Ali Fartoot

Machine Learning Engineer

Phone:	+98 9129221194	**Email:**	fartoot.ali.80@gmail.com
LinkedIn:	ali-fartoot-552428170	**GitHub:**	github.com/Ali-Fartoot
Location:	Tehran, Iran		

Education

University of Tehran, Tehran, Iran
M.Sc. in Computer Engineering – AI and Robotics; *Sep 2023 – Present*
Advisor: Dr. Majid Nili Ahmadabadi
Research Topic: LLMs can be utilized to accelerate and enhance the training of RL agents.

University of Science and Culture, Tehran, Iran
B.Sc. in Computer Engineering GPA: 17.65/20.0 *Sep 2019 – Jul 2023*

Skills

AI:	Machine Learning, Deep Learning, RL, NLP, LLMs Computer Vision, Deep Generative Models, VLMs Meta Learning, Transfer Learning, Unsupervised & Self-supervised Learning Speech Processing, Time-series Data Analysis, Explainability & Interpretability
Data Science:	Exploratory Data Analysis & Visualization, Feature Engineering Data Pre-processing & Wrangling, Data Augmentation, Data Imputation Experimental Design, Model Hyper-parameter Tuning, Model Evaluation Model Deployment, Model Management, Monitoring & Versioning, SQL
Python Frameworks:	PyTorch, TensorFlow, Scikit-Learn, Pandas, NumPy Gymnasium, Stable-Baselines3, Open-CV, Pillow, Django, FastAPI Transformers & HuggingFace, Llama.cpp, SpaCy, NLTK, Matplotlib, Seaborn
Software Development:	Object-oriented Programming, Functional Programming API Development, Code Debugging, Code Modularity, Web Application Readability, Maintainability & Reuse, Data Engineering, Apache Spark& Hadoop
Tools:	Git, Docker, PostgreSQL, MongoDB Slack, JIRA, Trello, Windows, Linux
Soft Skills:	Problem-Solving, Critical Thinking, Team Collaboration Time Management, Adaptability, Technical Writing Presentation Skills, Public Speaking, Literature Review Technical Documentation, Teaching & Mentorship

Experience

1) AIEdut,Tehran , Iran
Machine Learning Engineer & Backend Developer *Apr 2024 – Present, Part-time*

- Utilized Django framework to create web applications for clients.
- Developed and maintained Python applications to automate data processing tasks.

- Developed a Retrieval Augmented Generation (RAG) system based on specified requirements.
- Identified and resolved issues and defects within the system.
- Conceptualized and implemented improvements to enhance system performance and accuracy.
- Implemented machine learning algorithms using Python libraries such as NumPy and Pandas.

2) Cognitive Systems Lab. at University of Tehran ,Tehran ,Iran
Research Assistant & Data Scientist *Dec 2023 – Present, Part-time*

Conducted comprehensive data analysis on student performance from the previous semester. This analysis focused on evaluating the impact of AI tools on learning outcomes. By employing metacognitive analysis techniques, assessed whether AI interventions contributed to improved learning experiences and enhanced overall performance in courses.

The data was collected from students in three stages:
1. A prototyped version of ChatGPT was used to check the quality of student prompts.
2. Discord chat logs were analyzed to measure the engagement of students with each other and with the course.
3. Google Colab logs were examined to assess the quality of coding, code refactoring, etc.

The technologies I used:
- OpenAI API as a classifier (GPT-4o mini, GPT-4o)
- Multilingual embeddings for comparing sentences and measuring similarities between prompts
- Plotted the features and found some conclusions and correlations between them

3) epishkhan.org, Tehran , Iran
Machine Learning Engineer *Apr 2023 – Jul 2023, Full-time*

I had a responsibility of developing a neural network for a vision task with Pytorch for facial recognition, and the model was trained using an external dataset, achieving an accuracy rate of 95%. Moreover, the model was designed to be modular, facilitating its seamless deployment on various systems. The technique was used, called contrastive learning for face recognition.
Key responsibilities:
- Design a model which can work in any type and scale of pictures.
- Can be deployed in any system like an API server, docker, etc.
- Low latency respond and compact storage

4) Teacher Assistant

NLP *Sep 2024 – Present, Part-time*

Advance Programming *Nov 2019 – Apr 2020, Part-time*

Database *May 2020 – Jul 2020, Part-time*

Operating system *Jan 2022 – Mar 2022, Part-time*

Projects

NLP

1) Persian NER XLM-Roberta: A token classification to predict each token to (O : None, Loc :location, Per : person, Org : organization) label. The model accepts Persian and English and performed 95% f1-score.

2) PEYMA-ARMAN-Mixed: The dataset which is a combination of PEYMA and ARMAN Persian NER datasets. It contains named entity tags which you can see from given link

3) Text Summarization: A summarization model (mT5) is fine-tuned on the XLsum dataset for Persian and English.Result: ROUGE-1 -¿ 29.83 ROUGE-2 -¿ 17.4 ROUGE-L -¿ 21.95

4) A movie Recommendation system: This system allows users to find 3 types of familiar movies related to its movies in the playlist intergrate with a movie player in Python. Used IMDB dataset, TF-IDF, and cosine similarity.

5) Text Classification: A ParsBert Fine-tuned on Persian text performed 85%.

Computer Vision

1) Soccer-Vision: Overview This repository contains a Python program utilizing YOLOv8 from Ultralytics to perform team prediction and player detection in a custom dataset. The program achieves an mAP of 38 at 50 IoU (Intersection over Union) and maintains an average FPS (Frames Per Second) range of 18-20.

2) Face Recognition: face recognition app using PyTorch and the Siamese neural network architecture, achieving an impressive accuracy rate of 95%. This app is specifically designed for single-class identification, offering enhanced security and personalized user experiences.

3) Persian-Vehicle-License-Plate-Recognition: A Flask web application for Iranian Vehicle License Plate Recognition using Inception-ResNet-v2 and Persian OCR.

4) Number$_{Guesser}$: An application for recognizing numbers with almost 95% accuracy, written using the Scikit-learn library as the core of its processes, Pygame for drawing the application, and the Pillow library for editing images.

5) DCGAN: An implementation of DCGAN with TensorFlow for the MNIST dataset and generating numbers between 0-9.

6) SAM model fine-tuned: This is an image segmentation task utilizing the SAM model fine-tuned on satellite images oflakes. The model's performance was as follows:Validation IoU: 0.2809190211001732 Validation Dice Coefficient: 0.5618380422004365

Others

1) Kaggle Notebooks: All Kaggle competition notebooks that I have completed throughout my career as a data scientist, including Disaster-or-not tweets (an NLP task), House Prices (a regression task),Banco Santander (a regression task), and some more.

2) pyspark-recommendation-systems: This PySpark script implements collaborative filtering for video game recommendations, including data preprocessing, model training using ALS with parameter tuning via cross-validation, evaluation with RMSE, and computation of Precision and NDCG scores.

3) RateMyPic: A django web application for collecting picture from other websites and store in database & store like and dislike for each image. The application are served on docker compose & postgres.

4) RL-playground: Some Reinforcement Learning project which I've done through learning this field.