Incremental Task learning for semantic segmentation

Mudassar Hussain
dept. Data Science Engineering
Politecnico di torino
Turin, Italy
mudassar.hussain@studenti.polito.it

***Abstract*—Training neural networks in an incremental fashion to recognize additional classes while preserving the previously known classes is a challenging problem known as *catastrophic forgetting*. Moreover, in case of semantic segmentation, we have to deal with the issue of changing background known as *background-shift*[1]. Most of the existing class-incremental learning approaches either store data or use generative replay, many of which have drawbacks. On the other side, *'rehearsal-free'* methods such as parameter regularization do not reach high performance. Here, we propose another strategy for semantic segmentation in an incremental setting : using an ensemble of classifiers to learn additional classes at each learning step. Observing that the output of the background class tends to be the top most node in the network architecture given that the network is trained with cross entropy loss function, background shift can be solved by averaging and regularizing the background output from each classifier in the ensemble.**

I. Introduction

Neural networks have become really good at supervised learning, but only when the whole data set is available at once. On the other hand, Learning classes incrementally turns out to be very problematic, The network becomes biased towards recently added classes and tends to forget the previous ones, This phenomena is called *'catastrophic forgetting'*. Successful methods usually store and replay data from previous data during training, which can be computationally expensive or even not feasible.
Here we propose that, for each incremental step, a new classifier can be trained and added to the ensemble. The main benefit is that it turns a difficult class learning problem into an easier task incremental problem. Avoiding catastrophic forgetting in this setting is trivial since we are not training the same classifier over and over again. Later we demonstrate, how we solve back ground shift by averaging and regularizing the background output.

II. Problem Overview

In continual learning setting, the network does not have access to all the data set rather the data is fed in a sequential manner. Continual learning is further categorized into three different types. Incremental task learning where the algorithm learns distinct task over time. Domain learning (adaptation) where algorithm learns same task with shifting context. Lastly, class incremental learning where method learns to distinguish between a growing set of classes. Here, we convert a class incremental problem into task incremental one.

A. Class incremental learning

As alluded earlier, in class incremental learning, the classifier learns to accommodate additional classes over time. A very often used protocol is to split the data set into 'episodes' where each episode contains a different set of classes. A key insight here is that the data from previous task is no longer available after transitioning to the next task.

B. Task incremental learning

In task incremental setting, a set of classifiers can be trained distinctly. This approach requires that the knowledge of the kind of task is known in advance. This approach is considered to be easier compared to class incremental learning. Hence it would be advantageous if we can convert a former problem into later.

III. Proposed strategy: Ensemble classifiers

A. General intuition

The classical approach in deep learning is to learn a discriminating classifier which learns the conditional probability $p(y|x)$ e.g a convolution network with a softmax output and a cross entropy loss. This approach works well when the whole data set is available. In incremental learning process, the softmax will tend to be biased towards recently added classes.Thus this approach will not work. Hence dominant trend in continual learning is to alleviate this affect of catastrophic forgetting by introducing new techniques.
Here we propose one such technique. Rather then learning the incremented classes over the same classifier. We suggest having a pool of classifiers for each episode and then aggregating this pool into a single ensemble. Each classifier is trained over its own set of classes and hence catastrophic forgetting is out of question. Moreover, background shift can be avoided by considering the average of the background output from all the classifiers. We introduce a noise control factor (alpha) to regularize, hence toning down the probability of the output of the background class.

B. Switching from class incremental to task incremental

Another benefit of using an ensemble classifier is that challenging class incremental problem turns into a task incremental problem where each 'task' can be thought of as learning a new set of classes. Since each classifier has its own distinct data set, every classifier is trained separately from the others. Hence the training process can be parallelized or a classifier can be

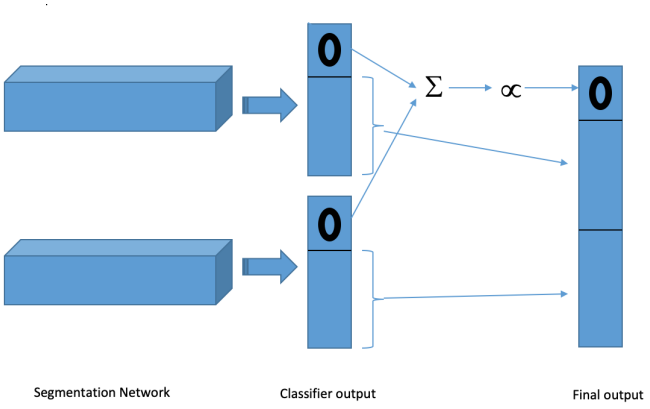

Fig. 1. Ensemble architecture

added to the ensemble in future without having to re-train the previous classifiers in the ensemble.
Usually task incremental problems are dealt by proposing a multi head classifier in the architecture with each head resopnsible for its own task, where as class incremental problems have a single head classifier. Figure 1. shows the architecture of the ensemble model where a multihead classifier is combined into a single head classifier. In this way classifiers are trained separately but are able to respond collectively during inference.

C. Understanding neuron selectivity

CNN architectures' performance is impressive yet it remains a black box model. This is due to the lack of understanding in the way these representations are build and the way the results are internally organised. [2] shows insight on indexing neuron's class selectivity. Class selectivity can be defined as a property of a neuron that can help to establish its discriminative power for one specific class or can allow to cluster neurons according to the ontological properties of their class labels. They discovered that the class selective neurons belong to the deeper layer. More succinctly, we are interested in the class selectivity indexes of the final layer. It can be shown that provided we are using standard cross entropy loss for back propagation in the network, the order of the classes being selected in the final layer is numerical. Since the background class is assigned the label 0, it is collected by the top most neuron in each classifier. Hence by combining and averaging, we can merge the background response from each classifier.

IV. Semantic Segmentation

Semantic segmentation requires rich feature extraction as the prediction has to be performed on pixel level. Moreover, advanced application such as autonomous cars require inference to be performed in real time. Keeping that in mind, we discus models that have been proposed uptil now in the next session. Then we describe BiseNet [3], architecture tailored for semantic segmentation in real-time.

A. Related Work

Most of the algorithms considered state-of-the-art are based on fully convolutional network (FCN). Long et al. [4] proposed one of the first deep learning works for semantic image segmentation, using a fully convolutional network (FCN). FCN, as the name suggests, is composed of only convolution layers for both feature extraction and classification tasks. These algorithms are popular yet have certain limitations, mainly not being fast enough for real time inference.
Ronneberger et al. [5] proposed U-NET for segmenting biological microscopic images. It is a modified version of FCN architecture encompassing encoder-decoder setting. The architecture tends to fuse the hierarchical features of the backbone network. With improved spatial resolution, the model is still computationaly expensive for real time processing.
Zhao et al. [6] proposed Pyramid Scene Parsing Network (PSPN), a multi-scale network to improve global context representation learning of an image. It uses a pyramid pooling module to distinguish pat- terns of different receptive fields from input and the outputs of these pyramid layers are concatenated to capture a rich global information.
Deeplab, developed by Chen et al., [7] is among state- of-the-art image segmentation algorithms. The model highlights three key features. First is the use of dilated convolution to address the issue of reduced feature resolution caused by sequential use of max-pooling and striding. Second is Atrous Spatial Pyramid Pooling (ASPP), which captures objects as well as image context at multiple scales to robustly segment objects at multiple scales. Third is the use of fully connected Conditional Random Fields (CRF) as a post processing step to improve the predictions near the borders of an object. Deeplab achieves good performances in reasonable time yet it is not suitable for real time inference either.

B. BiseNet

Bilateral Segmentation Network (BiseNet) is an architecture tailored for real time inference. It is a light weight model and is able to overcome lack of spatial details in feature representations and provides sizable receptive field. The arthors argued that compromising accuracy to speed is inferior in practice as in previous semantic segmentation approaches. The model comprises of two paths: First is a 3-layered spatial path to preserve the spatial details of the original input image and encode affluent spatial information. Second path is named as Context path and it is designed to provide a sufficient receptive field to the classifier. It is critical for each output pixel to have a large receptive field, such that no important information is left out when making the prediction. Techniques like PSPP [6] and ASPP [7] use pyramid pooling to extract the enlarged receptive field, but these methods are computationally expensive. The context path in BiSeNet uses a lightweight backbone model to down sample and extract high level features with large receptive fields. A global average pooling (GAP) is applied

on the tail of these features which encodes the maximum receptive field with global context information. Finally, the features of the backbone model and up-sampled GAP results are combined to form the output of the context path. In the Context Path, a specific Attention refinement Module (ARM) is used to refine the features of each stage. It employs global average pooling to capture global context and computes an attention vector to guide the feature learning. It demands negligible computation cost, and it can refine the output feature of each stage in the Context Path.

The output features for spatial path and context path are of different nature as one encodes low level, rich detailed information and the other provides high level context information, respectively. BiSeNet includes a Feature fusion module (FFM) which uses batch normalization on concatenated features to scale them appropriately.

V. Experiments

A. Pascal-VOC 2012

Pascal-VOC 2012 [8] includes 20 classes and an additional background class. There are mainly two incremental setting when it comes to sampling images to build the datasets. In *disjoint* setup each step contains unique set of classes with pixels from the current step or in previous ones. However, in *overlapped* setting, pixels can belong to any class, but only the pixels of novel class are annotated. Difference in the two cases that in the later setting, images may contain pixels from future class annotated as background. This setting is more realistic and better choice for performing experiments. Hence we adopted this approach.

We perform two types of experiments here: (15-5) introducing 5 classes sequentially, (15-5s) where we add each class one by one on top of previous classes.

miou value is shown in figure 2. after training both the classifiers separately. The final results are as follows:

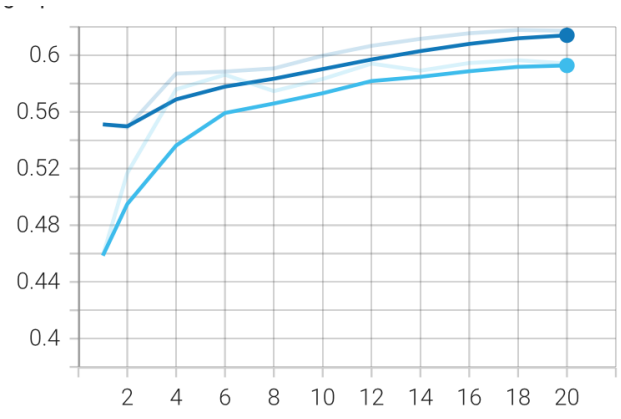

Fig. 2. miou value

VI. Conclusion

We studied the continual learning problem in general. Then we focused on class incremental learning in the context of semantic segmentation. We highlighted the phenomena of catastrophic forgetting as well as background shift . We proposed an ensemble of classifier which alleviates catastrophic forgetting. Mean while by averaging the output of the background class from each classifier and regularizing it, we resolved the issue of background shift.

Metric	Precision	miou	Loss
Measure	0.802	0.475	0.5284

Fig. 3. miou value

References

[1] Fabio Cermelli, Massimiliano Mancini , Samuel R Bulo, Elissa Ricci, Barbara Caputo. Modeling the Background for Incremental Learning in Semantic Segmentation (2020).

[2] Ivet Rafegasa, Maria Vanrella, Luis A. Alexandreb, Guillem Arias. Understanding Trained CNNs by Indexing Neuron Selectivity (2019).

[3] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, Nong Sang. BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation, (2018).

[4] Jonathan Long, Evan Shelhamer, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. Conference on Computer Vision and Pattern Recognition (2015).

[5] Olaf Ronneberger, Philipp Fischer, Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. Medical Image Computing and Computer Assisted Intervention (2015).

[6] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, Jiaya Jia. Pyramid Scene Parsing Network. Conference on Computer Vision and Pattern Recognition (2017).

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, Alan L. Yuille. DeepLab: Semantic Image Segmentation with Deep

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.