Age and Gender Classification

(Specifications)

Bui Gia Huy*†

March 5, 2019

*Artificial Intelligence Department
†Email: huygb@vietnews24.com

DOCUMENT CHANGE LOG

Rev.	Description	Author	Date	Approved	Date
0.1	Draft 1	Huy G.B	2019/03/04		

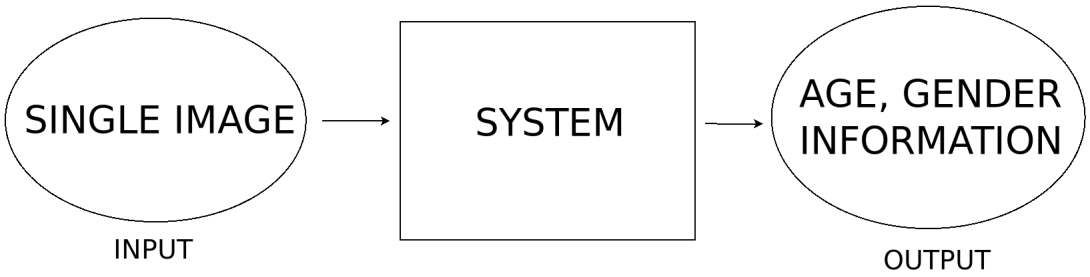
This document has been based on template TL-SSW-101 Revision 2.0

Contents

1 Introduction

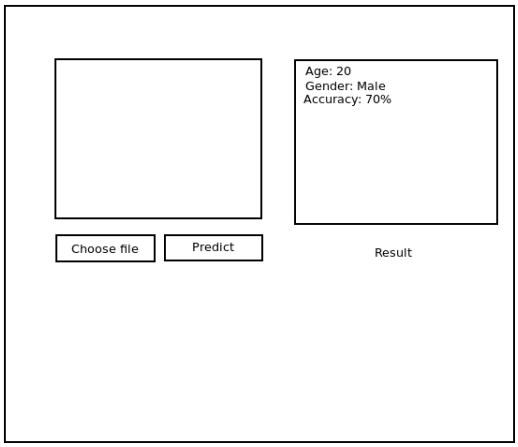
- This *Casptone Project 1* assits for my learning about Machine learning and Deep learning in training course of D-Soft company that Mr. Trung Anh is my supervisor.
- **Classification** is a interesting problem in Pattern Recognition field. Though, it has a lot of interesting applications in real life to serve and improve life quality. So, through this training course, i want to build a small classifier to classfy age and gender of a people with some knowledges that i studied to apply to real problems.
- The project is a system that basically can classify **Age** and **Gender** of a people if they feed image into my system through a website.

** Image condition: A Singe face, a single person*



Overview

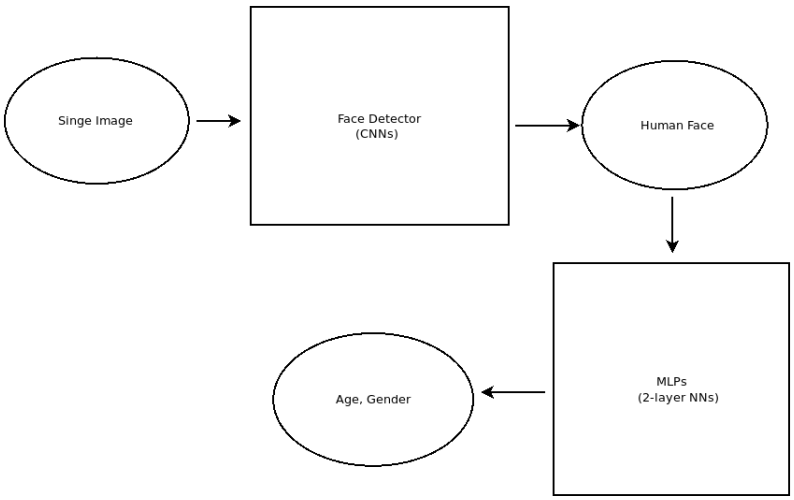
- Input: Single Image.
- Output: Age, Gender information.
- System:
  + Frontend: HTML, CSS



Frontend

  + Backend: NodeJS + ExpressJS
  + Classification System:

* Face Detector: Convolutional Neural Networks *(dlib.net)*
* Age, Gender Classification: 2-Layer Neural Network



Classification Overview

1.1 Purpose

- Reinforce knownledge about machine learning.
- Build a funny application that apply machine learning to the real world.

1.2 Scope

- The project focus mainly about:
+ Build a simple web frontend and backend.
+ Training a machine learning algorithm

1.3 Technical Overview

- Basically, system that i build will use Neural Network *(Multilayer Perceptron)* as a classfier algorithm.
- The system will use *Face Detector* of Dlib library for detecting human face. Then, it is feed into Neural Network to classify Age, Gender.

1.4 Definitions, Acronyms, and Abbreviations

AI	Artificial Intelligence
FNN	Feedforward Neural Network

References

1. https://talhassner.github.io/home/publication/2015_CVPR
2. https://talhassner.github.io/home/projects/cnn_agegender/CVPR2015_CNN_AgeGenderEstimat
pdf
3. https://gilscvblog.com/2015/11/19/age-and-gender-classification-using-deep-convolutio
4. http://dlib.net/
5. https://js.tensorflow.org/

2 Overall System

- In this section, I will introduce briefly about my overall system for Capstone Project 1.
- Project name: **Age and Gender Classficication.**
  + Input: Image
  + Output: Age, Gender information
  + Platform: Web App

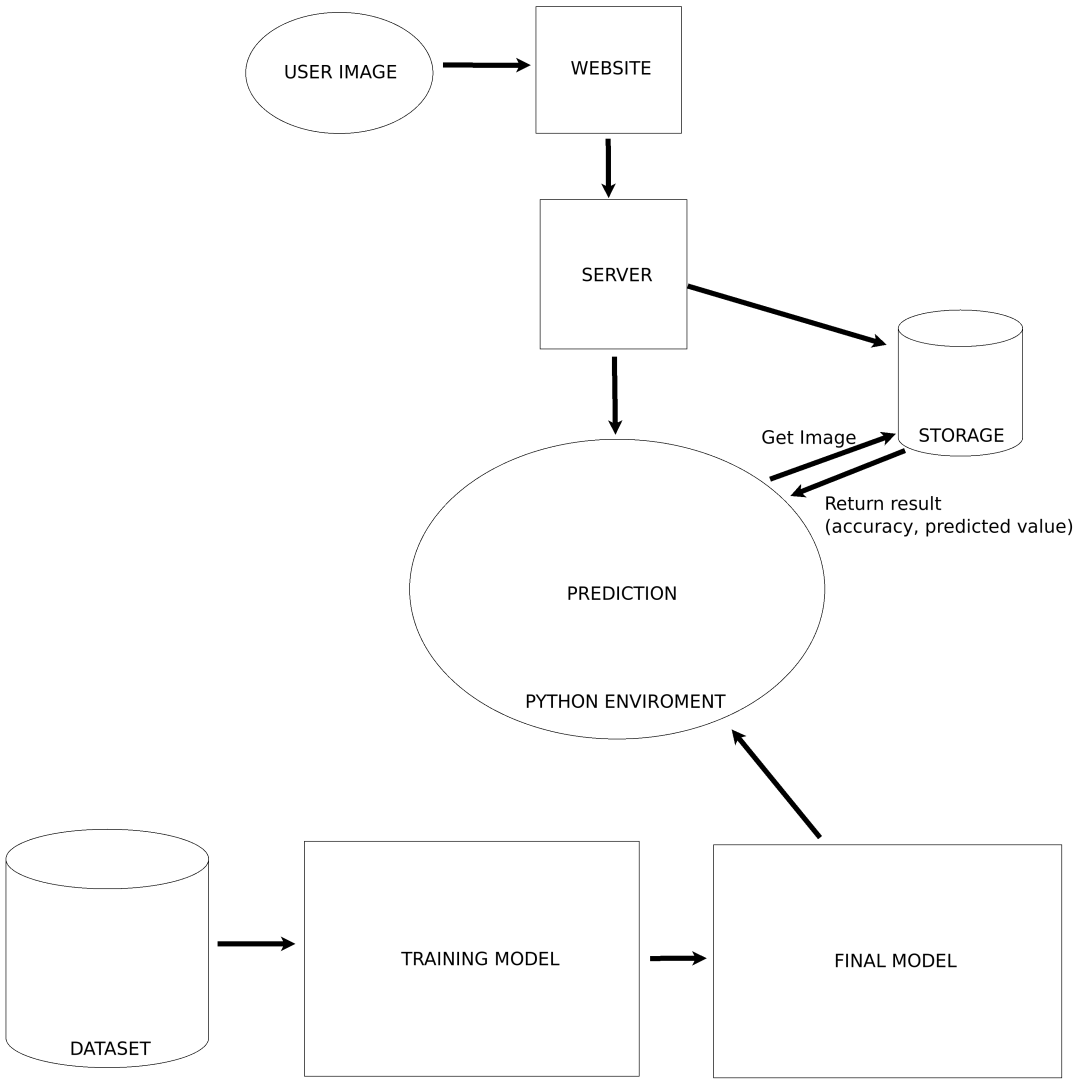


Figure 1: Overall System (A High level Diagram)

2.1 Age and Gender Classification

2.1.1 High Level Diagram

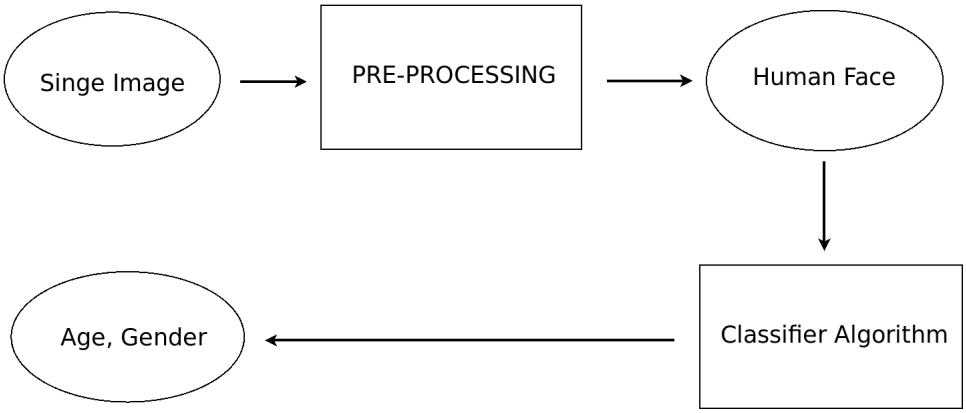


Figure 2: Classfication Pipeline

2.1.2 Dataset

- Preparing training dataset for classifier algorithm - Datasets that i am going to plan to use for my classifier algorithm as follows:
+ IMDB-WIKI (`https://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki/`)
+ VISAGE (`https://www.forensicsandsecurity.com/visage.php`)
+ UTKFace (`https://susanqq.github.io/UTKFace/`)
+ Blog Posts Labeled with Age and Gender (`https://www.kaggle.com/tomlisankie/blog-posts-label`

2.1.3 Pre-processing

- Grey + crop images in dataset.

2.1.4 Neural Network as Classifier Algorithm

- There are a lot of classifier algorithms that we can use. But here, we will use **Neural Network (NN)** for our problem.
- Feedforward Neural Network *(FNN)* is NN architecture that we apply for age and gender classification. Below is basic architecture of it **(figure 3)**.

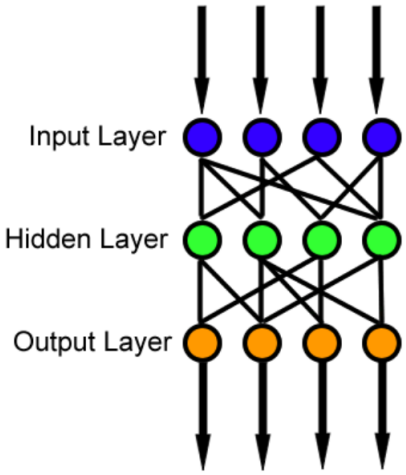


Figure 3: Feedforward Neural Network

- In this Capstone Project 1, we just only use FNN. The archictecture is not usually fit for **computer vision** because it has to have a very high parameters *(weights)* capicity in their model. These problems in computer vision field is often solved efficiently with Convolutional Neural Network *(CNNs)* because it can decrease dramatically large amount of weights in their model but accuracy of model is very high meanwhile still remaining *spatial arrangement* in addtional to *local connectivity*.
- Return our problem, our FNN architecture is namely as **figure 4**. That is a 2-layer neural network, the network's input is image. In training phase, we will use images from *training set* and in test phase, we use ones from *test set*. Maybe in hyperparameter-tuning phase, we can use *validation set* for regularization.

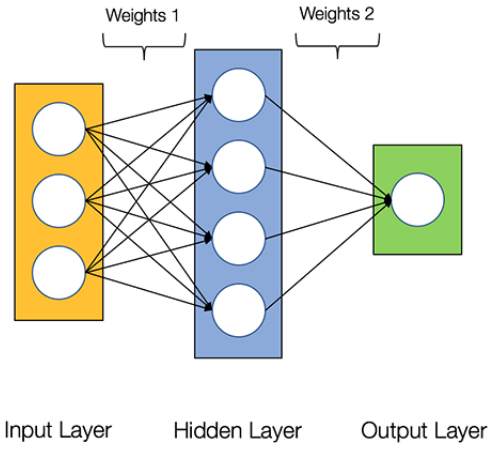


Figure 4: 2-layer Neural Network

2.2 Web Application

2.2.1 Client

- Clients are some members in our teams :)
- User can access to our website for testing purpose.

2.2.2 Server

- *Server language*: **Nodejs**
- *Server Framework*: **ExpressJS**

- RESTful APIs with Nodejs and Expressjs Diagram *(non-blocking IO)*:

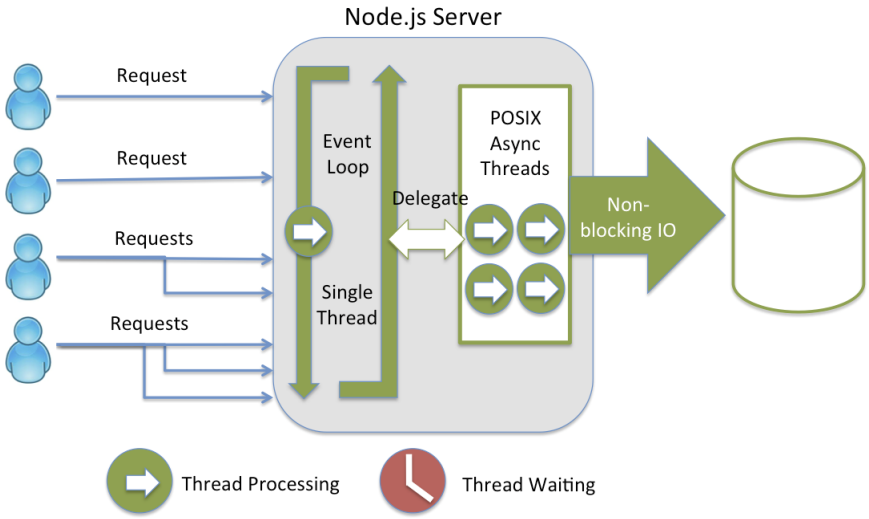


Figure 5: Internal Process of Nodejs

- My server will reiceives requests from users and holds their images taken from user my local storage. When user submits an image in a given form, it will be sent to my local server through RESTful API. Then, Nodejs server will send it to my Python script for predict Age and Gender with above 2-layer Neural Network depicted such as above.

- This figure below is high-level diagram for my backend architecture:

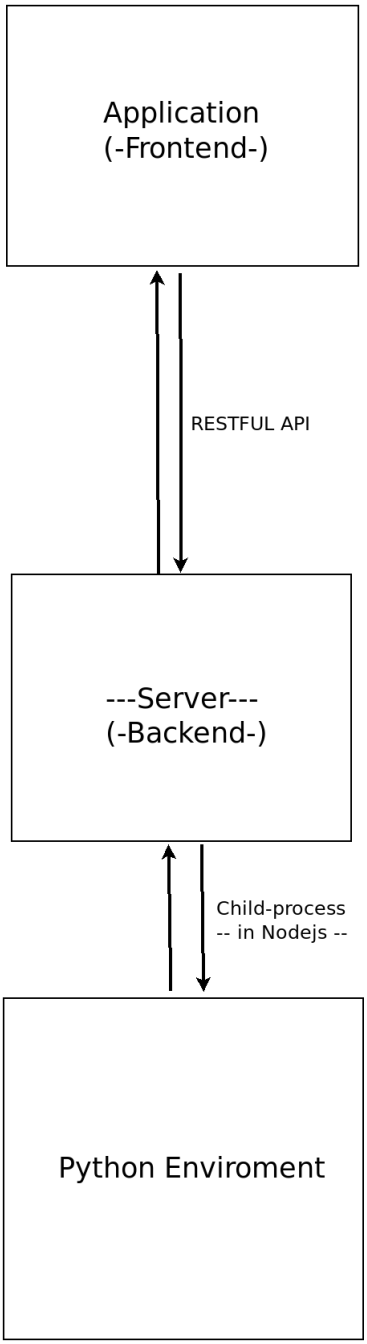


Figure 6: Backend Architecture

3 My Plan

....

Work Breakdown Structure & Estimate				
#	Activities			Expected (man-days)
1	**Preparation**		**1**	
	1.1	Setup the development environment		0.5
	1.2	Search + Pre-processing Dataset		0.5
2	**Design and Implement**		**3**	
	2.1	Training + Testing model		2
	2.2	Frontend + Backend		1
		Total (man-days)		**4**