Home Décor Helper

using Deep Learning & Natural Language Processing

Anjali Sebastian - December 11, 2018

Motivation

I often see furniture that I like, either in catalogues or at a friend's house and I wonder where I can buy this from or is there a similar product that is available for a lower price. The Home Décor Helper uses images which can be from a catalogue or photographs taken by the user to find products that are similar. It uses a combination of image classification and text analysis to provide the most appropriate suggestions. I decided to focus on Living Room Furniture to start with, as I felt that furniture in the living room typically is the first thing one notices in a person's house and it something that most of us are more invested in when making a purchase decision vis-à-vis other types of furniture.

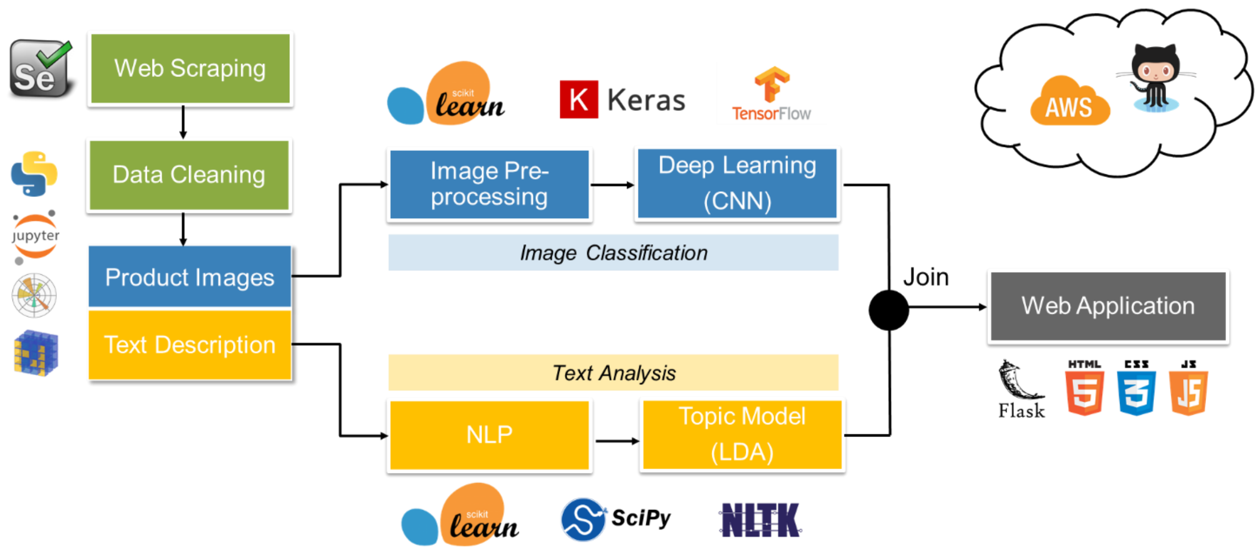
Data

I collected around 36,000 product images along with their descriptions from overstock.com. I used selenium to web scrape from the site. I only collected the URLs to the images and did not actually download the images. I did this since it was easier to main a list of URLs for manipulating and cleaning data before modelling. Along with the image links I collected the following information for text analysis

1. Description
2. Style
3. Material
4. Type_class
5. Color

While collecting the data I labeled it in batches based on the 5 categories of living room furniture that I was looking at i.e. Sofas, Ottomans, TV Consoles, Coffee Tables and Armchairs.

Design



Web Scraping: I used selenium to web scrape data from overstock.com. I first collected links from the banners and then used these links to hit individual pages to collect information.

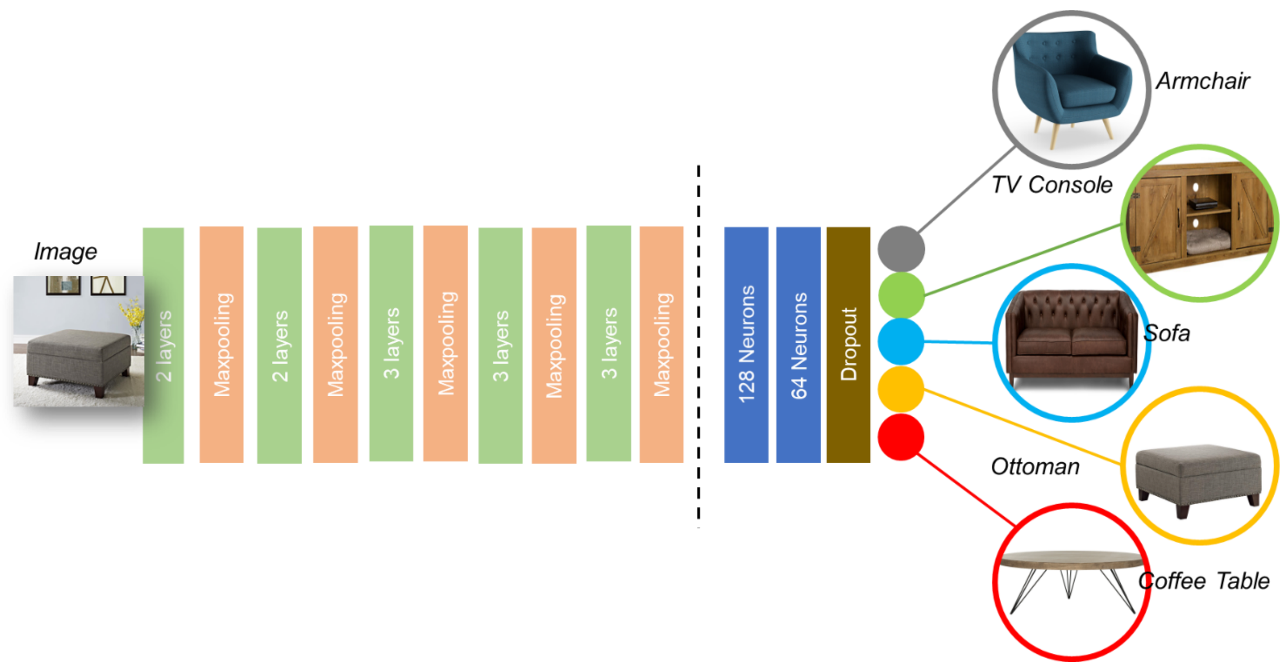
Data Cleaning: Cleaned the text data using pandas, numpy however, I had to clean the images manually. I manually removed images that were zoomed in, had in correct viewing angles…Here are samples of some of the images I removed

Although time consuming, after carrying out manual cleaning my model accuracy went up to 91% from 81%.

Splitting the image data: I did not use all 36,000 images to train my Convolutional Neural Network. I used a balanced set of images from each category/class of 3500 images each. However, for my recommender I used all the images collected.

Why? 17,500 image that is 3500 per class was enough to train my model. I did not want my model to be biased in its learning towards any one type of furniture category since in the real world there is an equal probability that the user picks any category of furniture to find a recommendation.

Deep Learning Convolutional Neural Network:
I used the clean image URLs (17,500) and pre-processed them into image matrices (224,224,3) since the VGG16, VGG 19 and ResNet50 CNN models use an input 224 px by 224px across the 3 channels (RGB).



After trying out several combinations, I used the pre-trained VGG16 after removing its top layers and adding the following layers

1. 128 Neurons Fully Connected Layer
2. 64 Neurons Fully Connected Layer
3. Dropout (0.5) Layer
4. Softmax Layer for classifying 5 classes of Living Room Furniture

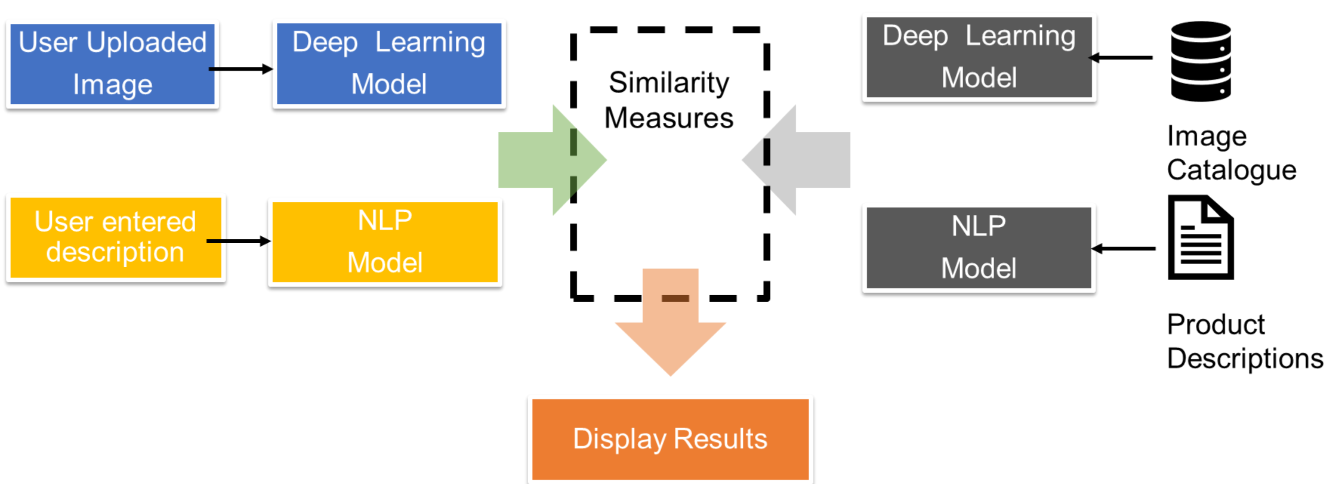
Running the model – used adam optimizer, 70-30 data split for training and testing

Text Analysis

I processed and tokenized the product description with 1 to 3 ngrams with the snowball stemmer. I carried out topic modelling with LDA across 7 topics. I processed my entire corpus of text descriptions and used the log transform of the topic features generated for my recommender.

Making the Recommender

1. **Feature Extraction for all images:** After developing the image classification model using Deep Learning CNN, I carried out feature extraction for all the images by using the first Fully Connected Layer of 128 neurons. I used the model to predict these features on the entire image catalogue. I had to perform this in parts as the AWS machine was unable to process the entire dataset due to memory constraints. Once all the image features were extracted for the entire catalogue this was stored to be later used by the recommender.
2. **NLP Features:** Features generated during text analysis for the entire data were stored to be used later by the recommender
3. **User workflow:** The user uploads an image and an optional text description. The user image and text description are processed in the same way the image catalogue and product descriptions were processed to extract features. To find similar products first Euclidean distance is used to find the top 15 closest images and then these are sorted by text similarity using cosine similarity measure. The second stage is omitted if the user does not give a text description. The top 6 closest options are then displayed.

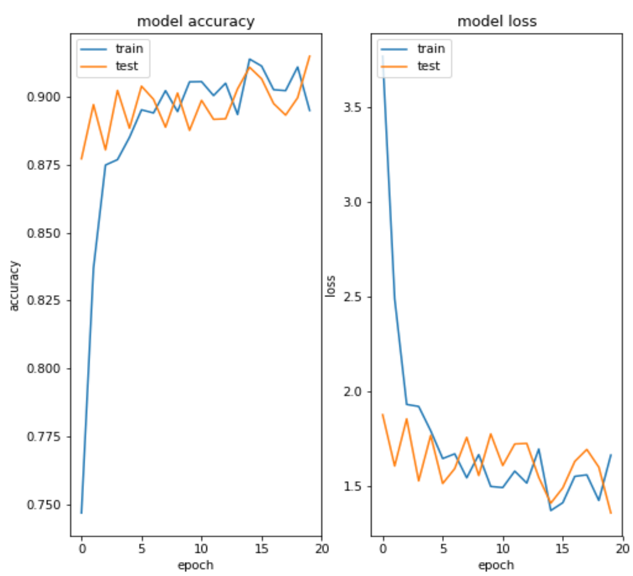


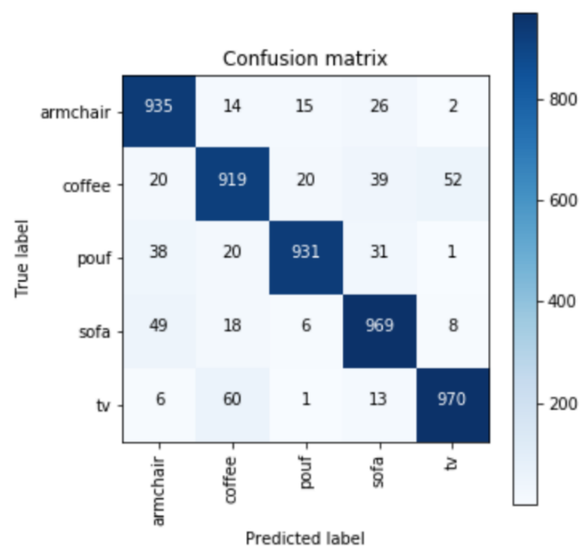
Tools

IDE	Jupyter Notebooks	Atom	
Data handling and cleaning	Numpy	Pandas	
Data Visualization	Matplotlib	Seaborn	pyLDAvis
API development	Flask	Javascript	HTML, CSS
Modeling	NLTK	Sklearn	Keras, TensorFlow
Documentation	Powerpoint	Typora	Pages
Version Management and Cloud Computing	Github	AWS	

Results

Deep Learning Convolutional Neural Network
Achieved an accuracy of 91% on the validation. 15th epoch for model was used.





The confusion matrix shows that the model performs well across all class. However, it is important to note that two class pairs tend to be mistaken for each other much more.

1. Sofa – Armchairs: They look similar and sometimes it is not very clear where the armchair could be a part of a complete sofa set
2. Coffee tables – TV consoles: Their shape and forms are very similar and this is probably why

Topic Model Results

Topic: 1 - Modern Contemporary Furniture
centuri , mid , mid centuri , centuri modern , mid centuri modern

Topic: 2 - Sofas that are comfortable
sofa , contemporari , comfort , seat , grey, cushion , leather

Topic: 3 - Cocktail/Coffee Tables
tabl , cocktail , cocktail tabl , tabl coffe , tabl coffe tabl , accent

Topic: 4 - wooden tables that are brown rustic
tabl , coffe , coffe tabl , wood , brown , rustic , home

Topic: 5 – ottomans,chair,poufs,armchairs
ottoman , chair , pouf , armchair , contemporari , home , seat , blue , storag , comfort

Topic: 6 – TV, entertainment, storage
stand , consol , entertain , storag , media , contemporari , brown , shelv ,

Topic: 7 – coffee tables contemporary modern glass
table , coffe , coffe tabl , contemporari , modern contemporari , glass

Real World Use

I developed a web application where the user can upload an image and type an optional text description to find products that are similar.

Use Case 1: Finding a Cheaper option

I uploaded the image of a very expensive ($1500) Crate and Barrel Coffee Table along with the following text description “rustic wood wooden distressed pine oak reclaimed coffee table natural” to get these recommendations.

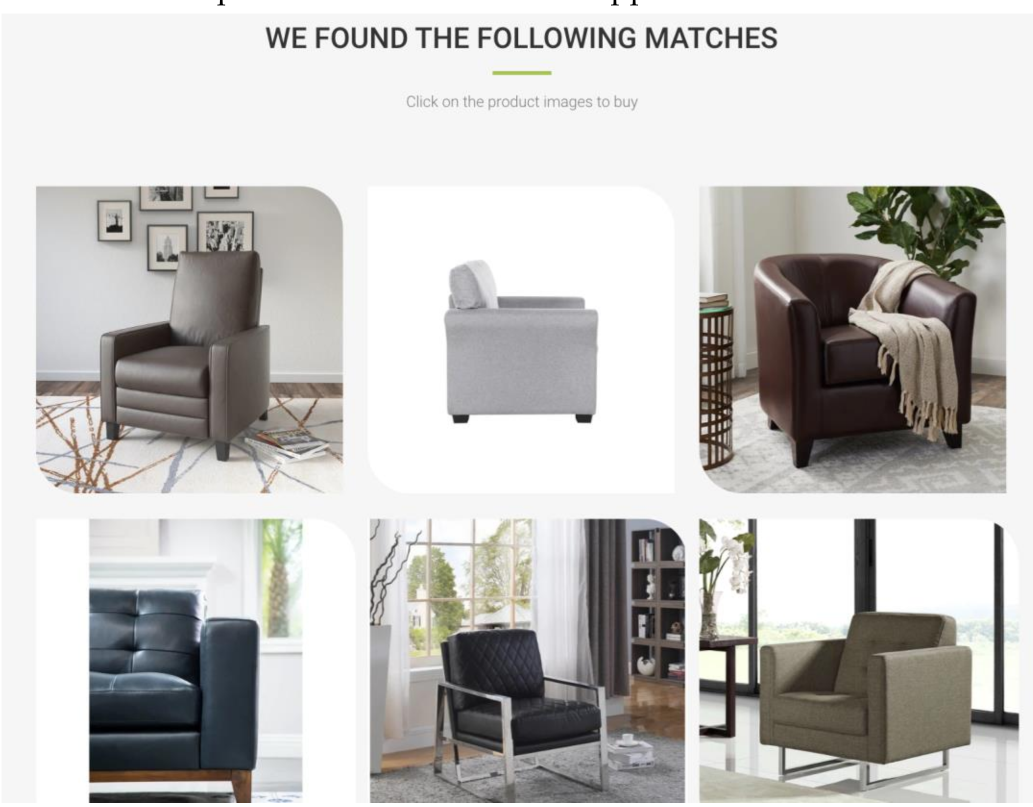




<u>All these options are less than a $1000</u>

Use Case 2: Find products from pictures I have taken
I took a photograph of an armchair at Metis Data Science and used my web application to find similar products.

Screenshot of options shown on the Web Application



Challenges

- Text analysis not as effective – The text analysis was not as effective a filter as I would have liked it to be - the reason being that a lot of descriptions were very subjective and sales oriented with words like elegant, beautiful, etc. I had to use a low max_df of 70% to get rid of words that kept occurring very frequently to extract meaningful words.
- Dealing with very large data files – For most of my project I had to use AWS Deep Learning p3.2xlarge large machines to run my model and store data. Even with AWS memory failure was a major issue as some of the data I was loading were close to 20gb and had to be split appropriately for processing.
- My photos vs catalogue photos – It took a couple of photographs for my photos to produce results vis-à-vis using catalogue images. I had to make sure the lighting condition were sufficiently. Viewing angles, background and lighting conditions can affect model outcomes and going forward this is a challenge that I would like to address more fully

Future Work

- More tolerant to bad photography – learn from user images.
- Add more categories and products from other websites
- Add hard filters to avoid bad suggestions. Have multilevel (nested) recommendations where user can selected the images he or she likes and the application now uses theses to find further recommendations.