E-Weaver: Sustainable Clothing Aggregation and Recommendation System (Modeling)

Matthew Merenich[1], Colin Murphy[2], Dayun Piao[3], Zifeng Wang[4]

mgm344@drexel.edu[1], cjm486@drexel.edu[2], dp636@drexel.edu[3], zw438@drexel.edu[4]

Drexel University: DSCI-592 Capstone II

Abstract

The fashion industry has been grappling with the demands of consumers to shift towards a more "sustainable" manufacturing process and offer consumers better options to meet their moral beliefs. Although many tools are used by industry to measure clothing sustainability, few brands consider the entire "cradle-to-grave" life-cycle of textiles to better their product and practices; more established household brands experience difficulty replacing timely, cost-effective methods for "Eco-friendly" ones. Thus we propose a sustainable clothing recommendation system called *E-Weaver*, which will incorporate state-of-the-art image processing methods for feature extraction and a novel sustainability metric to assist consumers with purchasing more environmentally-conscious products. This paper will focus on the data collection framework for scraping articles of clothing from several different brands to be used in the recommendation system. Additionally, the development of an item-brand level sustainability metric is described. The overall *E-Weaver* framework is outlined and the subsequent applications of the work in this paper are proposed.

1 Introduction

The E-weaver recommendation system's goal is to provide a ranking of similar articles submitted by the user (both aesthetically and monetarily), yet also more sustainable articles.

The system will consist of several different "stages" and will utilize both the clothing item dataset and the sustainability metric dataset. A diagram of the entire system framework is shown in Figure 1. The system input will request the item image, material, price, gender, and brand. With the exception of the image, all inputs will feed directly into the clustering stage.

In many classical item clustering methods, unsupervised learning is applied to entire images themselves to infer their similarities. However, this can be computationally intensive and it can be difficult to infer what precise design details make the items similar – providing only a *course* object-level classification. In order to obtain a refined model we limit the input for clustering to only the pixels that provide information – those that make up the t-shirt.

2 Related Work: Evaluation Metrics

Evaluation metrics or approaches should be consistent with the objectives of applying a recommendation system, and can clearly reflect the benefits or changes after adopting it. So far, most of previous research has been mainly concentrated on the algorithm accuracy of recommendation system evaluation(Adomavicius and Tuzhilin 2005), especially those based on objective evaluation or quantitative metrics (Herlocker and Riedl 2004).

More recently, researchers began examining issues related to users' subjective opinions(Jones and Pu 2007) and developing additional criteria to evaluate recommendation systems(Ziegler and Lausen 2005). In particular, researchers are investigating user experience issues in an effort to understand and identify effective preference elicitation methods(McNee 2003). Some research further focused on developing more unifying evaluation framework for recommendation(Pu and Rong 2011), which aimed at measuring the qualities of the recommended items, the system's usability, usefulness, interface and interaction qualities, users' satisfaction with the systems, and the influence of these qualities on users' behavioral intentions.

Since currently our project still lacks actual user behavior data and business target, we would evaluate the final results of our recommendation system mainly on subjective judgement of item relevance and sustainability relevance.

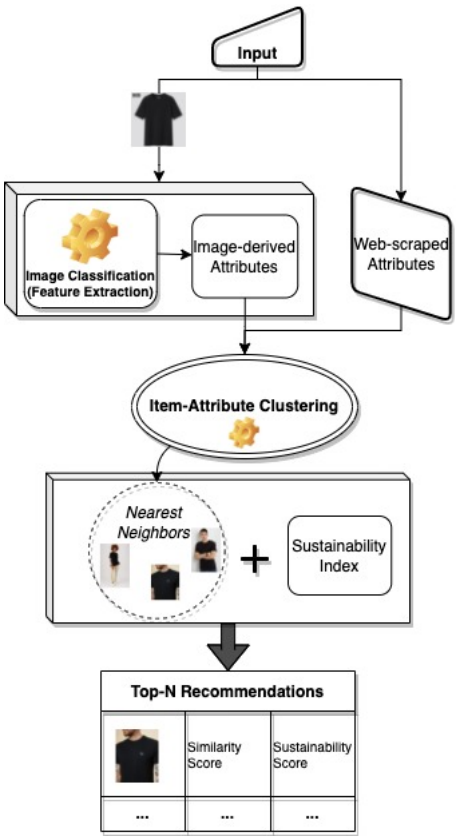

Figure 1: Schematic diagram of entire system framework – image classification and clustering.

3 Methodology

3.1 Data Source

Our dataset is a combination of several sub-sets scraped from various different clothing brand websites. From these sites we obtained meta-data for individual T-shirt products and associated images. Due to the differences in website design and the companies privacy policy, the common meta-data features of the final dataset were limited to the follow: color, gender, price, material, brand, and image. All auxiliary meta-data was still collected for possible use later in the development of the project.

The web scraping code was developed using the BeautifulSoup and Selenium libraries. The code requests information from each E-commerce website and scrapes data in multiple nested loops. After data is collected from each website, they are cleaned and converted to pandas dataframe, then concatenated to form the final dataset. Code can be found in the project GitHub Repo.

Data was obtained from both large brand name companies that are often associated with fast-

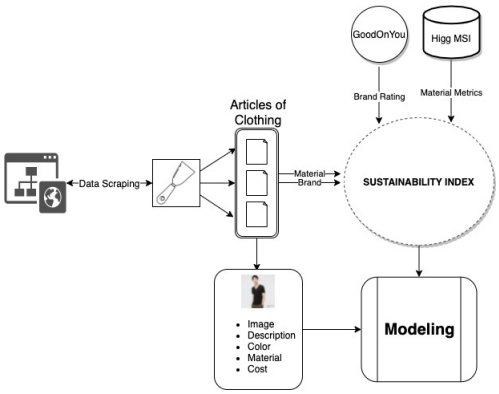

Figure 2: Schematic diagram of the data collection pipeline.

fashion, and smaller companies that have a greater sustainability focus. These will serve as the training data for the eventual model, therefore, we chose several different data sources to offer a better representation of varying sustainability levels.

Additionally, data from two other sites was obtained in order to build our sustainability metric for ranking items – the aggregate brand rating site GoodOnYou.eco and the industry standard MSI from higg.org. The data from these two sites will be fused together to create a metric that takes into account both the item specific metric (higg.org, Higg MSI) and an entire brand metric (GoodOnYou.eco).

3.2 Sustainability Metrics

The framework for the sustainability metric is divided into two main focuses: Item and Brand.

- **Item** - a fusion of the Higg MSI and academic surveys of consumer habits and uses, providing a score based on material blend and garment type. This is intended to be a more granular metric of sustainability.

  The scaled and weighted sustainability indicators of the per-item scores and their relative material ratios are summed up to form the overall score (Eq.1).

$$\text{Item Sustainability} = \sum_i \sum_j w_i \frac{CR_{ij}}{NR_i}(n_j) \quad (1)$$

  Where the characterized result (CR) of indicator, i, for material, j, is divided by the normalization reference (NR) of the corresponding indicator, i. The Higg MSI normalizes according to the industry's annual impact. The division is later multiplied by the weight factor (w) of indicator i –

for our current system, this will simply be set to "1" for all factors. And to account for items consisting of several different materials, the material score is scaled according to the relative content (n) of the material, j, in the item.

- **Brand** - GoodOnYou.eco aggregated rating of the overall brand. This is a more generalized metric of the brand as a whole – e.g. given two garments of the same material can be separated by brand rating.

The methodology we have developed should not be viewed as a perfect ranking for any sustainability purpose; however, the flaws and shortcomings of this are a result of the typical trade offs often faced in comparative analyses – assumptions must be made in order to aggregate disparate variables from across the entire life-cycle. Overall, there is no single material or brand option that performs best in all sustainability categories and so any ranking choice results in a compromise.

4 Modeling: Image Feature Extraction

Two Convolutional Neural Network (CNN) architectures were used for image feature extraction of our dataset — Very Deep Convolutional Networks for Large-Scale Image Recognition-16 (VGG16) and a masked Region-based CNN (R-CNN).

VGG16 is simple and widely used deep layer (16 layers) CNN imported from Keras library. This is one of the best vision architecture for image data related task. In this project, the VGG16 is used to extract the latent features from each image either applied masks generated from Mask R-CNN or not. Then the features matrix is used to calculate similarity matrix of each t-shirts against rest. Also feeding it to the KNN model for final aggregation and ranking.

The masked R-CNN provides a more interpretable instance segmentation, which combines the detection of an object (e.g. a T-shirt) in an image and also precisely segments each instance of it.

4.1 Masked R-CNN

The masked R-CNN is an advanced deep learning model with high effectiveness for object detection/image segmentation (He et al. 2017) — it is an extension of the Faster R-CNN framework by adding a branch for predicting segmentation masks on Regions of Interest (RoI) in parallel with the classification and bounding box branches. The model's instance segmentation for objects are provided as a binary "mask" within the RoI, providing a pixel-level classification.

The success of the masked R-CNN architecture has been proven in several different domains and with transfer learning methods, a relatively small and unrelated dataset can be used to train the model. Within the fashion domain, a recent advancement has even shown that fine-grain attributes of an object can be detected and classified (Jia et al. 2020); however, our dataset consists of non-uniform scrapped data of both high and low resolution photos and is limited to T-Shirts, thus a simple instance masked R-CNN model is better suited.

Using a Mask R-CNN model, pre-trained on a dataset of limited few fashion items such as tops, bags, and boots (https://github.com/sugi-chan/clothes_segmentation), we were able to quickly create instance masks for the dataset using the model as-is without the need to annotate our images.

MR-CNN Results The outputs of the model were assessed in a largely subjective manner – results were reviewed by brand as most brands had a uniform image resolution and modeling pose/background.

Of the 23 unique brands in our dataset, all but one (H&M) had images where a T-shirt instance was found by the model. In figure 3, an example mask from each of the brands are shown. We can see the successful segmentation of the object from the background and the human. However, depending on the person's pose and presence of other objects, the performance can vary – the image for "Mate" has segmented the jacket the person is hold in with the actual shirt, and instances where person has their arm crossed ("Living craft", "Thestandardstitch", and "Zara") the instance is cut off and the lower portion of the shirt is not captured.

MR-CNN Limitations As mentioned, the model often performed poorly when the person has some unique pose and for some instances could not detect a t-shirt at all, although all samples consisted of at least one t-shirt. Of the 2833 images processed by the model, only 1420 objects (50.1%)were found an given masks.

This failure to detect an instance was often when the shirt was placed in a different context. For example, images of a t-shirt without a person wearing it were not detected – 4 of the undetected examples. In figure 4, we also see some unique poses, backgrounds, and even a baby.

Also, the image size played a role - some scrapped images were only the thumbnail-sized image and thus did not provide enough pixel information for the model to detect.

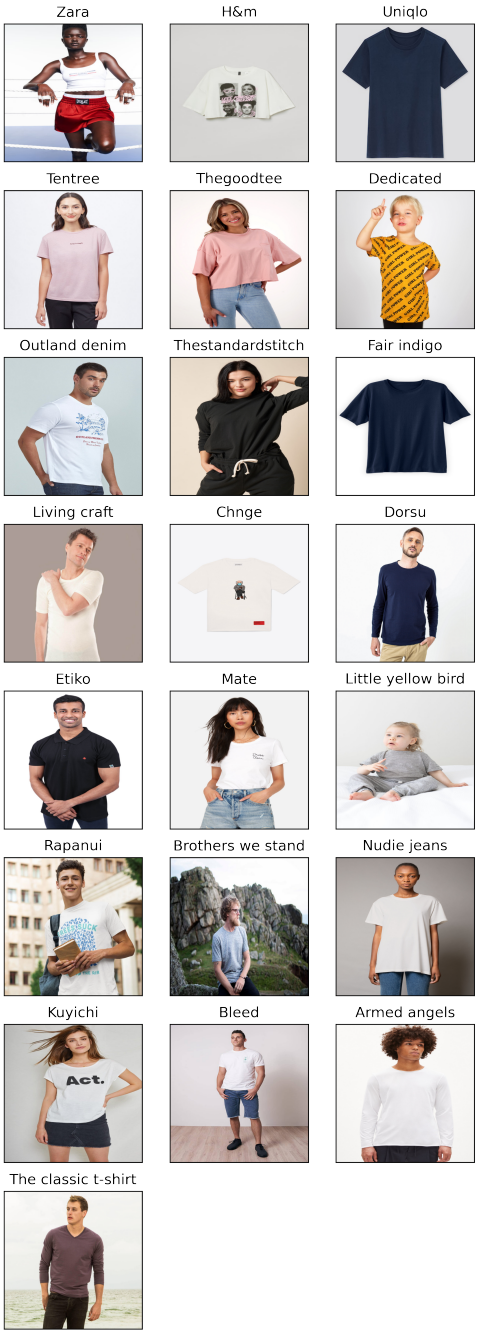

Figure 4: Example images with no mask found

4.2 VGG-16 Processing

The VGG16 model achieved the highest test accuracy score of 92.7% in ImageNet's competition. Consider it as one of the best image classification models yet with simple implementation (Perumanoor 2021). Only drawback is it requires a large amount of time to train neural networks. There are 41472 features learned by VGG16 for cropped size of 300x300 pixel, and then it is used to calculate initial recommend result from cosine similarity equation (Eq.2) (Ma 2017).

$$\mathrm{s(u,v)} = \frac{r_u r_v}{|r_u||r_v|} \tag{2}$$

VGG16 Results It is observed that some outputs from similarity matrix by VGG16 features extraction has good results when raw image only contains T-shirt only, see figure 5.

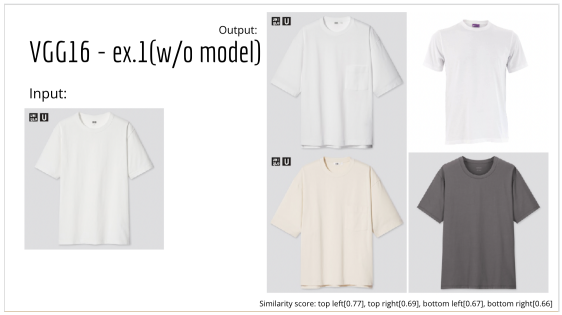

Figure 5: Good Example of Recommendation

While some are very off with completely different t-shirt color, styles when model exists in the raw image, see figure 6.

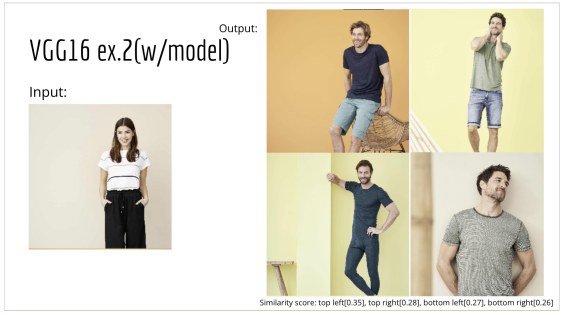

Figure 6: Bad Example of Recommendation

Both of the example figures are from tests without the mask applying from Masked R-CNN and the similarity matrix will give a high score based on image composition. For example, where images have models on the left side of the image leaning against a wall would be grouped and considered similar based on image pixels arrangement. In figure 7, the masks produced from Masked R-CNN is applied to the raw image and extracted features again from VGG16.

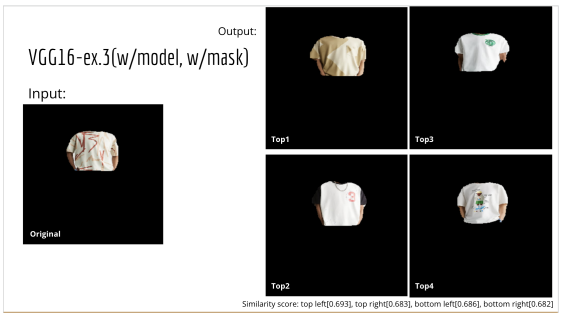

Figure 7: Masked Example of Recommendation

The figure 8 is where the result are interpreted through finding the original image index. The recommended T-shirts are not very close to the input image, due to the results of raw image applying the masks.

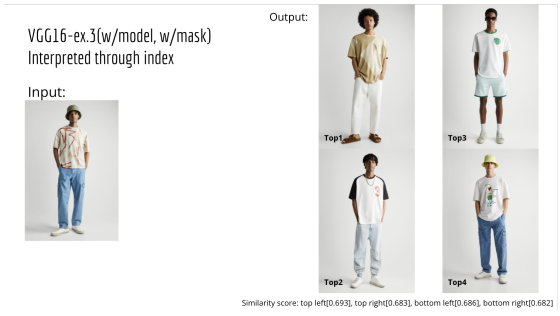

Figure 8: Reversed Masked Example of Recommendation

VGG16 Limitations At this stage, content based recommending algorithm through VGG16 features extraction gave unsatisfied result. It is due to the presence of elements other than T-shirt. The masks are suppose to ideally make the image data clean and only contain the T-shirt itself. But since there are only half of the images have masks and the segmentation contour is often time not precisely reflect the T-shirt contour, it lower the accuracy when recommending a truly related output. If masks can improve their accuracy when segment out the T-shirt, it is believed the recommending T-shirt from similarity matrix will gave a better result.

5 Modeling: Clustering with Nearest Neighbors

Our final phase in the data pipeline required training a model to produce the most similar t-shirts as the input it was given, and rank them by their sustainability index. All previous steps in our project were to develop the most essential training data for our last model: the combination of the attributes gathered in the web-scraping phase with our image features extracted from the previous steps (MR-CNN and VGG16 modeling). We selected the K-nearest neighbors (KNN) clustering algorithm for its ability to handle the tens of thousands of features accumulated from our other models, the relatively simplistic training process, and unsupervised modeling capabilities. The final version of our KNN model was a NearestNeighbor method (unsupervised) using cosine similarity (metric = cosine) to determine distance in the testing phase, MinMax scaling of the data, with an output of 10 closest results (n_neighbors = 10) of the closest results as output.

KNN Results Visualizing each model was the best way to gauge how accurate our results were, and developing separate training datasets based on each of our feature extraction models (masked R-CNN and VGG16) allowed us to grasp how effective each stage of our final process at producing the desired output.

We developed three NearestNeighbors models: the first trained on just the feature data extracted via web-scraping, the second on the feature data and additional image features of the raw images from the VGG16 model, and the final on feature data and VGG16 features of the masked images. The input (the same shirt for all testing) and 4 nearest neighbors of these models are shown below.

Figure 9 shows the results of training a NearestNeighbor model on features like price, brand, and material composition. The key takeaway is the "brand" feature played heavily in producing the nearest neighbors.

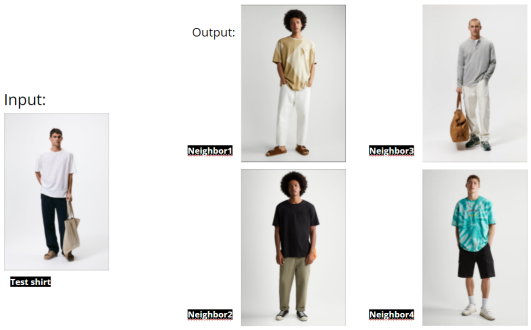

Figure 9: Results of NearestNeighbor (NN) model trained on web-scraped qualitative features

Our next test result in Figure 10 shows the results of training a our model on the initial features as well as the image features from VGG16 processing of the raw image for each t-shirt. Here we can

see there are different brands in the output, a step closer to our final goal. However, the shirt colors and style compared to the test image still indicate room for improvement.

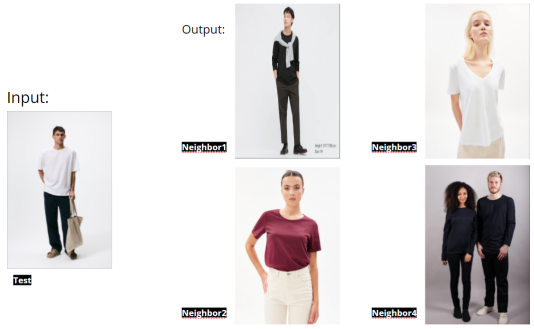

Figure 10: Results of NN model trained on scraped features & VGG16 raw image features

Figure 11 shows the results of our final model, which was trained on the same basic features with the addition of VGG16 features of the masked images from our masked R-CNN model. This was our best result - the color of each t-shirt is similar, and there are multiple brands within the entire 10-neighbor output.

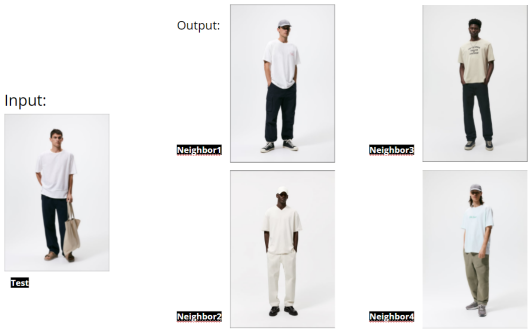

Figure 11: Results of NN model trained on scraped features & VGG16 masked image features

Though subtle, the difference of each model serve as a progression of our feature extraction and validate the implementation of each segment of the data pipeline thus far.

KNN Limitations As we have just seen in our KNN results, and as is with MR-CNN evaluation, analyzing the output of an unsupervised KNN model and determining it "successful" is almost entirely subjective – with a lack of metrics to aide in defining accuracy, we relied heavily on our knowledge of the database as well as the ultimate goal of the project to conclude the model a success. Ideally we would develop a more quantitative process to measure our output for optimal tuning and remodeling.

6 Conclusions

The focus of this work has been data modeling with the following accomplishments:

1. The application of CNN architectures to extract principle features from the images prior to cluster.
2. Trained simple Nearest Neighbors to cluster images and create rankings of similarity.
3. The further development of a specialized sustainability metric for ranking to incorporate both item and brand level details for a nuanced comparison of items of the same material composition.

Further work would focus on developing the implementation of utilizing customer feedback for model updates, and an interface for a UX/UI component for a more collaborative experience during testing.

References

[Adomavicius and Tuzhilin 2005] Adomavicius, G., and Tuzhilin, A. 2005. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17:734–749.

[He et al. 2017] He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

[Herlocker and Riedl 2004] Herlocker, J.L., K. J. T. L., and Riedl, J. 2004. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst* 22:5–53.

[Jia et al. 2020] Jia, M.; Shi, M.; Sirotenko, M.; Cui, Y.; Cardie, C.; Hariharan, B.; Adam, H.; and Belongie, S. 2020. Fashionpedia: Ontology, segmentation, and an attribute localization dataset. In *European conference on computer vision*, 316–332. Springer.

[Jones and Pu 2007] Jones, N., and Pu, P. 2007. User technology adoption issues in recommender systems. In *Proceedings of Networking and Electronic Commerce Research Conference*, 379–394.

[Ma 2017] Ma, W.-J. 2017. Deep learning meets recommendation systems.

[McNee 2003] McNee, S.M., L. S. K. J. R. J. 2003. Interfaces for eliciting new user preferences in recommender systems. In *Proceedings of International Conference on User Modeling*, 178–187.

[Perumanoor 2021] Perumanoor, T. J. 2021. What is vgg16? — introduction to vgg16.

[Pu and Rong 2011] Pu, P., L. C., and Rong, H. 2011. A user-centric evaluation framework for recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems*, 157–164.

[Ziegler and Lausen 2005] Ziegler, C.N., M. S. K. J., and Lausen, G. 2005. Improving recommendation lists through topic diversification. In *Proceedings of the 14th international conference on World Wide Web*, 22–32.

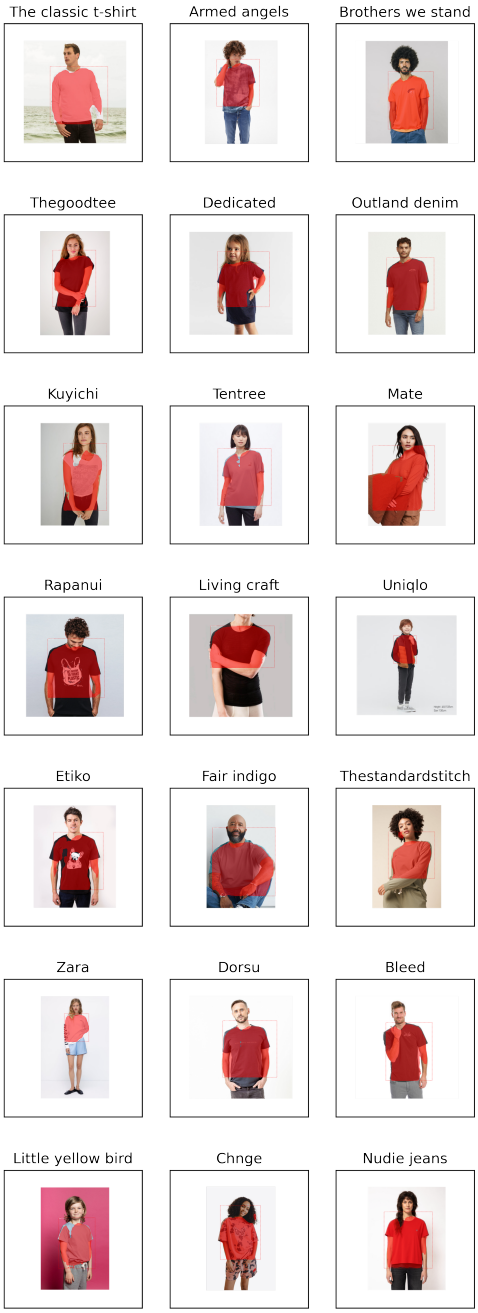

Figure 3: Example masks by brand