

Project Title: Deep Learning Visual E-Commerce Recommender

Project Team: Team Royals

Team Members: Wayne Nguyen, Hongcheng Jiang, Fei Wu, and Zhaobin Zhang

GitHub: <https://github.com/waynenguyen303/deep-learning-visual-eCommerce>

1. Project Goal and Objectives - Increment One

a) Significance

To improve and refine our initial project objectives, making a full working application was not advised at this time. Instead, an emphasis on the fashion data set should be a priority. In the world of big data, it is only as good as its data sets (garbage in, garbage out). For this increment, we want to focus on collecting staged images of the fashion item and categorize it accordingly. This is to help establish a baseline for accuracy of the implemented machine learning / deep learning algorithms. By using just staged images, we can also get a ballpark range of our implemented algorithms time efficiency. Since the end goal is to give recommendations to the user, the application should not exceed a certain time threshold. If it takes too long, this can make the user not use the application. Further on, we would want to incorporate 'live' images of the fashion item.

b) Use Case/Scenario

A broad use case is to have the user import any image and have our application recommend the fashion item of interest. This of course is a huge problem to tackle. A better starting scenario is to test staged or cataloged images. The fashion item image should be centered, in high definition, and formatted to a certain size. There should be no to little background noise in the image. With this, we would run a three-stage classifier module on the item. The first stage is a deep learning algorithm to classify what the item is. Second, a machine learning algorithm (Clarifai) to classify its design elements and features. The third stage would be a logo recognizer algorithm to see if there is any logos attached to the fashion item. Once the module is done, the image is now correctly tagged with what the item is (watch), what details it has (silver with diamonds), and what brand (Rolex). This tag can then be run through a knowledge recommendation algorithm to show the user where to buy that or a similar item.

2. Approach

a) Data Sources:

Fashion images and categories for our first data set was selected from a google.com search. The first data set is stored here @ <https://github.com/waynenguyen303/deep-learning-visual-eCommerce/tree/master/fashion-item-dataset/data4>

Further collaboration with a fashion company might be useful for getting a much larger data set.

b) Analytic Tools and Task:

For accuracy testing, a confusion matrix will be implemented. There are many libraries like pandas_ml that can be imported and used to show the applications run analytics. A confusion application was also given in class to analyze Naïve Bayes, random forest, and decision tree algorithms. A timer log will be used to show how long each algorithm runs for with a given data set size.

c) Expected Inputs and Outputs:

For this increment, expected inputs should only be staged fashion images for best accuracy and determination of which classification algorithms to use. Image annotations will be by the Clarifai machine learning tool. This will be used to get the details of the fashion item. There are Naïve Bayes, random forest, and decision tree classifier applications available to categorize what the item is, but so far, the accuracy does not look very promising and we would like to see if there are any deep learning modules like Alex-Net that can do a better job at categorizing the fashion item. An object detection application was given in class. This will be used to learn and detect logos. The output would be a vector array of these three stages to give to a knowledge algorithm (Google) for recommendations of the item.

d) Algorithms:

Naïve Bayes, random forest, decision trees, SVM, Alex-Net, linear regression, KNN, neural networks, K-Means, etc.

Libraries: spark, mllib, pandas_ml, openimaj, Clarifai, Google, etc.

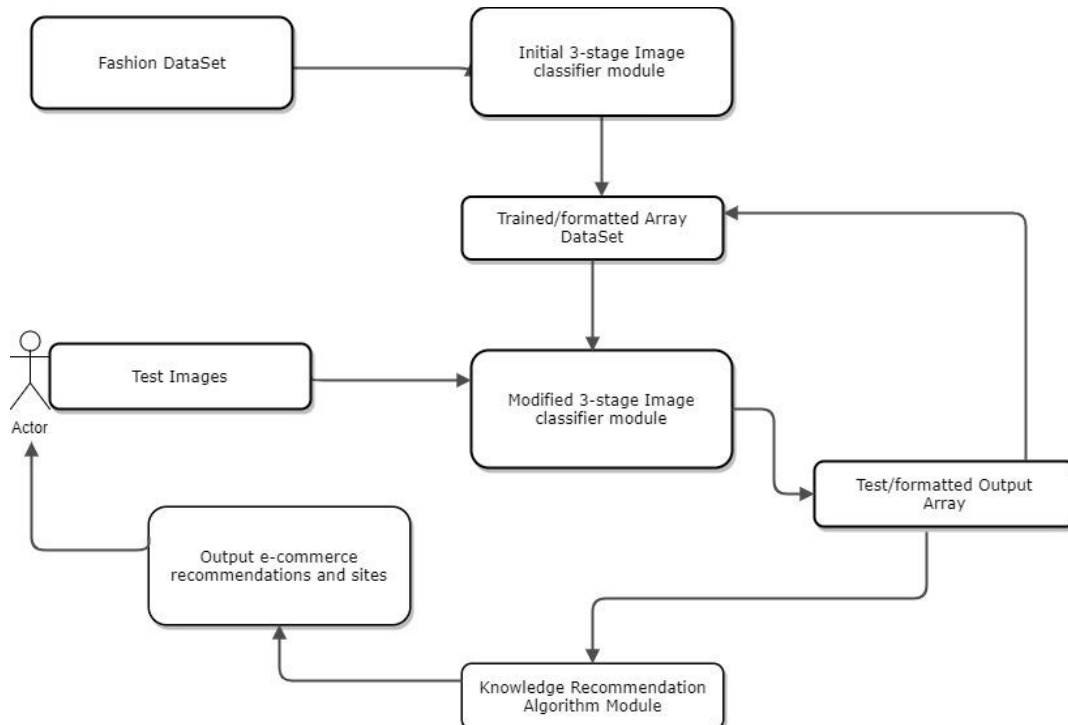
Source code: <https://github.com/waynenguyen303/deep-learning-visual-eCommerce/tree/master/algorithm-sources>

3. Related work

Deep learning [3] has become a promising technique to perform various tasks, especially in image/video-based tasks. For examples, face recognition, classification and retrieval to name a few. In [1] a novel E-Commerce system has been proposed which is based on VGG-16 [2].

4. Application Specification and Implementation

a) Workflow



b) System Specification

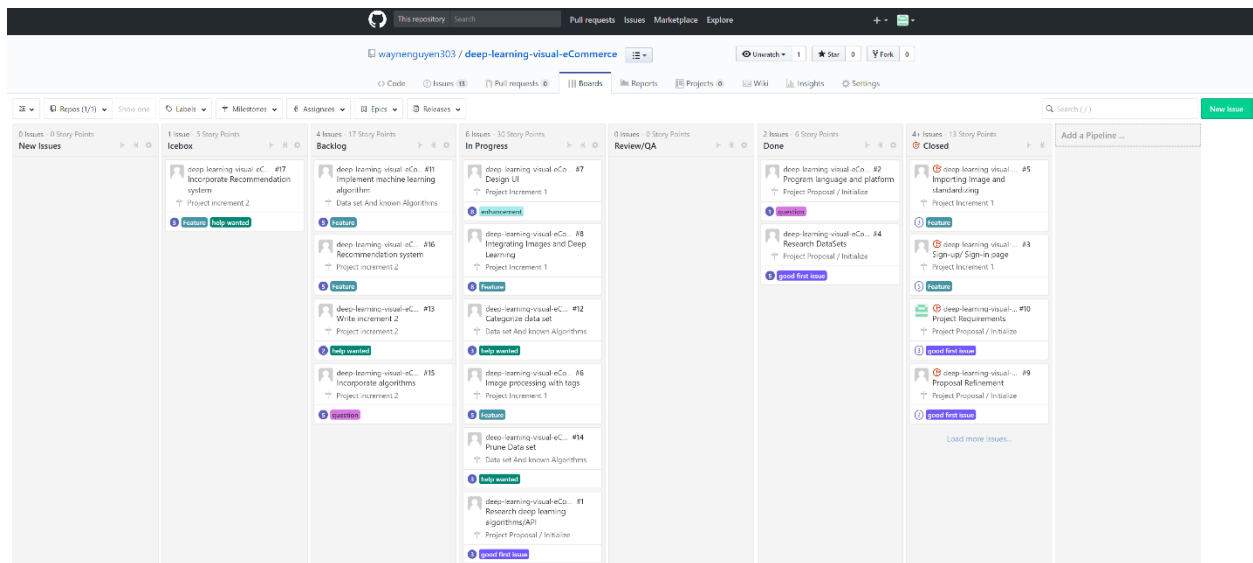
IDE: IntelliJ, SBT, Pycharm, Android Studio

Services: Image annotation - Clarifai API <https://www.clarifai.com/>, openimaj, Adobe spark <https://spark.apache.org/>, object detection - CS5542 lab tutorials.

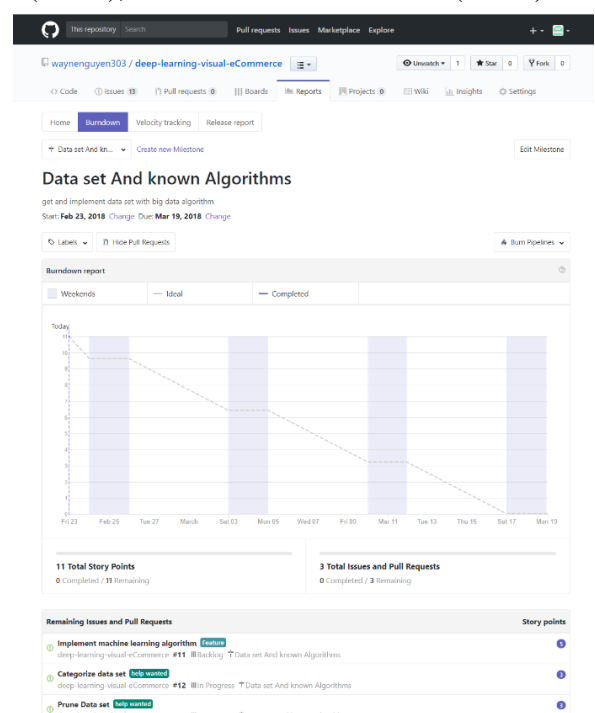
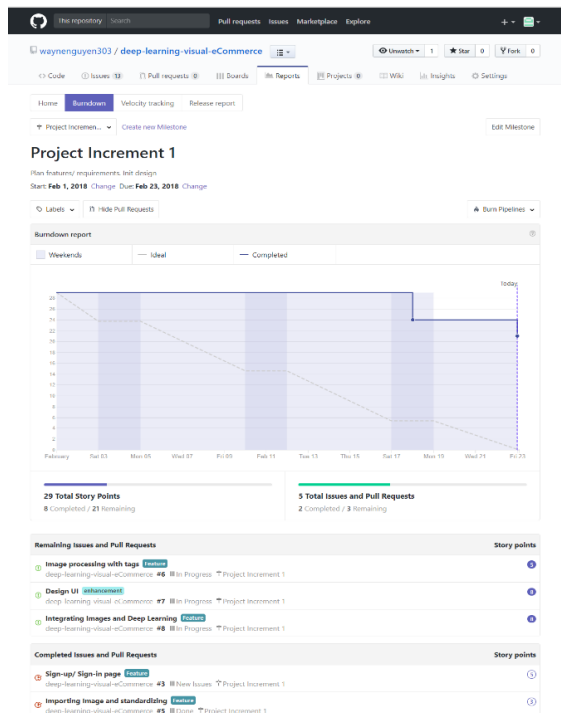
Sources @ <https://github.com/waynenguyen303/deep-learning-visual-eCommerce/tree/master/algorithm-sources>

5. Project Management

a) ZenHub Burn Down Charts



- To do and closed Issues on the Zenhub Board(above), Milestone burndown charts(below)



b) Work completed/to be completed

Completed:

Initial data set (hours = 3): Wayne Nguyen

Algorithms research (hours = 5): all members

Algorithm usage/code (hours = 5): Zhaobin Zhang, Wayne Nguyen

IDE application implementation (hours = 10): all members

Paper Incrementation (hours = 2): Wayne Nguyen

Image/data set categorization (hours = 3): Hongcheng Jiang

Application design/workflow (hours =3): Fei Wu, Zhaobin Zhang

Participation: all members 25% each

To be Completed:

Refine application design/workflow (hours = 3): Zhaobin Zhang

Get bigger data set (hours = 5): Fei Wu

Implement Algorithms for trained data array (hours = 8): Hongcheng Jiang

Converge algorithms (3-stages) for output array (hours = 8)

Start knowledge recommendation algorithm (hours = 5)

Paper increment 2 (hours =2): Fei Wu

UI implementation (hours = 5): Wayne Nguyen

- [1] Devashish Shankar, Sujay Narumanchi, Ananya H A, *et al*, Deep Learning based Large Scale Visual Recommendation and Search for E-Commerce, <https://arxiv.org/pdf/1703.02344.pdf>, 2017.
- [2] Karen Simonyan, Andrew Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, CVPR, 2014.
- [3] Yann LeCun, Yoshua Bengio and Geoffrey Hinton, Deep Learning, Nature, 2015.