Table of Contents

1 Machine Learning Engineer Nanodegree

1.1 Capstone Proposal

Saravanan Baskaran. 23 Dec 2017

1.2 Proposal

1.2.1 Domain Background

Now a days there are lot of data produced which are visual in nature. Many a times we get pictures and videos shared in social media in addition to text. In order to find the sentiment of a message it becomes necessary to use the visual content in addition to the textual content.This project attempts to build and train a deep learning model based on a research paper that classify an image as expressing positive, negative or neutral sentiment. The proposed model will be trained based on data collected from wild, ie without using any manually labeled data.

This project is based on the paper – Cross-Media Learning for Image Sentiment Analysis in the Wild by Lucia Vadicamo et al. Where they have outlined a method to use cross media learning to train a visual sentiment classifier. This is a first of kind approach where a classifier is trained using data from wild.

This is a very interesting pioneering effort in the field of visual sentiment classification. We try to replicate their results from the image classification part and then try to improve upon it. We will train models using the latest CNN architectures from imagenet contest ILSVRC, namely Inception and Mobilenet and compare their performance (accuracy, number of parameters and prediction speed) to the implementation of original models as described in the research paper. We will also evaluate our models with another human labeled dataset available at DeepSent to check if we get comparable accuracy scores listed in the paper available from the site.

1.2.2 Problem Statement

This is a multi-class classification problem where we build models to classify an image as expressing positive, negative or neutral sentiment. We train the model with the images and labels from B-T4SA dataset. Once trained the model will be able to take an image as input and classifies it into one of the three output labels positive, negative or neutral.

We will use the labeled data from the B-T4SA dataset to train our models. We try to replicate the results from the experiment using the architecture described in the paper. It will be trained with the downloaded dataset and set as baseline models.

We will try to improve upon the result by fine tuning two models based on InceptionV3 and two on Mobilenet architectures trained on imagenet data. We will compare the new models performance (accuracy, prediction time and number of parameters) to the baseline model. All these models will be trained using the same training and validation data and tested against the test set from the downloaded B-T4SA dataset and TDD dataset found at DeepSent.

1.2.3 Datasets and Inputs

As part of the paper, Lucia Vadicamo et al collected a dataset containing 3 million tweets with text and images. The images are labeled based on the content of the text in the tweet using an accurate classifier which works on text. It is then preprocessed to create a labeled image dataset containing balanced number of images(156,862) for each of the three classes. The dataset is available online and can be downloaded from T4SA

We use the B-T4SA dataset to train and test our models. The data set consist of colored images collected from twitter which are labeled as positive, neutral or negative based on the text content by a classifier. The images are preprocessed to remove duplicates and near duplicates. To balance the distribution among the three classes the number of tweet images N(156, 862) from the negative class is selected as maximum for each class. N number of images are picked from the other two classes without replacement to get the final dataset. We finally have 156,862 images for each category with a total of 470,586 images. This is a balanced dataset which contains equal number of samples from each category (positive, negative and neutral).This data is split approximately into 80% for training, 10% for validation and 10% for testing.

1.2.4 Solution Statement

We will follow an approach as described in the T4SA paper to construct our solution. We will use a CNN model to classify the images. Training a deep CNN model from scratch is hard since we don't have sufficient data to learn all filters. To solve this we use a pre trained model that is trained on imagenet data as our feature extractor and then fine tune on it using our data. We use two types of fine tuning. In one instance we lock all layers except FC and only fine tune the FC layers. In another instance we fine tune all layers with the new data.

We will use a Vgg19 model trained on imagenet and finetune all its layers and call it Vgg19-T4SA-A and only the FC layers and call it Vgg19-T4SA-F. We should check that we achieve an accuracy as listed in the table on B-T4SA test set and on DeepSent test data which is called Twitter Testing Data, abbreviated as TTD. TTD had 3 sets of data based on the number of workers agreement to the classification. We have TTD 5 where all the 5 workers agreed on a label and it contains data with high confidence. TTD 4 contains samples where 4 workers agreed on the majority label and similarly TTD 3 contains samples where 3 agreed on the majority label. These models and their accuracy scores are used as baseline to test and compare to other models.

[Tabel1]

Model	TTD 5 agree	TTD 4 agree	TTD 3 agree	B-T4SA Testset
Vgg19-T4SA-F	0.768	0.737	0.715	0.506
Vgg19-T4SA-A	0.785	0.755	0.725	0.513

We finetune two models based on InceptionV3 architecture trained on imagenet with the B-T4SA training dataset. We finetune all layers in model and call in InceptionV3-T4SA-A and only the FC layers in another model and call it InceptionV3-T4SA-F. We finetune two more models based on Mobilenet architecture on all and FC layers and call them Mobilenet-T4SA-A and Mobilenet-T4SA-F respectively

Both model must achieve comparable performance to the Vgg19-T4SA-F and Vgg19-T4SA-A. The accuracy on TTD and B-T4SA should be within 5% of the accuracy of the baseline models. We expect the new models to have a better accuracy and performance than the baseline models.

1.2.5 Benchmark Model

We use the Vgg19-T4SA-F and Vgg19-T4SA-A models that we constructed based on the methods outlined in T4SA paper as benchmark models for score and performance metrics.

We compare the score and performance of our models namely InceptionV3-T4SA-F, InceptionV3-T4SA-A, Mobilenet-T4SA-F, Mobilenet-T4SA-A to that of the baseline models.

We expect our model to have comparable or better score and performance to the base models.

1.2.6 Evaluation Metrics

The models are evaluated on the accuracy score they reach on the TTD 5, TTD 4, TTD 3 and B-T4SA test set. We record the prediction time on TTD5 test set to find the model that takes less time in predicting the result. We also compare the number of parameters of each model which acts as surrogate measure for memory usage of the models.

We expect the new models InceptionV3-T4SA-F, InceptionV3-T4SA-A, Mobilenet-T4SA-F, Mobilenet-T4SA-A to be better than baseline models in terms of accuracy and performance metrics we defined.

1.2.7 Project Design

We use keras running on top of tensorflow to design our models. Keras provides templates for VGG19, InceptionV3 and Mobilenet along with the weights trained on imagenet which we will use as a base to train our models. We will reshape the images from B-T4SA and TDD dataset to 228x228 to match the imagenet image size. Keras also take care of centering the image pixels in all three channel to zero by subtracting the mean pixel values of each of the RGB channels from the input image.

We will use the VGG19 trained on imagenet and finetune all the layers and only fully connected layers on two instances and get Vgg19-T4SA-A and Vgg19-T4SA-F which are set as baseline models. The results from this model will be compared to the results in T4SA to check if our baseline is correct. They should reach an accuracy as given in table Table1 on the TTD and B-T4SA dataset.

Next we finetune a model based on InceptionV3 architecture trained on imagenet. We finetune two instance of this model, where we finetune FC layers for one of the model and all layers in another. We get two models finetuned on two different set of layers InceptionV3-T4SA-F, InceptionV3-T4SA-A which will be compared with the base model. Similarly we will also finetune a set of models Mobilenet-T4SA-F, Mobilenet-T4SA-A based on Mobilenet architecture which will be used in the comparison against the base model.

We graph the results and present the accuracy, prediction time, number of parameters of InceptionV3-T4SA-F and Mobilenet-T4SA-F as a multiple of Vgg19-T4SA-F and InceptionV3-T4SA-A and Mobilenet-T4SA-A as a multiple of Vgg19-T4SA-A model in three different bar charts.

We tabulated the accuracy of Vgg19-T4SA-F, Vgg19-T4SA-A, InceptionV3-T4SA-F, InceptionV3-T4SA-A, Mobilenet-T4SA-F, Mobilenet-T4SA-A on TDD5, TDD4 and TDD3 along with the results from other models listed in the paper at DeepSent

It is expected that the InceptionV3 based model to train faster and more accurate than the baseline model. The Mobilenet based model is expected to perform well in memory utilization and in prediction in terms of running time and may have less accuracy than baseline. We will try to tune the Mobilenet model to give as good accuracy as baseline models.

1.2.8 Future Work

- Improve the accuracy of classifier by using both the image and textual features together and check its effect on the TTD and B-T4SA test set.
- Collect data using amazon mechanical truck on tweets containing both text and images in English and check the performance of the combined model on this human labeled data.
- Create a web application using the top performing model that classifies an image given an url as having positive, negative or neutral sentiment content.
- Do the reverse. Finetune an InceptionV3 model trained on imagenet with augmented version on images from DeepSenti and compare the performance on B-T4SA test set to see if we can get away with less data
- Explore on using the complete data set from T4SA to further improve the scores. Use class weights to account for unbalanced class distribution in the dataset.

1.2.9 Reference

- T4SA – http://www.t4sa.it/
- DeepSent – https://www.cs.rochester.edu/u/qyou/DeepSent/deepsentiment.html
- Cross-Media Learning for Image Sentiment Analysis in the Wild Lucia Vadicamo, Fabio Carrara, Andrea Cimino, Stefano Cresci, Felice Dell'Orletta, Fabrizio Falchi, Maurizio Tesconi link
- Quanzeng You, Jiebo Luo, Hailin Jin and Jianchao Yang, "Robust Image Sentiment Analysis using Progressively Trained and Domain Transferred Deep Networks", the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI), Austin, TX, January 25-30, 2015.link

Author: Saravanan Baskaran

Created: 2017-12-24 Sun 04:49

Validate