Applying Neural Graph Learning

1st Anand Sharma
MTech CSE (MT19059)
IIIT Delhi
New Delhi, India
anand19059@iiitd.ac.in

2nd Ayan Raha
MTech CSE (MT19032)
IIIT Delhi
New Delhi, India
ayan19032@iiitd.ac.in

I. Problem Statement and Motivation

Neural Network and architectures has given good improvements in field of machine learning .Neural Graph Machines,can combine the power of neural networks and label propagation.Applying Neural Graph Learning on any type of Neural networks such as FFNN,CNN,RNN has performed better on both supervised and semi-supervised learning.This project has shown power of neural graph learning on images a supervised learning task with CNN

II. Literature Review

Classification of images into different classes is old part of machine learning field . Various research has been completed and various are ongoing on it. There are various feature extraction and selection techniques such as Hog,SIFT,SURF,Color Histogram for images that can detect edges,objects,points in concise manner .After applying good feature selection and making hybrid model for detecting features give us good results for classifying images.Machine learning algorithms such as Random Forest, SVM, XGBoost,Decision Tree,Bagging and Boosting helped to classify images after applying suitable feature selection Techniques. This is tedious task and require good knowledge of Computer Vision Techniques. .Development of Convolution Neural Networks has given excellent results on images . Without any feature selection neural network can do proper classification task just by tuning the hyper-parameters and building decent layers in it. Convolutional Neural Networks (CNN) have recently gained popularity at image classification tasks since the initial work by Yann LeCun et al [1] (LeNet) involving the classification of handwritten digits and Alex Krizhevsky's [2] AlexNet which obtained a record performance metrics on the CIFAR dataset [3] at ImageNet classification competition in 2012. The major highlight of CNNs is their ability to extract higher level representation of image features without feature engineering, a manual and expensive process that uses domain knowledge to create features for training in machine learning algorithms CNNs comprise several learnable filters that convolve with input images at specified strides. Another major advantage of CNNs are their ability to reduce the numbers of network parameters and consequently the computational burden, while still attaining increased performances. This report explores effects of modifying the sizes of the convolution filters as well as the overall network architecture. Changes are made to the sequential order of convolution and pooling layers Given a set of N training input-output pairs

$$x_i, y_{i_{n=1}}^{N}$$

,such neural networks are often trained by performing maximum likelihood learning, that is, tuning their parameters so that the networks ,outputs are close to the ground truth under some criterion,

$$CNN(\theta) = \sum_{n} c(g_{\theta}(x_n), y_n) \tag{1}$$

where g represent the overall mapping, parameterized by , and c (·) denotes a loss function such as l-2 for regression or cross entropy for classification

III. Dataset Details

The CIFAR-10 dataset consists of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images. The dataset is divided into five training batches and one test batch, each with 10000 images. The test batch contains exactly 1000 randomly-selected images from each class. The training batches contain the remaining images in random order, but some training batches may contain more images from one class than another. Between them, the training batches contain exactly 5000 images from each class.The classes are airplane,automobile,bird,cat,deer,dog,frog,horse,ship,truck.The classes are completely mutually exclusive. There is no overlap between automobiles and trucks. ”Automobile” includes sedans, SUVs, things of that sort. ”Truck” includes only big trucks. Neither includes pickup trucks. MNIST-10 Dataset is used to validate Neural Graph Learning and improvised the accuracy . These 2 datasets are used to validate the objectives

IV. Proposed Architecture

Normally Neural Networks are trained on same types of class in training instance. if it is trained on dogs images to classify it will only correctly classify dogs and not able to classify cats Such shortcoming of neural network training can be rectified by biasing the network using prior knowledge about the relationship between instances in the dataset. If dogs

Identify applicable funding agency here. If none, delete this.

and cat have same label then we encourage the neighbouring data points to have same hidden representation . This can be done by embedding the graphs of neighbours images with original dataset combined to the neural networks. Tensorflow

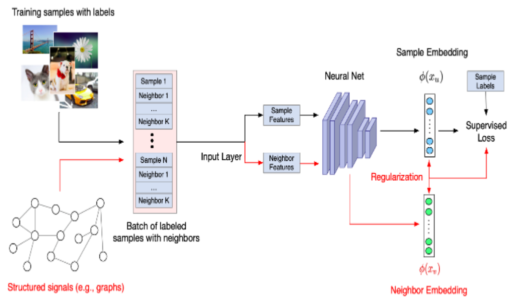

Fig. 1. Workflow of Neural Structural Learning.

framework of Neural Structured Learning (NSL) focuses on training deep neural networks by leveraging structured signals (when available) along with feature inputs. As introduced by Bui et al. [**?**], these structured signals are used to regularize the training of a neural network, forcing the model to learn accurate predictions (by minimizing supervised loss), while at the same time maintaining the input structural similarity (by minimizing the neighbor loss, see the figure below). This technique is generic and can be applied on arbitrary neural architectures (such as Feed-forward NNs, Convolutional NNs and Recurrent NNs). OPTIMIZE:

$$loss = \sum_{i=1}^{B} c(g_\theta(x_n), y_n) + \alpha \sum_{i=1}^{B} W_{ij}.D(h_\theta(x_i), h_\theta(x_j)) \quad (2)$$

loss=Supervised loss+Neighbour loss

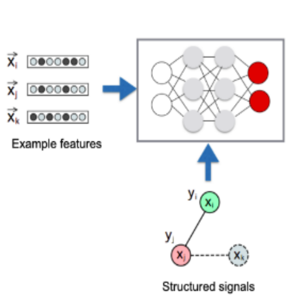

Fig. 2. Features input and graphs input to neural network

Out proposed architecture will take care of Convolutional Neural Network with embeddings of graph by creating adversarial learning concept which will allow the model to train more accurately. This additional adversarial loss will make neural networks more complex and more time to train model.

V. Visualizations

Classes are balanced and all 10 classes has 5000 training images each .Converted RGB images into GRAY image for Classification task and normalized the pixels . Fig3 shows the RGB image of classes of dataset and fig 4 shows the gray images.

Fig. 3. Different RGB images of 10 classes

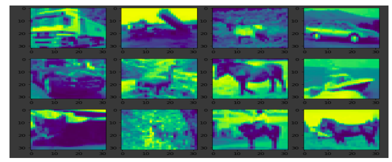

Fig. 4. Different GRAY images of 10 classes

VI. Results

- In Fig7 Graphs showing Accuracy of Random Forest using Different Features selection Technique
- In fig7 A table is represented showing the results of CNN model and CNN model with Neural Graph Machines at different epochs with precision ,recall and fscore. Training accuracy and Testing accuracy are also shown in this figure table
- in Fig6 No of epochs vs loss and epochs vs accuracy plots are shown for Training and Testing accuracy . Similarly CNN with NGM plots are shown for training and testing loss and accuracy at different epochs.

VII. Analysis of Results

- From the fig7 we can see that sift,surf and color histogram has given bad classification with lower accuracy as compared to normal images . HOG feature selection Technique has given good accuracy
- Random Forest classifiers are used as it has given better accuracy when compared to others classifiers

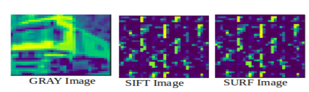

Fig. 5. Feature selection based on SIFT and SURF

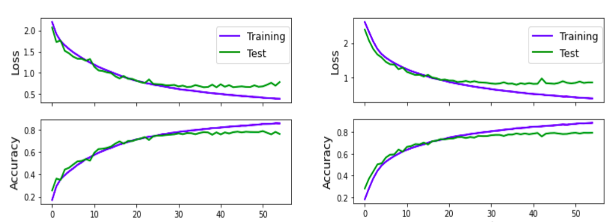

Fig. 6. Training and Testing loss and accuracy for diffrent epochs

Classifiers	Feature Selection	Testing Accuracy
Random Forest	Color Histogram	10
Random Forest	HOG	50.93
Random Forest	SIFT	22.59
Random Forest	SURF	23.13
Random Forest	NO	47.42

Results:

With the MNIST data and an FFNN, the accuracy results I got with and without the Neural graph machine is given here

No of epochs	Without graph function	With graph function
5	96.05	96.72%
10	97.01	97.45%

Classifiers	Epochs	Training Accuracy	Testing Accuracy	Precision	Recall	F-score
CNN Block1	41	100	65.850	0.6345	0.6689	0.6430
CNN Block1 +NGM	41	98.24	70.090	0.7014	0.6723	0.6820
CNN Block2	34	96.69	71.80	0.7010	0.6881	0.6920
CNN Block2+NGM	32	96.99	72.280	0.6810	0.655	0.655
CNN Block3	32	95.11	68.750	0.6678	0.6789	0.654
CNN Block3 +NGM	34	99.22	70.570	0.7014	0.6723	0.6820
CNN Block3 +Dropout9	55	86.01	79.060	0.7723	0.7644	0.7618
CNN Block3 +Dropout9+NGM	55	88.50	80.15	0.7974	0.7938	0.805

Fig. 7. RESULTS TABLE

- Convolutional Neural Network has performed excellent without any features selection techniques
- CNN Block1 has lower accuracy as compared to CNN Block2 and CNN Block3 which has almost same validation accuracy
- With increase in Training accuracy Validation accuracy increases up-to certain epochs after that it remain almost constant or slight increase
- With the use of Neural Graph Machines Validation accuracy has increased up-to 5 percent in CNN Block1 and 1-2 percent in CNN Block 2and 3.
- Adding Dropout in CNN model and CNN Block3 model has increased the validation accuracy up to 9-10 percent ie 79 percent
- In making adversarial Network value of alpha is choose to be 0.2 as it gives us good results

VIII. Inferences and Conclusion from results

- Applying Convolutional Neural Network produce excellent results in classifying images on cifar-10 dataset
- We have tried 3 Differnt architecture of CNN . with kernel size of (3,3) and MaxPooling of(2,2) we have added more convolutional layers in the model . With 4 convolutions and maxpooling is decent for this classification
- Adding Dropout layers in the model will stop the model to overfit . With less training accuracy Dropout increases Validation accuracy very much
- Reprocessing Data and normalizing only shows slight variations in validation accuracy and not contribute much
- Adding more Dense layers and more convolutions only over-fits the data and not increase the validation score
- Applying Neural Graph Learning on CNN models will make best use of Neighbours data and helps to regularize the hidden layers and increases the validation score
- The training and validation loss tells us that our model has good fitted the data and not done over-fitting and under-fitting
- Using Neural Graph Learning makes our model Robust against adversarial perturbations designed for misleading a model's prediction or classification and Higher accuracy reached
- Neural Graph Learning method performs well on semi-supervised learning as less label data is required for training the model .
- NSL allows the network to train using labeled data as in the supervised setting, and at the same time drives the network to learn similar hidden representations for the "neighboring samples" that may or may not have labels

IX. Individual contributions of each group partner

A. *Anand Sharma*

- Done all the operations specified on cifar-10 dataset
- Applied various feature selection Technique such as SIFT,SURF,HOG,Color histogram and applied Random Forest and SVM model for classification
- Applied 3 different architecture of CNN models and do some hyperparameter tuning
- Applied Neural Graph Learning ON CNN using Adversarial Network and improved the accuracy of CNN model
- Calculated the Precision,recall,fscore,training accuracy testing accuracy for all the models applied

B. *Ayan Raha*

- Applied all the operations on MNIST-10 Dataset
- made a Feed forward Neural Network to classify digits on MNIST-10 dataset
- Constructed graph by calculating neighbours and embedding the graph to the FFNN to do Neural Graph Leaning .

References

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradientbased learning applied to document recognition," Proc.IEEE, vol. 86, no. 11, pp. 2278–2323, 1998.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks."

[3] "CIFAR-10 and CIFAR-100 datasets." [Online]. Available: https://www.cs.toronto.edu/ kriz/cifar.html. [Accessed: 20-Oct-2019].

[4] Thang D. Bui, Sujith Ravi, and Vivek Ramavajjala ,"Neural Graph Learning: Training Neural Networks Using Graphs ," WSDM 2018, February 5–9, 2018, Marina Del Rey, CA, USA ACM ISBN 978-1-4503-5581-0/18/02.