

CICS/CMPE 351-Assignment2

February 2022

1 Introduction

E-commerce has changed the business world and the way we purchase items. One common problem faced by the ecommerce owner/platform is to tag appropriate labels with images of products so that those products can pop out when a relevant query is entered. Unfortunately, human annotation is expensive and error-prone. Thus, an automated solution that can identify the tags of images is needed.

The goal of this assignment is to create a classifier that can identify the label of a fashion product image using convolutional neural networks.

Reference/Tutorial (you can reuse code from online resources, but you can not copy answers from others, this is an independent homework):

- <https://towardsdatascience.com/image-classification-python-keras-tutorial-kaggle-challenge-45a6332a58b8>
- <https://machinelearningmastery.com/how-to-use-transfer-learning-when-developing-convolutional-neural-network-models/>
- [https://www.overleaf.com/learn/latex/LaTeX_video_tutorial_for_beginners_\(video_1\)](https://www.overleaf.com/learn/latex/LaTeX_video_tutorial_for_beginners_(video_1))
- <https://towardsdatascience.com/interpretable-machine-learning-for-image-classification-with-lime-ea947e82ca13>

Submission and marking: you need to conduct experiments and answer the following questions using your experiment results. You need to submit your runnable code so that the TAs can replicate your experiments. You also need to present your answers in a pdf file. Your submission will be evaluated based on the performance of your model on the provided testing dataset, and the quality of your presentation in the pdf file.

2 Part 1: Image Classification using CNN (50 points)

In the first part, your task is to create a basic CNN model that can identify if an image is related to the following 13 subcategories: Topwear, Bottomwear,

Innerwear, Bags, Watches, Jewellery, Eyewear, Wallets, Shoes, Sandal, Makeup, Fragrance, Others. More specially, you are given a dataset containing 44,441 fashion product images. You are also given two files, i.e., train.csv and test.csv containing the meta-data (image id, label, productDisplayName) related to each image. We do not have specific requirements on the structure of the CNN model. You can decide it by yourself.

Your answer to this question in the pdf file must contain:

- A report of the performance of your CNN model on both train and test datasets in terms of accuracy.
- Describe the parameters of your model and the structure of your model (you can use `model.summary()` if you are using keras).
- Write steps for us to replicate your experiments.

3 Part 2: Improved Image Classification (50 points)

Improve your base CNN model proposed in part 1. You can consider the following ways to improve your baseline:

- Tuning parameters (e.g., number of filters, number of units for the dense layer, add layers, activation function) on a dev set (part of your training data)
- Apply transfer learning using a popular pre-trained image classification model, e.g., VGG-19, Inception V3, and ResNet50 (<https://keras.io/api/applications/>)
- Combine image and text (product description) for image classification
- Generating more image data for training (e.g., <https://keras.io/api/preprocessing/image/#imagedatagenerator-class>) .

Your answer to this question in the pdf file must contain:

- A description of how you modify the base CNN model in part 1.
- Report the performance of your new model on the testing dataset.
- Write steps for us to replicate your experiments.