RealTimeML Image Analytics and Prediction Recommendations: Data Augmentation Model

Casey Yoon

loxz digital

Abstract

This RealTimeML Data Augmentation model provides real time predictive analytics on the images used for digital marketing campaigns. The model is essentially an image classification model that will identify the best image augmentations and alternate images and recommend these options for higher target variable conversion rates. Target variables are chosen from a drop down menu and recommendations will provide five image augmentations and several, if any, alternate images.

Ⅰ. DATA

A. Data Source

The dataset used in this analysis is a compilation of athletic shoe & sneaker images scraped from https://www.zappos.com. The dataset contains roughly 10,000 images of sneakers. With data augmentation techniques applied onto each image, our dataset size increases **six-fold**, totaling **60000 images of sneakers**. The models used in this project are trained specifically on these sneaker images.

B. Data Set

Along with web scraping images, several other features were extracted:

- relative_im_path: file path to the image
- shoe_brand: shoe brand
- shoe_gender: shoe gender type (men, women, boy, girl)
- shoe_name: name of the specific shoe
- augment_label: One of the applied image augmentation techniques
  - {0: 'brighter', 1: 'darker', 2: 'original', 3: 'rotate', 4: 'sketch', 5: 'swivel'}

The following image augmentation techniques were applied to the original 10,000 images:

1. Brightness adjustments (both brighter and darker)
2. 2D 90-degree rotation
3. Pencil sketch
4. Swivel - 3D 30-degree rotation on its side

Future models will utilize existing historical data alongside the introduction of new datasets to model target variable conversion rates; however, for this iteration, we synthetically create variables and assign random conversions. These are our target features:

- **open_rate:** a binary value for an open rate event
- **click_through:** a binary value for a click-through event
- **abandoned_cart:** a binary value for an abandoned cart event
- **unsubscribed:** a binary value for an unsubscribed event
- **campaign_type:** categorical variable describing email campaign type
- **industry :** categorical variable describing industry type

C. Tools used to build Dataset

The images and dataset features were scraped utilizing a Python script using the BeautifulSoup package, a tool that parses through and can extract data from HTML documents. In our case, the Zappos website contains a listing of shoes based on gender types, highlighting names, brands, and showcasing the shoe image. The Beautiful Soup takes the html page and allows us to parse its contents extracting png files, and the respective shoe names and brand names that are featured. Our model and algorithms are all

housed in Python notebooks hosted in Google Colab. Future iterations of this model will be built in Sagemaker using Jupyter notebooks. All models within the Loxz RealtimeML predictive models are now being converted to SageMaker.

D. Feature Engineering

To simulate real world conversions, our synthetic variables, the target features, were randomly assigned a binary success variable based on augmentation type. For the target variables like "open_rate", or "click_through-rate" we assigned 80% to be a successful conversion if the image was a brighter augmentation or a rotated augmentation, whereas the other augmentations were assigned 20% to be a successful conversion. This 80%-20% assignment is arbitrary and is only meant to highlight the effectiveness of our model.

Model & Algorithms

A. Model Development

Image classification problems usually require a two-step process:

1. Converting images to vectorized embedding using a neural network
2. Feed vectorized embeddings to a target variable classification algorithm

B. Algorithms Used

The first step of our model utilizes the pre-trained ResNet50 CNN (convolutional neural network). Convolutional neural networks are artificial networks most commonly applied to visual imagery analysis and in our case, we use a pre-trained version with 50 hidden layers to better classify images into vectors – where vector lengths equal to the number of augment classes. As we have thousands of images, we utilize **Pytorch** to handle this deep learning algorithm. The use of this algorithm correctly recognizes images to their augment labels which is significant for the next step of our model.

Once we have our vectorized image embeddings, we feed them to our XGBoost Classification algorithm. This ensemble algorithm has fast execution speeds and hosts optimal model performance amongst other methods in classifying the success of a target variable from these vectors.
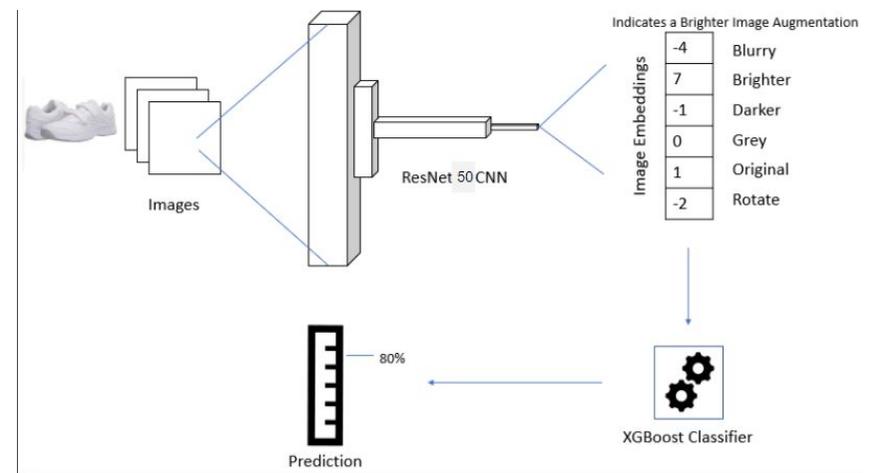

Fig. 1. ResNet50 CNN w/ XGBoost Classifier Model Architecture

C. Model Assumptions & Validation

All images in the dataset must have uniform structure and format to allow for image augmentation techniques to be applied uniformly across all images, otherwise such techniques will be applied improperly.

This model is trained on existing data of conversion rates for sneaker images and aims to replicate conversion predictions given this history. For example, if historically, brightening an image caused an increase in conversion rates, then our model will predict a higher conversion rate for the user. Future assumptions can include but not limited to image styling, background, and borders to ascertain higher engagement rates. A brighter image might help along with a darker background image.

The first step of our model categorizes images to their vectorized embeddings with a test accuracy of **99.5%**, allowing us to significantly trust our neural network to recognize which image augments have been performed on the original image. Next, given our data and our synthetically derived 80% conversion rates for

brighter and rotated image augmentations versus the 20% conversion rates for all other augmentations, the XGBoost Classifier gives a 79% test accuracy. Our classifier test accuracy suggests that it is closely predicting our derived conversion rates.

For this model and its future iterations, we focus on accuracy over other scoring methods like precision/recall because this model is only concerned with the percentage of correctly labeled conversions.

TABLE 1. Model Results

Model	Accuracy
ResNet50 CNN	99.5%
XGBoost Classifier	79.5%

Note: For further clarification, the XGBoost Classifier accuracy should equal the true conversion rate. For this model and its future iterations, we focus on accuracy over other scoring methods like precision/recall because this model is only concerned with the percentage of correctly labeled conversions.

Ⅲ. USE CASES

For an email marketing campaign advertising sneakers whose goal is to drive conversion rates, our model provides image augmentations or alternate images that have historically proven to showcase better conversion rates. This is a single RealTimeML use case inferring higher engagement rates prior to the campaign engineer clicking send. A consideration can be made that when you blend one or more models together, such as this Data Augmentation model with a Send Time optimization model, you could derive higher engagement rates. These rates are shown as an inference in an in-session recommendation all in RealTime.

Ⅳ. NOTEBOOK USAGE

The *Data_Augmentation_Model.ipynb* notebook hosts the final product where it asks for an image input and features a drop down menu for these variables:
{Target variable, Campaign Type, Industry Type, Sneaker Type}.

The make_prediction_recommendation function gives the following:

1. Probability of conversion for the selected image in RealTime
2. Augmented images and their predicted probabilities of conversion
3. Recommendations on the best augment and probability of conversion
4. Alternate images and their probabilities of conversion
5. Recommendations on the best alternate image and probability of conversion

We note that the image recommendations are derived from a Neural Network model designed to model real-world conversion rates. There is no causal explanation for these RealTIme Online predictions; however, we are able to deduce that the recommended images and image augmentations will be the most effective in driving higher engagement rates and ultimately conversions.

V. CONSTRAINTS/LIMITATIONS

Due to GPU limitations in Google Colab, only 5000 of 60000 images were used in training. This reduced dataset, however, **does not** impact the accuracies of our models. Test accuracies for the image embeddings to their respective augment labels were at 99.5%. Utilizing more images in training does not increase conversion rate accuracy, as it merely improves the categorization process of image embedding extractions – which already proves to have a high success rate.

Different image sets will require different image embedding extraction methods (e.g., handwatch images - ResNet34 CNN; shoe images - ResNet50 CNN) Certain image objects require more deep training than others. As we begin to work with additional datasets we will report on which ResNet Layer CNN work best with each type of product.

Google Colab has runtime issues with accessing folders with large amounts of images – file path scans through a folder of 60000 images at times results in an error.
 If the make_predictions_recommendation function gives an input error for an image, simply run the function again.

VI. CONCLUSIONS

As the data augmentation model is based on real-world data, the effectiveness of our predictions and recommendations will be directly correlated to the conversion rates of the alternate and augmented images. For example, if there are 8 of 10 conversions for brighter images, our model predicts a successful conversion at an 80% rate. In summary, historical data is paramount and acts as the basis for our predictions model that recommends the best alternate and augmented images for driving optimal conversion rates.

VII. FUTURE CONSIDERATIONS

Depending on the image dataset – currently, our model is based on images of sneakers – we will look to add more image augmentation techniques like changing the background color or changing the color of a certain aspect of the image object. For example, if our model is based on hand watch images, then changing the band color would be of interest. Our current lineup of image augmentation techniques are fairly broad and can be applied to multiple image objects; however, more specific augmentation techniques can be applied given a specific image object.

RealtimeML Predictions based on image augmentation is part of the family of RealTImeML models at Loxz Digital.

VIII. ABOUT CASEY YOON

Casey Yoon is a Data Scientist with a Masters Degree in Information & Data Science from the University of California, Berkeley. As a relatively new hire as of November 2021, he looks to make his mark in the email space with his experience in computer vision and building machine learning models, which has enabled him to develop the RealTime Data Augmentation Model for Loxz Digital.