



Interview Task: Face Recognition System

February 20, 2024

1 Introduction

This document outlines a practical task designed for AI developer candidates to assess their skills in AI development, microservices architecture, and system deployment. The task involves creating a face recognition system using a pre-trained model, designed around a microservices architecture for enhanced scalability and maintainability. The software side of this task is not too important but it has own bonus. You received an attachment folder that it contains sample test video and the person images file that you should process them in this task.

2 Task Overview

The candidate is required to develop a system that:

- Detects faces in test video.
- Recognize a specif face that stored in database.
- Save the time of video that the face seen in that frame.
- Log the all of action time like face detection and similarity check time.

3 System Architecture

The system should be divided into the following microservices:

1. **Video Input Service:** Read video file and send making pre-process on raw frame and then send processed frames to second service.
2. **Face Detection Service:** Receive pre-processed frames numpy data and detect faces of that frame and then Send faces to third service.
3. **Data Processing Service:** Receive detected faces and make similarity check to existing face embedding data. If specific person find in that frame send face data and the time of frame to next service.
4. **Data Forwarding Service:** Save a log when a data is received from the previous service.

4 Technologies and Communication Methods

Candidates may use the following technologies and communication methods between services based on their experience:

4.1 Technologies

- Gin(based on Go)
- FastAPI,Django,Flask(based on python)
- Vector databases (e.g., faiss)

4.2 Communication Methods

- RESTful HTTP APIs
- gRPC
- Messaging Queues (e.g., RabbitMQ, Apache Kafka)
- Event Streams
- WebSockets

5 Evaluation Criteria

Candidates will be evaluated based on:

- Functionality and accuracy of the face recognition system specially in similarity stage
- Effective deploy strategy (e.g., docker)
- Quality of the code
- System Operation Manuals (README.md)
- Using Cutting-Edge Technologies and Tools
- Commit good message in git

All of these items are not forced to implement but it has own bonus.

6 Submission Guidelines

Please submit all deliverables via git in your github account and share it with us. If you have any question, contact with this email miladisjobs@gmail.com