Coursera
- **Kaggle Competitions :**
 - **ARC Prize 2024:** Kaggle: [Explore Data](#), Colab: [CNNs1](#), [CNNs2](#)
 Trials: Tried to solve by making a foundation model, then using it with a few shots to learn to solve the competition problems. The foundation model I thought about is
 1. Treating the input as a matrix, and the output is a transformation version of the input.
 Built a matrix neural network layer, not FCN/CNN, which does rotations and translations to the input matrix.
 2. Tried to make CNN do a logical transformation of the input picture MNIST not to predict the class, but it was a bad idea
 - **House Prices Advanced Regression Techniques** [Kaggle](#), [Colab](#)
 - Using Random Forest to fit the data, achieving an RMSE of 0.147. after data preprocessing.
 - A multi-layer neural network results in a higher RMSE of 0.416 with 5-fold cross-validation. indicating that the neural network performed worse on tabular data.
 - **Digit Recognizer:** Using NN to recognize handwritten digits [Kaggle](#)
 - **Community Prediction Competition** [Github](#)
 - We [team: El-Noussour] applied Random Forest and got an RMSE of 0.733 and an XGBRegressor RMSE of 0.721.
 - then we tuned it and implemented 7-fold cross-validation for XGBRegressor, achieving an improved RMSE of 0.717
- **Soft Skill Courses:** ITI Recorded Course: AI/Nasr City/Job Seeking Skill | OS/Mansoura/Freelancing | AI/Business Sessions

Extra

- **Contribution:** Reported 2 XSS Bugs in doc.oracle.com 🔗 🔗
- **Projects:** - Built a RISC-V Processor in VHDL 🔗
 - We were presenting a research paper on optimizing wireless communication (convex optimization) [requirements](#), [paper](#), [slides](#), and some CVX tasks ([1](#), [2](#)).
- **Certs:** - HCIA-Datcom: Huawei, Jun 2020 - Jun
 - Google IT Support: Coursera, Oct 2020 - No Expiration
- **Competition:** One of the winners(~2%) in the [SCU Competition](#) Sep 2018 to get into the Faculty of Engineering
- **Small Projects:** Arduino and C++ [🔗, 🔗, 🔗]

Languages and Personal Info

- **Arabic:** Native Speaker
- **English:** Tutored Arabic to non-native speakers on Preply using English and talked to people on English-practicing websites
- **Location:** Beheira, Egypt—willing to relocate domestically or internationally
- **Military Status:** Exempted
- **Interests:** Boxing, Reading, Entrepreneurship, and Business
- **Birth Year:** 2000
- **Skills:** Figma 🔗 | Notion 🔗 | MS Office | Web Design 🔗