

Report

1. Function/Features

Receive a photo or URL of a photo containing human faces from users. After the user chooses to predict emotion/gender, the system replies the user with the input image plus predictions of emotion/genders of faces.

2. Machine Learning Task

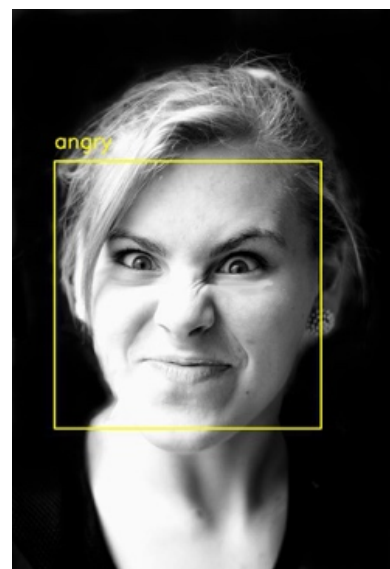
Pre-trained models from https://github.com/oarriaga/face_classification are used in this project.

Face detection and emotion/gender classification use fer2013/IMDB datasets with a keras CNN model and openCV.

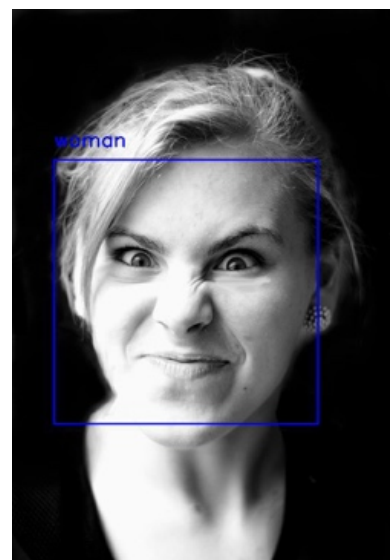
- IMDB gender classification test accuracy: 96%.
- fer2013 emotion classification test accuracy: 66%.



Input

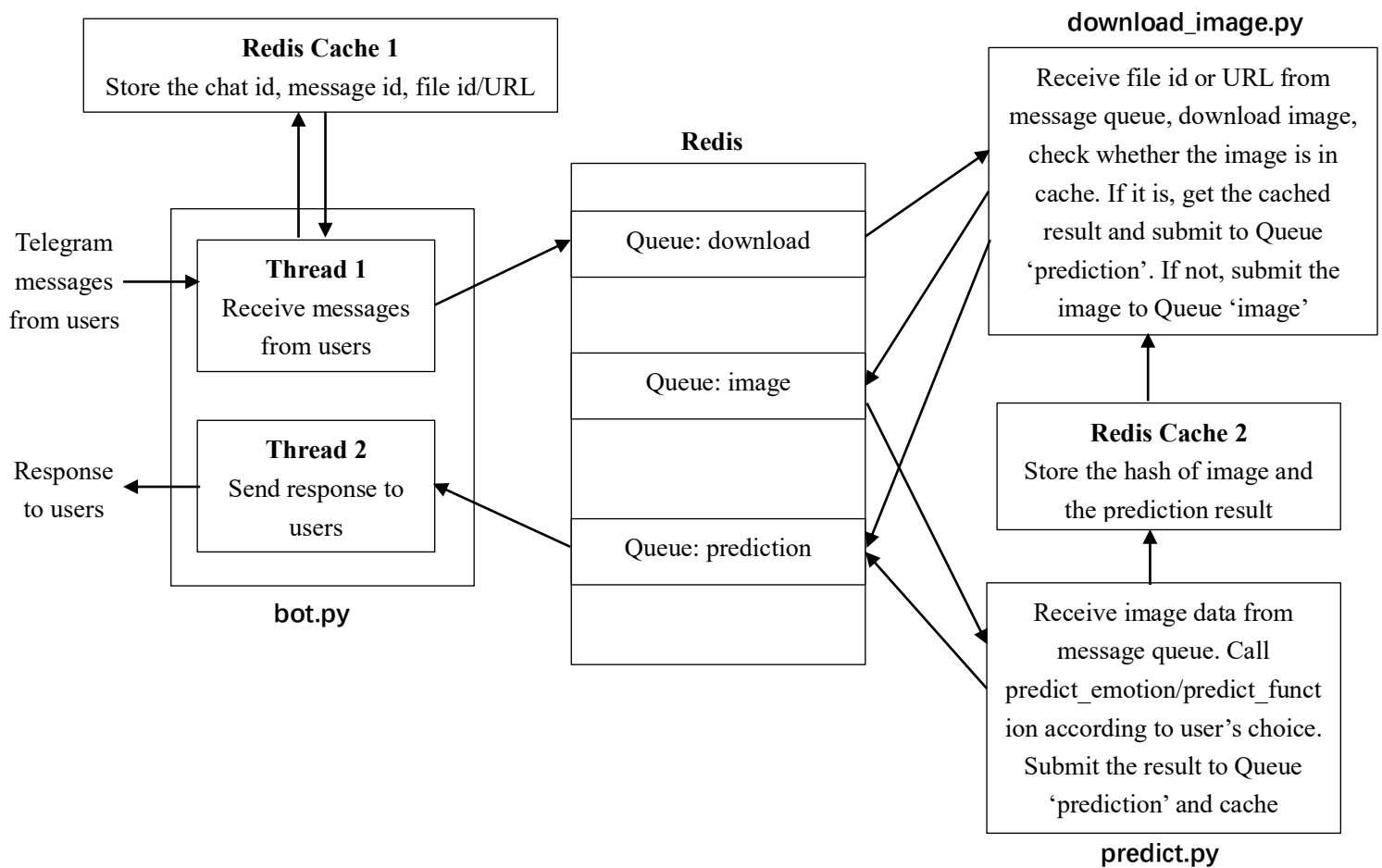


Output: emotion



Output: gender

3. Diagram of the System Architecture



4. Designed to Be Scalable

- 1) Both `download_image.py` and `predict.py` can work in multiprocessing. They can consume messages from the message queues alternatively. The number of processes can be changed according to the number of requests and the number of server machine processors.
- 2) We can also add computing nodes to run `download_image.py` and `predict.py` so as to handle increasing workload.
- 3) Storing information in Redis Cache 1 make user be able to choose different prediction and repeat doing prediction of the same photo in a while.
- 4) Storing information in Redis Cache 2 can decrease the waiting time for users if the image has been predicted before.