Machine Learning Engineer Nanodegree

Capstone Proposal

Ronald Wahome
January 7, 2018

Proposal

Domain Background

Over the last few years, the mainstream media has been increasingly paying more attention to self driving vehicles and there is a lot of speculation as to how soon we can have them on public roads. This mainstream attention only underpins the tremendous time and effort that the research and scientific community is putting towards realizing the dream of self driving cars on the roads today. For self driving cars to be allowed on the roads, they have to achieve the very highest of standards in terms of safety. Like their human counterparts, they have to recognize and obey traffic laws and other road rules and that means reading traffic signs and following directions.

Reading road signs and correctly interpreting them is a topic that has been explored extensively in the machine learning community and in detail with the GTSRB Dataset [1] including a classification challenge held by International Joint Conference on Neural Networks (IJCNN) 2011 [2] that yielded some very promising results. Below is a table showing the top 4 winning results utilizing different machine learning algorithms from the competition.

Rank	Team	Method	Correct recognition rate
1	IDSIA	Committee of CNNs	99.46 %
2	INI	Human Performance	98.84 %
3	sermanet	Multi-Scale CNNs	98.31 %
4	CAOR	Random Forests	96.14 %

As the above results show, Machine Learning was able to achieve very high results in recognizing and interpreting traffic signs. An interesting point to note here is that the best performing algorithm was a CNN(convolutional neural network) which went on to outperform humans in the given task.

Given the above examples, I think it is worth more exploration into the many ways we can improve self driving agents' safety and efficiency as it is only a matter of when not if before we share our roads with self driving cars and possibly replace human drivers altogether. This is an interesting field as it could go on to eliminate the majority of the car accidents and free up a lot of time for humans to use on other tasks.

Problem Statement

The task ahead is one of classifying the different types of road signs in our dataset and matching them to the correct labels so as to achieve an accuracy that is better than actual humans before we can deploy it in a self driving car. I will use my new found knowledge in Machine Learning from the Udacity Machine Learning Nanodegree program to build a Convolutional Neural Network that will be trained on the GTSRB dataset and my goal is to surpass the human recognition accuracy of 98.84%.

Datasets and Inputs

The German Traffic Sign Recognition Benchmark(GTSRB) database [3] was made available by the Institut für Neuroinformatik for the GTSRB single image multi classification challenge and is made up of more than 50,000 images with 39,209 images in the training set and 12,630 images for the test set. I further plan to split this original training set into 80% training set and 20% validation set. The dataset contains a total of 43 classes with unbalanced class frequencies and a wide variation of color, shape and presence of pictograms [6] and as such I will have to use a performance metric that is less immune to misleading classification. The images are used as input and the classes as output. A sample of the images contained in the database is shown below.

The database is made up of large lifelike 3 dimensional RGB channel images of road signs where each real-world sign only appears once in the dataset which is important in helping the model generalize better. The images in the dataset contain one traffic sign each and contain a border of 10% around the actual traffic sign to allow for edge based detection approaches. They vary in size between 15x15 to 250x250 pixels and are stored in a Portable Pixmap format. Additionally, the actual traffic sign is not necessarily centered within the image.

Solution Statement

To achieve an accuracy higher than human performance, I am going to use a convolutional neural network which is a form of supervised machine learning which have shown to perform better [4] than other machine learning algorithms at classifying images when supplied with a large enough dataset because they can discern emerging patterns and formulate relevant tags and categories without being overly computationally expensive. As per the observation above, the images vary in lighting and dimensions so I would like to first resize the images and do a 1-D or Multi-D histogram equalization.

Benchmark Model

As mentioned above, there has been a lot of work carried out on this dataset with the goal of correctly classifying the road signs. My goal is to train a deep CNN that will exceed the human accuracy [5] benchmark of 98.84% which I feel is the minimum acceptable level to deploy in a self driving agent.

Evaluation Metrics

The GTSRB dataset is highly unbalanced and for that I will first try out different methods of balancing the classes before feeding the data to the CNN and evaluating the results with the ‘accuracy metric’. I would like to further explore the results through a confusion matrix which should shed more light on the model’s overall performance. The confusion matrix breaks down predictions into a table showing correct predictions and the type of incorrect predictions made. If that doesn’t work, I can also try the Precision Metric which gives a measure of the classifier’s exactness.

Project Design

1. **Exploratory Data Analysis**

   At this stage, I want to understand the different characteristics of the images in the dataset. I will visualize different images to get an idea of the kind of preprocessing that needs to happen before we can feed the data to the model. Since we are dealing with images, this is a great step to understand the lighting, shadows, partial occlusions, common shapes etc.

2. **Data Preprocessing**

   For deep learning model to achieve good performance, it needs a large amount of training data. I am going to use data augmentation to create more samples of the training data by creating different variations of the original training data. Depending on the observations that we make from the data exploration above, I will preprocess the data accordingly. Some probable ways I could do this are random rotation, flips, inversion etc. Depending on the original size of the images, they may need to be resized to a smaller size (say 20 pixels by 20 pixels) to allow for computational efficiency. At this stage I will also experiment normalizing the data as well as converting the images to grayscale images.

3. **Define Model Architecture, Training and Parameter Tuning**

   I will experiment with different architectures but the key thing to remember is that we are trying to extract features from highly localized regions of the images, so we want to take the original image array with a goal of decreasing its width and height but making it deeper. For this, I am going to use convolutional layers to make the array deeper as it passes through the network and max pooling layers to decrease the resulting spatial dimensions from the convolution layers. So ideally, a max pooling layer will follow every one or two consecutive convolutional layers. A flattening layer will be added after the max pool and convolutional layers to classify the extracted features before a fully connected layer at the end of the network with a Softmax activation to match the features to a label.

Additionally, I will also experiment with different dropout layers which is a regularization technique for reducing overfitting in the deep model. Different variations of the hyperparameters at every layer will also be explored.

4. **More Exploration and Summary**

Before making my final observations, I will test the final model on other real world road signs that are not in the dataset and document its performance on these unseen data.
Finally, I will document the model's performance metrics with comparisons made to the benchmark models. I will explore in detail the model's accomplishments and failures and examine the different ways I could improve on the overall performance.

Reference

[1] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In Proceedings of the IEEE International Joint Conference on Neural Networks, pages 1453–1460. 2011.
[2] http://benchmark.ini.rub.de/?section=gtsrb&subsection=news
[3] http://benchmark.ini.rub.de/?section=gtsrb&subsection=dataset#Downloads
[4] https://ip.cadence.com/uploads/901/cnn_wp-pdf
[5] Human Performance, INI-RTCV , Man vs. computer: Benchmarking machine learning algorithms to traffic sign recognition, Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition, J. Stallkamp, M. Schlipsing, J. Salmen, C. Igel, August 2012, Neural Networks (32), pp. 323-332
[6] http://benchmark.ini.rub.de/?section=gtsrb&subsection=news