AI CS461 Assignment 1

Saif Ul Islam

February 12, 2020

Abstract

This assignment deals with trying to understand the technical background and context within the paradigms of Visual Recognition in the world of Deep Learning. The following contents intend to take a descriptive look at *DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition* and break apart into different sections

Contents

1 Introduction And Overview **2**
- 1.1 Conversion and Pre-processing . . . 2
- 1.2 Relevancy To Subjects . . . 2
- 1.3 Approach Used . . . 2

2 Training And Statistics **3**

3 Algorithm Implementation and Development **4**

4 Computational Results **4**

5 Summary and Conclusions **4**

1 Introduction And Overview

The *DeCaf* paper evaluates performance of a trained model, with features extracted from a fixed set of objects related to fashion items, with the intention of categorizing them into different sections, and attempting to understand whether it can be used generally as a multitude of other tasks, with intentions of seeing how well or unwell the model performs with similar problems, but with different types of data.

DeCaf deals primarily with perpetual learning, with an emphasis on producing a salient, multi layered sectioned version of a model, since it has long been the center of the debate of many past research papers. Recent results have often shown that deeply layered unsupervised models have out performed state-of-the-art shallow in detection based models.

1.1 Conversion and Pre-processing

A model is first trained with a state-of-the-art model that works on the fashion data-set, under a fully supervised setting, from which certain key features are extracted. These features are than saved and passed down to the model, pre-trained on these features, suited for the novel generic tasks, and their results are then judged quantitatively and qualitatively through the process of visualizations.

Two primary questions arise,

1. Do features that are extracted from the prior CNN model generalize well to act as pretrained features on the new, novel task based model? What features effect the output the most?
2. Does increasing or decreasing the layers effect the overall result produced from the newly trained model?

1.2 Relevancy To Subjects

The paper is relevant to computer vision and related tasks, by taking a look at a general model that aims to provider a broader selection of objects from a single given model.

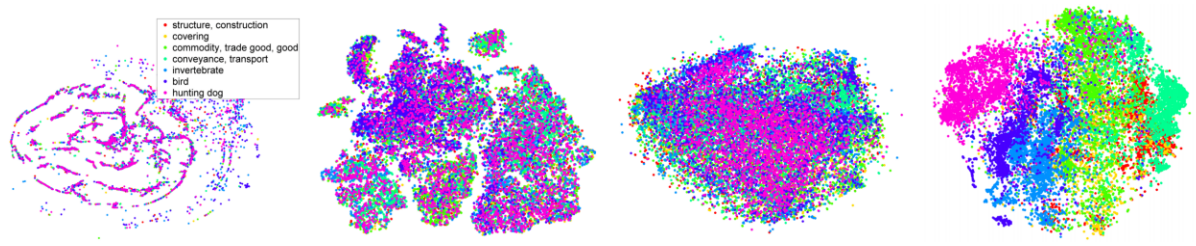
Learning from related tasks also has a long history in machine learning, for optimizing representations from related tasks. Key objects that captures the object category white discarding irrelevant noise such as illumination is also a focus for *DeCaf*, which it aims to remove by trying to perform well on generic tasks.

1.3 Approach Used

The features are generated to visualize and gain insight into the semantic capacity of DeCaf with other features, such as GIST features, and LLC features.

These features are visualized as follows,

1. t-SNE algorithm to find a 2-dimensional embedding of high-dimensional space
2. Plotting according to semantic value
3. Cross check if the data-set has any effect on the results



(a) LLC (b) GIST (c) $DeCAF_1$ (d) $DeCAF_6$

Figure 1: This figure shows several t-SNE feature visualizations on the ILSVRC-2012 validation set. (a) LLC , (b) GIST, and features derived from our CNN: (c) DeCAF1, the first pooling layer, and (d) DeCAF6, the second to last hidden layer (best viewed in color

2 Training And Statistics

The training involves taking the activation of the nth layer of the DeCaf model, in a way to extract the features generated at that level, before the results are propagated to the final network. The images are taken in as input with the following dimensions on the center,

1. 224 x 224 crop
2. 256 x 256 resized

Using the *Deform-able Parts Descriptor* (DPD), by (Zhang et al., 2013), the following below results were obtained,

Method	Accuracy
$DeCAF_6$	58.75
DPD + $DeCAF_6$	**64.96**
DPD (Zhang et al., 2013)	50.98
POOF (Berg & Belhumeur, 2013)	56.78

Table 2. Accuracy on the Caltech-UCSD bird dataset.

Figure 2: Accuracy on the Caltech-UCSD bird dataset.

3 Algorithm Implementation and Development

The algorithm is simply a very deep convolutional neural network (CNN) publicly available for viewing with the respective network parameters. It was trained heavily on supervised related data, particularly related to fashion items, and then exposed to extracting the features from that trained model, and applying the selected features to a model that would train on some general or related task.

---

4 Computational Results

	$DeCAF_5$	$DeCAF_6$	$DeCAF_7$
LogReg	63.29 ± 6.6	84.30 ± 1.6	84.87 ± 0.6
LogReg with Dropout	-	86.08 ± 0.8	85.68 ± 0.6
SVM	77.12 ± 1.1	84.77 ± 1.2	83.24 ± 1.2
SVM with Dropout	-	$\mathbf{86.91 \pm 0.7}$	85.51 ± 0.9
Yang et al. (2009)		84.3	
Jarrett et al. (2009)		65.5	

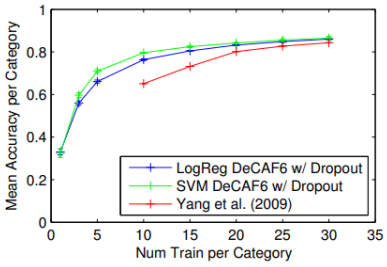


Figure 3: Left: average accuracy per class on Caltech-101 with 30 training samples per class across three hidden layers of the network and two classifiers. Our result from the training protocol/classifier combination with the best validation accuracy – SVM with Layer 6 (+ dropout) features – is shown in bold. Right: average accuracy per class on Caltech-101 at varying training set sizes.

---

5 Summary and Conclusions

The end results show us that deep features applied to semi-supervised related tasks often demonstrates using a large dataset to learn features, with just about enough power to perform and give the proper representations for edges, object detection, feature extraction, noise filtering, can often result in state-of-the-art approaches with multi-kernel based solutions as well.