Clothing Similarity Understanding with Deep Learning

Shivani Vogiral
svogiral@iu.edu

Ajinkya Pawale
ajpawale@iu.edu

Aniket Kale
ankale@iu.edu

***Abstract*—The main focus of the project is to find the most suitable clothing matches for the current selection and choice of clothing using existing pre-trained Deep Learning based feature extractors. In addition, we also aim to use Siamese Network to find the similarity between images across different categories.**

I. Introduction

Modern day online search engines rely heavily on their knowledge bases and employ key word matching as a search tactic to locate the most likely goods that customers want to purchase. This is inefficient in the sense that product descriptions can differ significantly from vendor to buyer. In the course of this project we present an interesting way to capture the visual and clothing type similarity that match the real intentions of the user. We focus on fashion products and design a system that returns a ranked list of similar-style clothing items using only a single input image. Furthermore, we also extend our study to finding a similarity score between pairs of images from different categories of clothing.

II. Goals

There are two major problems that we want to solve in our project and they are:

- Given images of clothing find top k similar clothing images using a nearest neighbors ranking method on the features extracted using popular pre-trained Deep Learning based feature extractors.
- Build a Siamese Network Model that shares weights between two VGG16 Networks each producing embedding vectors of its respective inputs using which we compute the image similarity.

III. Dataset

The dataset that we have used is the Attribute and Category Prediction subset of the Deep Fashion Dataset. This is a large subset of DeepFashion, containing massive descriptive clothing categories and attributes in the wild. It contains:

- 45 Image Categories: Blazer, Pants, Dress, Coat ..etc
- 29 Image Attributes: Floral, Graphic, Long Sleeved, Neckline
- Training Set: 14000 images, Testing Set: 4000 images, Validation Set: 2000 images

For the course of our project we have utilized only the Training set with 14000 images as our main dataset. In order to achieve the first goal of the project we have to split this data into train and test sets. Every image in the dataset falls into a unique category and hence when we create the train-test split of the data we have to ensure that 80% of one category falls in the train set and the remaining 20% is counted in the test set. By doing so, we achieve a balanced train test split of the data.

But, as we move on to the remainder of the project we realize that this method of creating a balanced dataset has some drawbacks:

- Ideally we expect to see 45 unique categories in both the train and test sets. As a result of executing balanced train-test split we observed 45 and 41 unique categories in the train set and test set respectively. There are some categories with less than 5 images in them as a consequence of which we do not get a perfect 80-20 split and the data may have fallen into the train set and not the test set.

Before we built our Siamese Network Model, we have utilized the dataset with 20 categories where all of them had at least 100 occurrences.

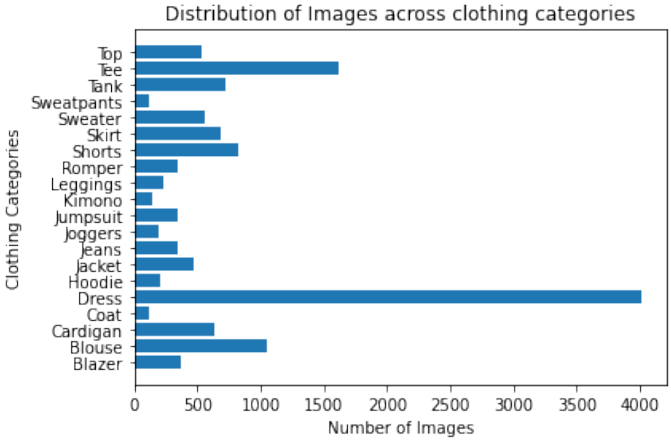


Fig. 1. Distribution of images across clothing categories

IV. Methodology

A. Similarity Computation and top-k ranking using Nearest Neighbours

- The pre-trained Deep Learning based models that we have used are popular CNN based architectures such as:

Inception, ResNet50, VGG16.

- The initial step in using pre-trained deep learning models as feature extractors was to remove the last output layer because these models were not intended to be used as classifiers.
- The output of a convolutional layer is pooled and lowered using a global average pooling layer followed by a flatten layer to provide a linear feature vector for an image in this stage.
- The feature vectors are generated for both the test and train examples.
- In order to compute the similarity scores between feature vector of target image and feature vectors of all images in the training set we use "cosine similarity".
- The principle behind nearest neighbor method is to find a predefined number of training samples closest in distance to the new point, using this idea we can obtain the first k images with feature vectors closest in distance to the test image vector.

B. VGG16 and ResNet50 based Weighted Ensemble Technique

- After reviewing our models, we came to the conclusion that VGG16 and ResNet50 were the best performing models out of all of them, therefore we chose to continue working on them in order to improve our outcomes.
- We used a weighted average of both the training feature vectors to obtain the final feature vector in order to create a rich feature representation that includes both VGG16 and ResNet50 models.
- SKlearn's SelectK method was employed here as a feature reduction strategy, which selects the top K features from a set of features depending on the target value for the features, because the size of ResNet50's feature vector was larger than VGG16's feature vector.

C. Siamese Network Model for Clothing Image Similarity

- Siamese networks are neural networks that share weights between two sister networks having the same architecture and each producing embedding vectors of its respective inputs. In our case we have utilized VGG16 which is a CNN architecture.
- The objective of training such a network is to maximize the contrast between embeddings of inputs of different classes while minimizing the distance between embeddings of similar classes. This results in embedding spaces that reflect the class segmentation of the training inputs.
- As a first step to achieving this goal we extracted feature maps of all the images in the dataset.
- We then created positive and negative pairs of the extracted feature maps. Positive pairs are pairs of images from the same clothing category as opposed to negative pairs that have images from different categories. We created extra negative pairs to improve the robustness of model as it enhances the ability of the model to properly distinguish between positive and negative pairs.
- There are be two input layers in the Siamese Network Model, each leading to its own network, which produces embeddings.
- The two networks are identical copies of each other. In each we have a Global Maxpooling Layer, followed by a Flatten and a Dense layer to extract the linear feature representation ie. the embedding vectors.
- A Lambda layer then merges them using an Euclidean distance and the merged output is fed to the final network.
- In our implementation of Siamese Networks we have used the contrastive loss as the objective function. Contrastive loss is known for its ability to accurately and effectively train the Siamese Network. This loss helps in evaluating how good a job the siamese network is distinguishing between the image pairs.

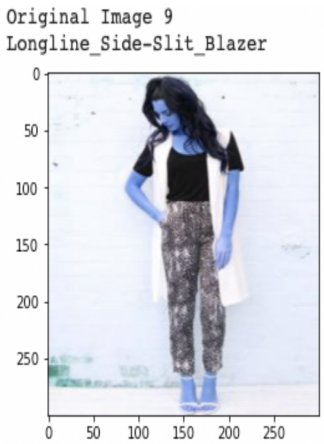


Fig. 2. Example Test Image used for extracting top-k similar clothing styles

V. Results

A. Feature Extraction and Nearest Neighbours matching

- The color of the clothing item is the most closely matched of all the features.
- The input test image belongs to a single class, but the matched features look for patterns in the same class as well as images from other classes.
- As an illustration, if the supplied image is of a flower-patterned top, the returned recommendations will include not only tops but also bottoms(shorts/pants/leggings) with the same patterns.
- We experimented with various k = 5,7,10,... values. Upon observing the results, we discovered that a value of k 5 or 6 is good, and a higher value of k resulted in some images that were slightly unrelated to the input image.
- One major advantage of using the features vectors extracted by VGG16 and ResNet50 was that the results were much more similar looking to the test image than seen earlier and the precision with the model was able to capture the visual similarity is much higher in this case.

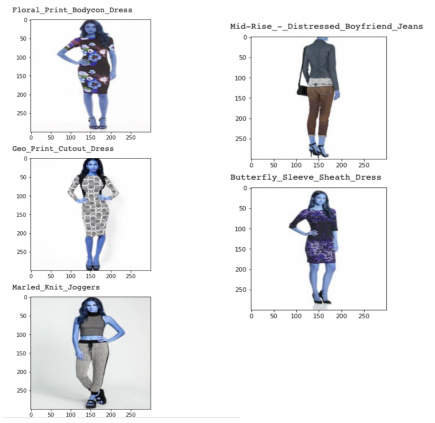


Fig. 3. Results of Nearest Neighbours using feature vectors extracted by InceptionV3

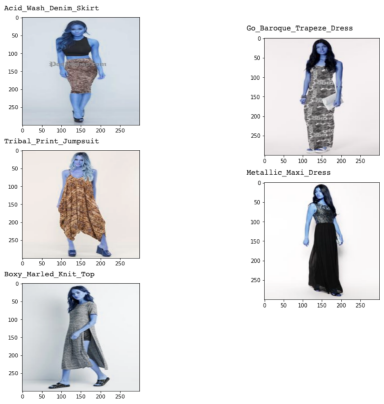


Fig. 4. Results of Nearest Neighbours using feature vectors extracted by VGG16

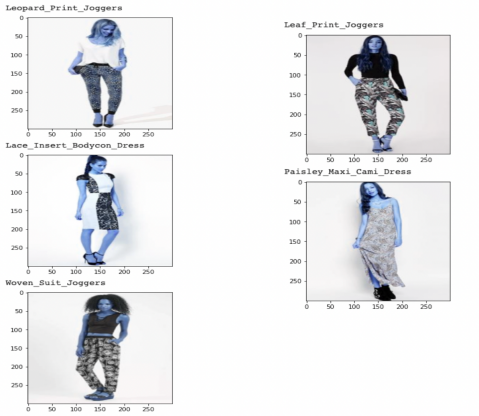


Fig. 5. Results of Nearest Neighbours using feature vectors extracted by ResNet50

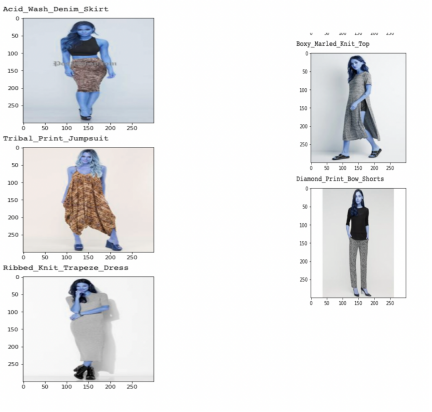


Fig. 6. Results of Nearest Neighbours using feature vectors extracted by Ensemble Method

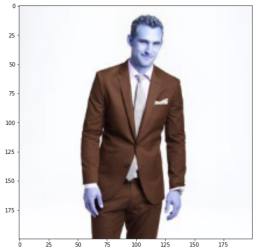
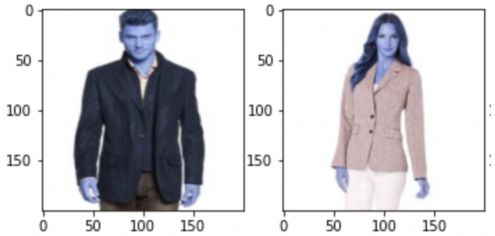


Fig. 7. Input Test Image for Siamese Network Model

B. *Siamese Network Image Similarity*

- Upon seeing the results of the performance of the Siamese Networks we can observe that the model does a near perfect job in matching the category of clothing.
- Besides the type of clothing, color of the clothing item is the next best matched feature.
- Here for instance we have supplied the image of a person in a blazer. The returned similar matches contain people wearing similar types of blazers.
- The Siamese Network incorporated with the feature maps extracted from VGG16 is able to fully capture the visual similarity and clothing category matching the real intentions of the user.

```
Top 5 predicted images:
Similarity: [0.9995795]
Similarity: [0.99961203]
Similarity: [0.99966085]
Similarity: [0.99981993]
Similarity: [0.9998235]
```



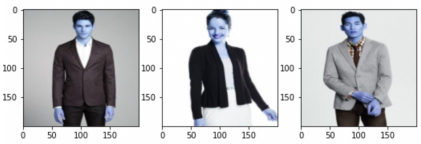


Fig. 8. Results of the Siamese Network Model

C. Evaluation Strategy

- As an evaluation strategy we decided to plot the convergence curves - both loss and accuracy of the Siamese Network. From the convergence plots we can observe a large divergence between the validation and training curves for loss and accuracy.
- There are two possible explanations to such behaviour in the model:
  - This implementation does not utilize the entire Deep Fashion Dataset.
  - In addition to category information, clothing attribute information is also available which can enhance the feature extraction process and contribute to the improving the performance of the model.

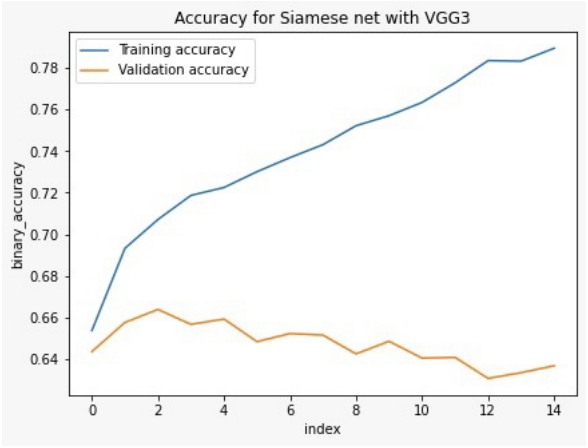
VI. Conclusion

We demonstrated a visually aware, data-driven, and relatively simple system that can produce similar images that captures the discernible features and clothing type similarity successfully by matching real intentions of the user. The Deep Learning based pre-trained feature extractors have benchmarks for very good architectures on achieving our task. The Siamese Network which uses the VGG16 model was able to effectively capture the image similarities using the embedding vectors ie. feature maps of their respective inputs. Our Siamese Network was not only able to compute the similarity of images but could also distinguish between patterns of images from similar and different classes.

VII. Future Scope

Several interesting extensions of our project are possible:

- While generating the positive and negative image pairs for our Siamese Model we can use the image attributes in addition to the category information.
- Instead of using Nearest-Neighbors, we can add a Neural-Net to find similarities between the output vectors (since optimization of weights in ensembling is difficult)
- A major catalyst to improving the performance of both the Feature extraction and Siamese Network is utilizing the entire Deep Fashion Dataset. For our project we used only 14000 images from the training set.



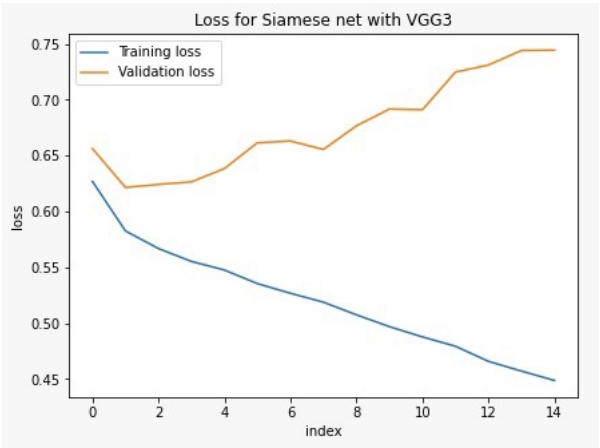


Fig. 9. Convergence Plots

References

[1] H. Tuinhof, C. Pirkir, M. Haltmeier, Image Based Fashion Product Recommendation with Deep Learning

[2] Z. Mustafa, A. Nasar, M. Khan, Image Based Product Recommendation System

[3] L. Chen, F. Yang, H. Yang, Image-based Product Recommendation System with Convolutional Neural Networks

[4] Mehdi, Image similarity estimation using a Siamese Network with a contrastive loss

[5] A. Rosebrock, Contrastive Loss for Siamese Networks with Keras and TensorFlow