MLP and SVM Networks- Comparative analysis

Gayeon Kim
Department of Electrical, Computer and Software Engineering
The university of Auckland
Auckland, New Zealand
gkim902@aucklanduni.ac.nz

This report covers the process of designing machine learning models for a real- world traffic sign recognition application using popular neural networks (multi-layer perceptron) and support vector machine models. The results of numerical experiments and the comparative analysis of these two models are discussed in this report. **(***Abstract***)**

I. Introduction

Machine learning is a technique in which you train the system to solve a problem instead of explicitly programming the rules. The data should be gathered, and a model should be trained to solve the problem using machine learning. It is common to have a dataset with inputs and known outputs. The task is to use this dataset to train a model that predicts the correct outputs based on the inputs. The image below presents the workflow to train a model using supervised learning [1]

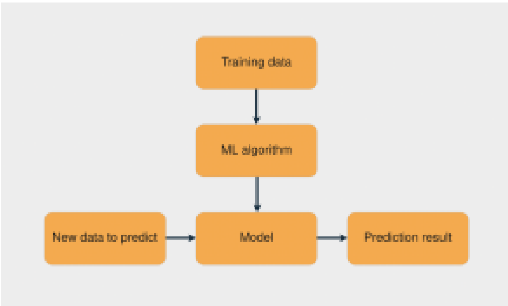


Fig. 1. Workflow to train a model using supervised learning

The learning portion of creating models spawned the development of artificial neural networks. ANNs utilise the hidden layer as a place to store and evaluate how significant one of the inputs is to the output. The hidden layer stores information regarding the input's importance, and it also makes associations between the importance of combinations of inputs. Deep neural nets, then, capitalise on the ANN component. So, the deep net has multiple hidden layers. 'Deep' refers to a model's layers being multiple layers deep. [2]

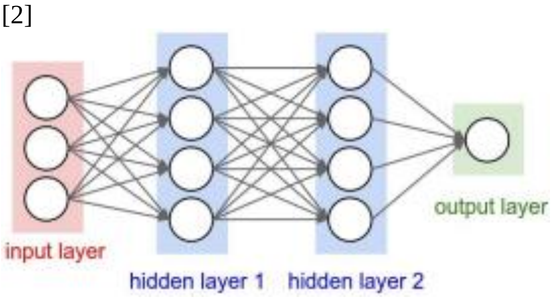


Fig. 2. Multi layers training diagram

MLP and SVM are neural networks that are commonly used AI algorithms. This report will describe how two networks, MLP and SVM model were designed with the given traffic dataset. The report will also summarise and compare these two models using classification report and confusion matrix.

II. Data and Application

A. Dataset used

The dataset used in this project to train the models is the traffic signs classification dataset downloaded from Keras. The dataset contains 43 different types of traffic signs and 73139 images in total.

B. Application of the dataset

The dataset was loaded by looping through the sub directories of the main dataset folder. The images were stored into data array and the name of the folders were stored into the labels array. Then the data were split into a train set and a test set. The train set was used to train the model and the test set was used to test the trained model.

III. MLP (Multi-layer Perceptron)

A fully connected multi-layer neural network is called a Multilayer Perceptron (MLP). The learning algorithm of MLP is based on the minimization of the error function. MLP has 3 types of layers including the input layer, the output layer and the hidden layer. If it has more than 1 hidden layer, it is called a deep ANN. An MLP is a typical example of a feedforward artificial neural network. [3]

A. Design of the model (Parameters used)

The total of 3 hidden layers were used to design the model in this project, since using 3 hidden layers reached the maximum accuracy. When more than 3 hidden layers were used, the accuracy has been decreased due to the overfitting problem of the complex model. It was verified that using 3 hidden layers does not contribute to overfitting problem by plotting loss curve (training loss vs validation loss) and accuracy curve (training accuracy vs validation accuracy). For the loss curve, the validation loss tends to decrease. For the accuracy curve, the gap of the training accuracy and the validation accuracy is small which means no overfitting has occurred.

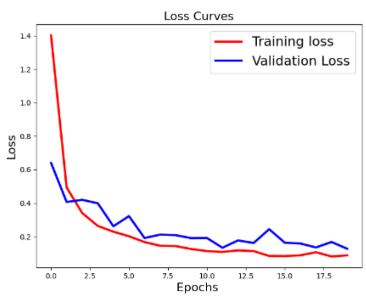


Fig. 3. Loss curves for MLP model

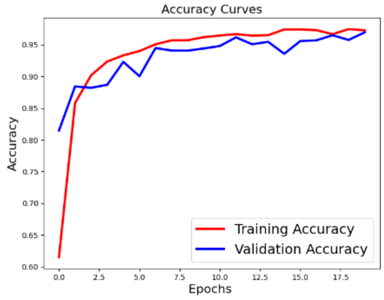


Fig. 4. Accuracy curves for MLP model

The activation functions used for the model are ReLU (Rectified Linear Unit) and SoftMax function. The ReLU function was used for hidden layers and the SoftMax function was used for the last output layer. The reason why ReLU function was used for this model is because it allows better optimization, helps to avoid vanishing gradient problem and provides better and more efficient computation performance compare to other functions. The SoftMax function was used to produce the output values that can be interpreted as probabilities.

For the optimizer, Adam was used. Since the results of the Adam optimizer are generally better than every other optimization algorithm, have faster computation time, and require fewer parameters for tuning [4], Adam optimizer was used.

B. *Data preprocessing*

After loading the dataset, the dataset was split into a test set and a train set using the test size equal to 0.2. This means 20% of the dataset was split into the test set and 80% of the dataset was split into the test set. After splitting the dataset, the size of the images of the train set and the test set were flattened into 1- dimensional array for inputting it to the MLP model.

After changing the dimension of the data, the data was normalised by using StandardScaler function, which is making the values to range between 0 and 1. If the image as it is used and passed through the model without normalisation, the computation of high numeric values may become more complex and could cause problems during modelling. Since normalisation avoids these problems and improves the model's accuracy with the easier computation, the normalisation was applied to the data before training the model.

IV. SVM (SUPPORT VECTOR MACHINE)

After Support Vector Machine (SVM) is a supervised machine learning algorithm used for both classification and regression. The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N- the number of features) that distinctly classifies the data points [5]. SVM model takes kernel function as one of the parameters which is a method used to take data as input and transform it into the required form of processing data.

A. *Design of the model (Parameters used)*

To determine the parameters used for SVM function, the grid search method was used to find the optimal values for each kernel function, C and gamma values. C parameter in SVM is a penalty parameter of the error term. It can be considered as the degree of correct classification. The gamma parameter defines how far the influence of a single training example reaches, with low values meaning 'far' and high values meaning 'close'. The result of the grid search showed that using 'rbf' function for kernel function, 10 for C, 0.1 for gamma produce the best model.

```
The Model is trained well with the given images
{'C': 10, 'gamma': 0.1, 'kernel': 'rbf'}
```

Fig. 5. Result of the grid search for SVM model

B. *Data preprocessing*

After the train- test set split, the images in RGB colour were gray scaled. Gray scaled image has a one-layer image from 0–255-pixel value, whereas the RGB has three different layers. Since the colour increases the complexity of the model, it was gray scaled.

Then, HOG, which is a histogram of oriented gradients which is a feature descriptor that is often used to extract features from image data. HOG feature descriptor was used extract the features and reduce the computational load.

After applying HOG, just like the data pre-processing of the MLP model, the data was normalised using StandardScaler function since it allows the values to range from 0 and 1 which helps to avoid the problem during the modelling and make the computation easier.

V. COMPARATIVE ANALYSIS

A. *Classification report*

The classification report shows the precision, recall, F1 score and accuracy.

Precision is the ability of a classifier not to label an instance positive that is actually negative. For each class, it is defined as the ratio of true positives to the sum of a true positive and false positive. (Accuracy of positive predictions)

Recall is the ability of a classifier to find all positive instances. For each class it is defined as the ratio of true positives to the sum of true positives and false negatives. (Fraction of positives that were correctly identified)

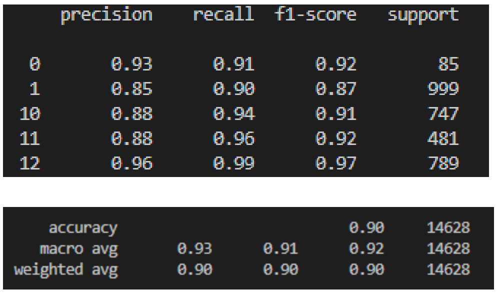
The F1 score is a weighted harmonic mean of precision and recall such that the best score is 1.0 and the worst is 0.0. [6]

The image below shows the classification report of MLP model.

	precision	recall	f1-score	support
0	1.00	0.89	0.94	85
1	0.99	0.96	0.97	999
2	0.94	0.96	0.95	536
3	0.91	0.97	0.94	532
4	0.98	0.96	0.97	738
accuracy			0.97	14628
macro avg	0.97	0.96	0.96	14628
weighted avg	0.97	0.97	0.97	14628

Fig. 6. Classification report for MLP (cropped for report use)

The image below shows the classification report of SVM model.



	precision	recall	f1-score	support
0	0.93	0.91	0.92	85
1	0.85	0.90	0.87	999
10	0.88	0.94	0.91	747
11	0.88	0.96	0.92	481
12	0.96	0.99	0.97	789
accuracy			0.90	14628
macro avg	0.93	0.91	0.92	14628
weighted avg	0.90	0.90	0.90	14628

Fig. 7. Classification report for SVM (cropped for report use)

TABLE I. COMPARE MLP AND SVM

	Type of model	
	MLP_	***SVM***
Precision	0.97	0.93
Recall	0.96	0.91
F1 score	0.96	0.92
Accuracy	0.96	0.90

a. Sample of a Table footnote. (*Table footnote*)

Fig. 7. Table of classification report for both models

All the precision, Recall, F1 score and accuracy are higher for MLP than SVM. This means the accuracy of positive prediction was better for MLP and the number of correction prediction of MLP were more than SVM.

B. *Confusion matrix*

In the confusion matrix, the diagonal elements represent the number of points for which the predicted label is equal to the true label, while off-diagonal elements are those that are mislabeled by the classifier. The higher the diagonal values of the confusion matrix the better, indicating many correct predictions.

The images below show the confusion matrix of both normalised and not normalized of MLP model

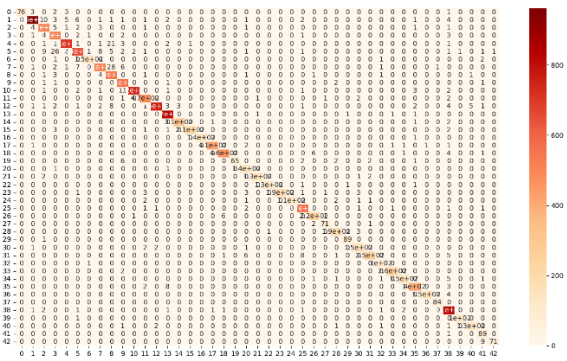


Fig. 8. Confusion matrix for MLP model (not normalised)

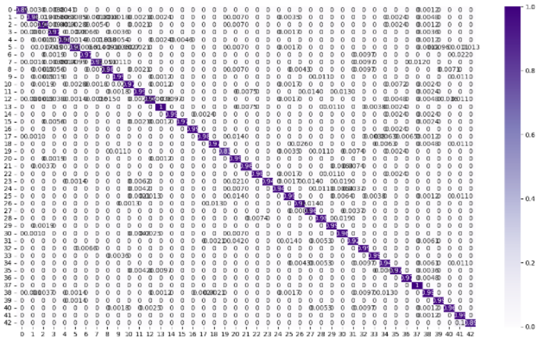


Fig. 9. Confusion matrix for MLP model (normalised)

The images below show the confusion matrix of both normalised and not normalised of SVM model.

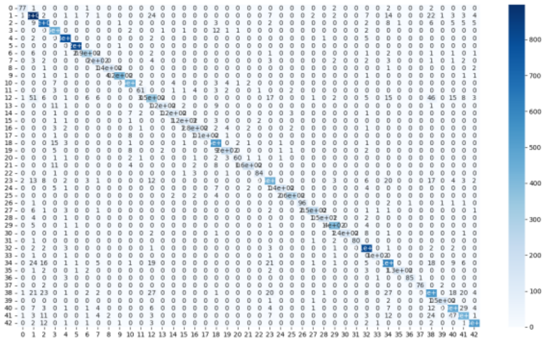


Fig. 10. Confusion matrix for SVM model (not normalised)

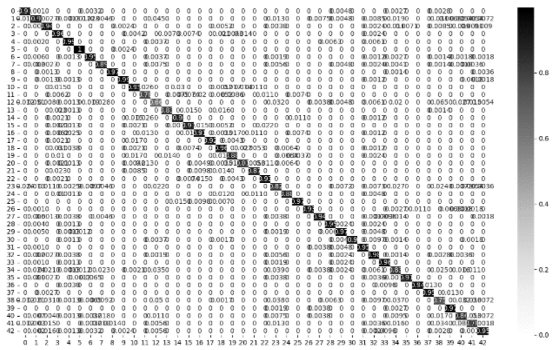


Fig. 11. Confusion matrix for SVM model (normalised)

The numbers of the diagonal elements are higher (have darker colour) for MLP model than SVM model. It could be even more clearly seen in normalised model. More and higher values of off-diagonal elements which are mislabeled data, were presented in SVM model compared to MLP model.

VI. CONCLUSION

The comparative analysis done above shows that both MLP and SVM model performed well on the data with accuracy equal to or above 90%. According to the classification model and confusion matrix analysis, the performance of MLP model was better than SVM model. MLP model had higher precision, Recall, F1 score and accuracy in the classification report. In addition, the confusion matrix presented that MLP model had more predicted labels that are equal to the true labels which means higher accuracy for predicting the data. Generally, SVM model performs better than MLP model, but for this project, I assume SVM model performed worse than MLP model due to the high complexity of SVM model caused by large dataset.

REFERENCES

[1] Déborah Mesquita, Python AI: How to Build a Neural Network & Make Predictions

[2] bmc blogs, What's a Deep Neural Network? Deep Nets Explained, 2020

[3] simplilearn, An Overview on Multilayer Perceptron (MLP), 2022

[4] crown iconAyush Gupta, A Comprehensive Guide on Deep Learning Optimizers, 2021

[5] Rohith Gandh, Support Vector Machine — Introduction to Machine Learning Algorithms, 2018

[6] Shivam Kohli, Understanding a Classification Report For Your Machine Learning Model, 2019