**Recognizing Brand Usage patterns using Social Media Images**

|  |  |  |
| --- | --- | --- |
| Karan Barai | Gauravi Chaudhari | Pranav Swaminathan |
| College of Engineering  Northeastern University  Boston, United States of America  barai.k@husky.neu.edu | College of Engineering  Northeastern University  Boston, United States of America  chaudhari.g@husky.neu.edu | College of Engineering  Northeastern University  Boston, United States of America  swaminathan.p@husky.neu.edu |

***Abstract*- The purpose of this research is to identify brand usage patterns by social media activities of users. We are using images posted online to identify logos present in them for examining visibility of brands in social media. We are performing sentimental analysis on tweets to analyze user behavior towards brands. Hence, we are trying to combine brand visibility and user sentiments data which is not readily available. Every image is passed through object detection function which identifies logo in the image, crops it and passes this cropped image to convolutional Neural Network. We have collected data related to starbucks and dunkin donuts. After subjecting our data to their respective models our analysis suggests that dunkin donuts has a better brand visibility but starbucks received higher number of positive tweets. Our convolutional neural network model gave an accuracy score of 98% on subjecting it to 10 epochs.**

***Keywords-*** **Convolutional Neural Network, Object Localization, CNN, Computer Vision**

1. INTRODUCTION

To extract brand visibility information from images we are looking for pixel ranges of brand logos that we are interested in and drawing bounding box around it. To gain more accurate classification results we are passing the localized parts of image to CNN for prediction. To create the data for visualizing brand usage analytics, we are extracting text of user’s tweets based on their username as well as the hashtags. Since, we are getting the text part of user’s tweets, we are parsing the sentiment of the tweet text using Google language API to store the sentiment of each tweet to understand the user’s experience with that brand. The sentiment is visualized in the form of “Positive”, “Neutral” and “Negative”. We have created a Convolutional Neural Network for classifying brand logos in different brand classes which is explained below:

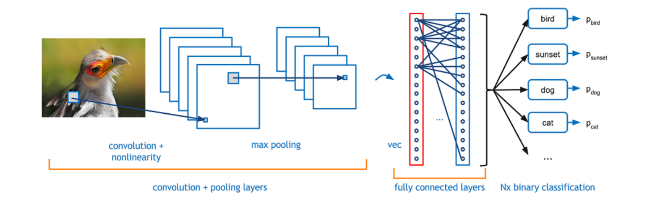


Fig.1. CNN – Architecture

1. *Convolutional Layer:*

The first layer in a CNN is always a Convolutional Layer. Consider input image as 32 x 32 x 3 array of pixel values Now, the best way to explain a conv layer is to imagine a flashlight that is shining over the top left of the image. Let’s say that the light this flashlight shines covers a 5 x 5 area. And now, let’s imagine this flashlight sliding across all the areas of the input image. In machine learning terms, this flashlight is called a filter and the region that it is shining over is called the receptive field. Now this filter is also an array of numbers (the numbers are called weights or parameters). A very important note is that the depth of this filter has to be the same as the depth of the input, so the dimensions of this filter is 5 x 5 x 3. Now, let’s take the first position the filter is in for example.  It would be the top left corner. As the filter is sliding, or convolving, around the input image, it is multiplying the values in the filter with the original pixel values of the image (aka computing element wise multiplications). These multiplications are all summed up and we get a single number. Now, we repeat this process for every location on the input volume. (Next step would be moving the filter to the right by 1 unit, then right again by 1, and so on). Every unique location on the input volume produces a number. After sliding the filter over all the locations, you will find out that what you’re left with is a 28 x 28 x 1 array of numbers, which we call an activation map or feature map. The reason you get a 28 x 28 array is that there are 784 different locations that a 5 x 5 filter can fit on a 32 x 32 input image. These 784 numbers are mapped to a 28 x 28 array.

1. *Pooling Layer:*

Pooling is a sample-based discretization process. The objective is to down-sample an input representation (image, hidden-layer output matrix, etc.), reducing its dimensionality and allowing for assumptions to be made about features contained in the sub-regions binned. This is done to in part to help over-fitting by providing an abstracted form of the representation. As well, it reduces the computational cost by reducing the number of parameters to learn and provides basic translation invariance to the internal representation. Max pooling is done by applying a max filter to (usually) non-overlapping sub regions of the initial representation. Let's say we have a 4x4 matrix representing our initial input. Let's say, as well, that we have a 2x2 filter that we'll run over our input. We'll have a stride of 2 (meaning the (dx, dy) for stepping over our input will be (2, 2)) and won't overlap regions. For each of the regions represented by the filter, we will take the max of that region and create a new, output matrix where each element is the max of a region in the original input.

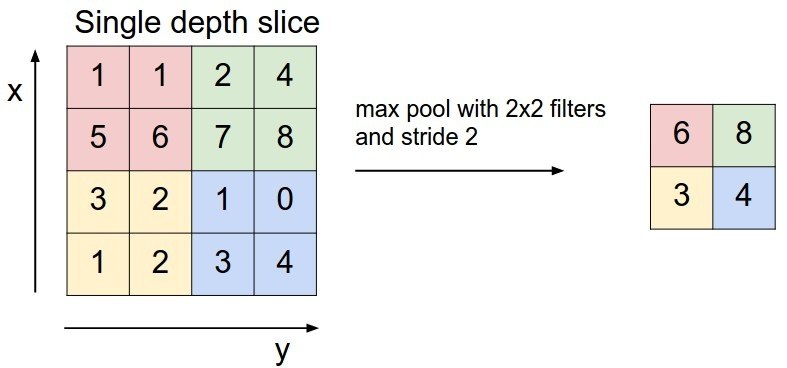


Fig.2. Max-pooling

1. *Fully Connected Layer:*

This layer basically takes an input volume (whatever the output is of the conv or ReLU or pool layer preceding it) and outputs an N dimensional vector where N is the number of classes that the program has to choose from. For example, if you wanted a digit classification program, N would be 10 since there are 10 digits. Each number in this N dimensional vector represents the probability of a certain class. The way this fully connected layer works is that it looks at the output of the previous layer (which as we remember should represent the activation maps of high level features) and determines which features most correlate to a class. Basically, a FC layer looks at what high level features most strongly correlate to a class and has particular weights so that when you compute the products between the weights and the previous layer, you get the correct probabilities for the different classes.

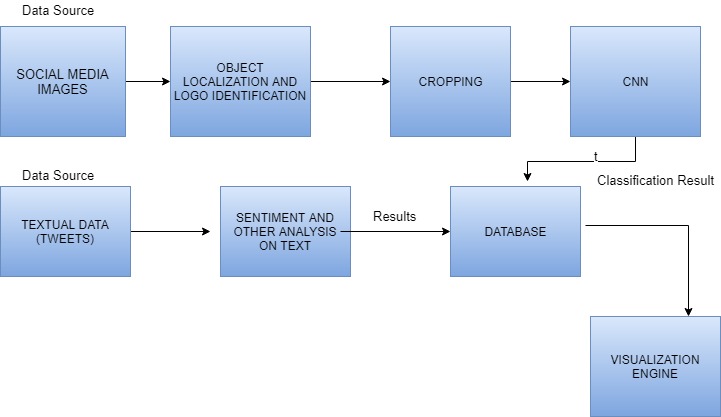


Fig.3. Architecture of the project

1. *Object Localization:*

Algorithm we are following for object localization -

1. We are using OpenCV in Python 2 for converting the RGB pixel values to HSV (Hue, Saturation, Value) in order to identify the regions which have the color intensity similar to that of the brand logo.
2. We are defining the two boundary ranges (min\_red, max\_red) for the OpenCV library to skim through image pixels and yield required areas. That area is stored in a variable by invoking the cv2.inRange function.
3. After identifying the regions most important to us, we use the cv2.morphologyEx which is a morphological operation for eroding the boundaries followed by image dilation which is necessary because erosion also shrinks the image.
4. Once, we have identified the brand logo, we crop it and save it to further pass the image through the Convolutional Neural Network.
5. METHODS USED

The following methods are used in the project for identifying brand usage pattern:

1. Tweepy – The Twitter API library in Python which is used to scrape the text, images, location and various other parameters from the JSON file that twitter API provides.
2. OpenCV – The Computer Vision library in Python 2 which is used to obtain boundaries around the brand logos in the scraped images.
3. Convolutional Neural Network: This Neural network has been implemented using Keras which runs Tensorflow in the backend. Keras is a deep learning library in python which helps build a CNN for efficient classification of various brand logos. The functions that have been used to accurately detect brand logo images are as follows:

* Activation function – ReLU
* Loss Function – Categorical\_crossentropy
* Optimization Function – SGD

1. Tableau – This visualization tool helps us in understanding the various trends and usage of people in a area in terms of the sentiment attached to the brand, the number of people using it and the demographics of the brands.
2. RESULTS

The proposed approach of identifying range of pixel in an image for localizing logos showed decent results. Out of 103 Starbucks images examined 67 images contained logos within the bounding box.

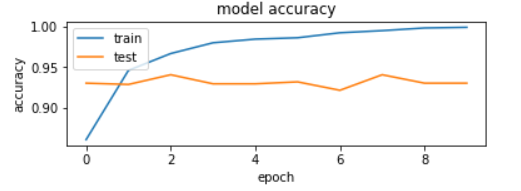




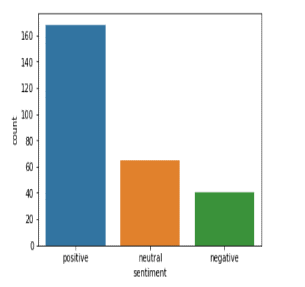
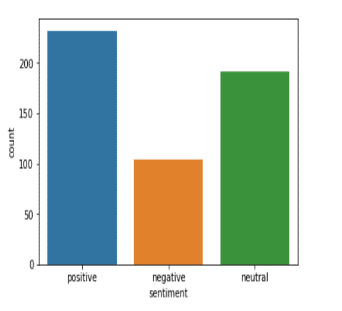
We observed that our localization function doesn’t work in some cases where images are relatively dark, like in the image shown below.



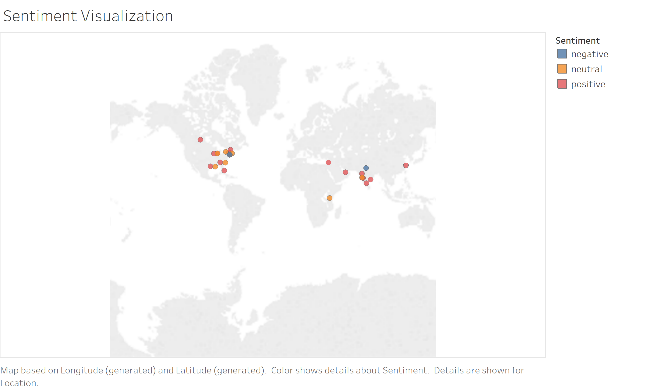
Bounding box if present is then cropped and pass into convolutional neural network which gave us an accuracy of 98% with 10 epochs using stochastic gradient descent as optimizer and binary cross entropy as loss function.



In textual analysis part, We analyzed tweets to get an overall idea about how users feel about brands while posting on social media. From the data we collected Starbucks received higher number of positive tweets, dunkin-donuts received less negative tweets but large number of neutral tweets.



Right graph denotes tweets related to dunkin-donuts and left graphs denotes tweets related to Starbucks.



Above visualization displays sentiment spread across world map for dunkin-donuts.

1. DISCUSSION

The objective of developing this CNN was to classify images by recognizing the brand logos present in the picture and analyze the preferences of people. This analysis would help the brand companies to see the visibility of their products thereby tailoring their marketing and production strategies per use. The results obtained fairly allows a brand company to understand the number of people belonging to a particular location using a particular product which in essence gives them insights into future business growth by ensuring customer satisfaction. The sentiment analysis provides a sentiment score corresponding to a tweet and an image which will help the brand company to see how their products are getting reviewed. The techniques implemented are yielding promising results but one area where more improvement will be required is the Object Localization method as we are defining the range of image pixel values for the OpenCV library to pick the area from the image. The brand companies can use these results to understand the visibility of their products and tailor the marketing and the production strategies to yield better business values.

1. CONCLUSION

The CNN can predict the cropped images and give valuable insights into how people use what kind of brand and what are the sentiments attached to that brand. We believe that this information would be greatly sought after by companies to change their marketing strategies to gain more business from targeted customers and to understand market trends.

REFERENCES

[1] Deep Learning for logo Recognition – <https://arxiv.org/pdf/1701.02620.pdf>

[2] Guide to understanding Convolutional Neural Network –

<https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/>

[3] Logo detection using YOLO - <https://medium.com/@akarshzingade/logo-detection-using-yolov2-8cda5a68740e>

[4] An intuitive explanation of convolutional neural network –

<https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/>