Critical Review: *Deep Clustering for Unsupervised Learning of Visual Features*

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. "Deep Clustering for Unsupervised Learning of Visual Features." arXiv:1807.05520 [cs.CV]. Proc. ECCV (2018).

Group 13: Suraj Shah
Mentors: Dr. Mathias Unberath, Dr. Jim Fackler, and Dr. Jules Bergmann
April 23rd, 2019

Project Overview

Pneumonia continues to be a disease that has long-lasting and detrimental effects on the U.S. healthcare system. Nearly one million diagnoses are made per annum, with more than 50,000 individuals dying from the disease each year[2]. Chest X-rays are the empirical standard for diagnosing pneumonia, however, the diagnosis for each individual patient is still heavily reliant on the knowledge and practice of radiologists. Furthermore, complications are much more exacerbated when patients are diagnosed with pneumonia in the ICU (Intensive Care Unit) or PICU (Pediatric Intensive Care Unit). Mechanical ventilation (and intubation) is a critical, life-sustaining ICU therapy. "More than half of the patients in the ICU are ventilated the first 24 hours after ICU admission."[3] Ventilation greatly increases the risk for acquiring pneumonia, known as VAP (Ventilator Associated Pneumonia); 10-20% of ICU patients are diagnosed with VAP annually and acquiring VAP increases the risk of mortality by 30%. Thus, it is imperative to build a system that is able to detect pneumonia for ventilated patients in the ICU that does not solely rely on radiologist overview and a system that we are able to diagnose the risk of pneumonia early on. My project aims to prepare an automatic classifier to be used in the ICU to monitor VAP.

Paper Summary and Background

This paper was submitted to ECCV (European Conference on Computer Vision) 2018 in aims of presenting DeepCluster, a novel clustering method that performs two objectives : 1) learns the parameters of a neural network and 2) clusters the assignments of the resulting features. DeepCluster was produced to combat the saturation in performance of pre-trained convolutional neural networks that are trained on a limited set of images. Thus, the DeepCluster objective is to produce a machine learning method that "can be trained on internet-scale datasets with no supervision."[1] The paper was submitted by members of the Facebook AI Research team, as the Facebook organization continues to be a leader in computer vision and applying techniques for classification. Given that this paper was published in late 2018, there is still advancements to be paved to apply the clustering techniques in both academic and industrial settings. However, the authors provide a GitHub open-source annotation for users to run on their own datasets. This will hopefully provide other researchers ways to test model(s) for themselves and prove the robustness of the group's techniques. For my maximum deliverable, I plan on applying DeepCluster to my own dataset.

Selection Motivation

This paper aims to take a novel approach in how image classification is treated with the limited amount of standardization that is present with image annotations. For example, the most highly-used convolutional neural networks are trained on ImageNet, which contains 1 million images to cover most types of classification (up to 1000 labels). To increase performance, one could feasibly increase the size of the dataset by a factor of 10-100x, but that would concern much more manual annotation, placing a burden on human effort and is much more extensive than what is currently available in the data science community. Thus, it is imperative to produce a model that can create generalizations for visual features to apply to any large-scale dataset that does not require supervision. This is critical to understanding of classification for my project because of the complexities of X-ray data when applied to pneumonia diagnoses with different demographics (specifically ages), ventilations, and machines. One goal of our collaborators is to be able to apply the data in the PICU – however, most of the publicly available data has only adult chest X-ray data. Thus, applying unsupervised clustering methods might unlock generalization of pneumonia features in my project that previously was not available with supervised learning.

Technical Methods

Overview

Mathilde et. al. identifies that that convolutional neural networks (CNNs) are the best (and most popular choice) for "mapping raw images to a vector space of fixed dimensionality."[1] However, they suggest when parameters θ are sampled from Gaussian distributions, the CNN mapping does not produce proficient features because of the reliance of a strong prior on an input signal. The authors aim to build off the existing CNN work to cluster the CNN's outputs using 'pseudo-labels' to learn the true features of the input and group them without labels[1]. The approach used in clustering is *k-means*, a commonly accepted practice in clustering approaches which in the case of the CNN chosen by the authors, takes the set of features produced by the CNN, and clusters them into k distinct groups based on a certain geometric criterion. The actual geometric criterion is more or less irrelevant; what is of significance is the assignments based on proximity to the centroid of the criterion, assigning the features into 'pseudo-labels'[1]. Once these labels are assigned through clustering, the DeepCluster is trained by switching between producing the pseudo-labels and updating the parameters of the CNN by predicting the pseudo-labels.

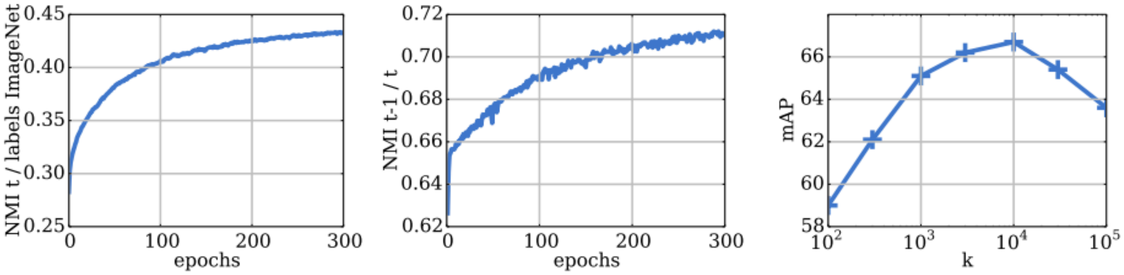
Implementation and Architecture Design

As mentioned before, the group uses previously trained CNN architectures to start with. Specifically, they use AlexNet and VGG-16. The difference between using the two models is based on the specifics of the architectures – AlexNet has 5 convolutional layers while VGG-16 has 16 convolutional layers. Thus, it is a tradeoff between intensiveness of the training experiments when applying techniques on large-scale datasets over n epochs. Because clustering does not work well with color channels, the team preprocessed the data by "applying a fixed linear transformation based on Sobel filters to remove color and increase local contrast."[1] The main training data used here is the standard ImageNet framework which has 1.3M images divided into 1000 classes.

In terms of parameter selection, Mathilde et. al. trains the network with fairly standard chosen parameters: dropout, a constant step size, an L2 penalization of the weights θ, and a momentum of 0.9. After the CNN is trained and produces features, the features are PCA-reduced to 256 dimensions (still relatively hefty in my opinion), whitened and L2-normalized. The authors chose to update the clusters after each epoch of training (vs. updating clusters after *m* epochs), which proved to be pretty computationally intensive, as they trained the model for 500 epochs which took 12 days on a Pascal P100 GPU for AlexNet. However, results as shown in following sections proved to be optimal.

Experiments and Results

When actually measuring the effective of clustering on the processing of features and pseudo-labels, Mathilde et. al. ran three different experiments as seen in the image below.

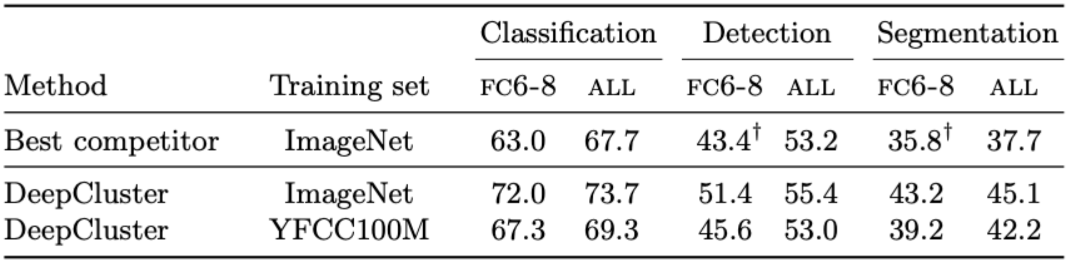


(a) Clustering quality (b) Cluster reassignment (c) Influence of k

NMI (Normalized Mutual Information) is the parameter that is measured in figures *a* and *b* above. NMI measures the information shared between two different assignments, i.e. a measure of independence. If NMI is 0, the two assignments are considered to be completely independent, and vice versa. Figure a) uses NMI to track the dependence between clusters and labels; given that this increases over the number of epochs, the features are capturing the information related to object classes. Figure b) measures the number of assignments between epochs, and since the NMI is increasing, this means the actual clusters predicted by *k-means* are stabilizing over time. Lastly, we see the impact of the *k* clusters on model performance; this peaks at 10,000 clusters, which is surprising given there are only 1000 image classifications in ImageNet, however, this probably means that over-segmentation is more important with clustering performance.

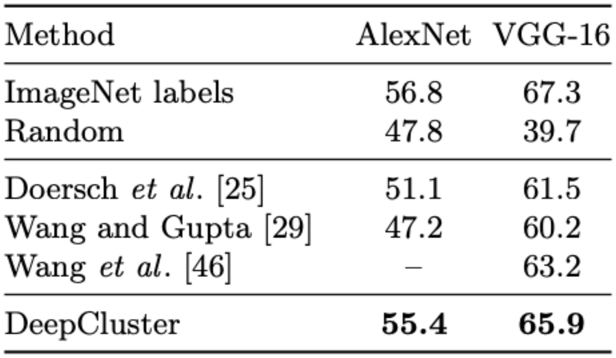
Mathilde et. al. also provides a way to compare where supervised learning starts to become too dependent on the classes provided in the onset. They trained a linear classifier on top of frozen convolutional layers in DeepCluster to showcase these effects. On ImageNet, DeepCluster outperforms the state of the art from conv3 to conv5 layers by 3−5%; however, to truly visualize the predictive power here, when looking at the last layers of the network, improving performance from a standard AlexNet by 14%. This truly shows where labels start to become less significant and 'pseudo-labels' become more important.

Finally, we look at the performance of DeepCluster across all state-of-the-art networks. As shown below, DeepCluster outperforms the best competitor on all relevant metrics, including image classification, object detection and semantic segmentation on both ImageNet and YFCC100M, which has a much more unbalanced object class than ImageNet.



		Classification		Detection		Segmentation	
Method	Training set	FC6-8	ALL	FC6-8	ALL	FC6-8	ALL
Best competitor	ImageNet	63.0	67.7	$43.4^{\dagger}$	53.2	$35.8^{\dagger}$	37.7
DeepCluster	ImageNet	72.0	73.7	51.4	55.4	43.2	45.1
DeepCluster	YFCC100M	67.3	69.3	45.6	53.0	39.2	42.2

Beyond training on ImageNet, Mathilde et. al. validates the results on a much smaller set (PASCAL VOC, 2500 images). They tested both AlexNet and VGG-16 architectures here. VGG-16 outperforms given the deeper layer approach (however the tradeoff of time and computation).



Method	AlexNet	VGG-16
ImageNet labels	56.8	67.3
Random	47.8	39.7
Doersch *et al.* [25]	51.1	61.5
Wang and Gupta [29]	47.2	60.2
Wang *et al.* [46]	–	63.2
DeepCluster	**55.4**	**65.9**

Personal Critique

Positives

Approach: while many researchers take for granted the robustness of CNN models that are trained on ImageNet, many only apply those techniques to their own datasets with adjustments in the parameters used in the specific training on respective datasets. While this shows how empirically powerful these pre-trained models are, there is an unmet need to go beyond and try to improve accuracy on large-scale multi-classification. Thus, it is both surprising and exciting to see researchers improve upon these models by not only training on other datasets, but by applying techniques previously not used in image classification. This builds a strong framework for applications of clustering in image classification, and perhaps a rethinking of how we approach this problem from the onset. The approach the researchers take here is both unique and pioneering, leading the way for an expansion of work in class segmentation.

Training and Testing Methods
Mathilde et. al. recognizes that the empirical models that they use as a base for classification (AlexNet and VGG-16) are both trained on ImageNet, which is a very clean dataset that is balanced evenly within object categories and annotated. In practice, most datasets will never be very balanced (for example, in my project, a large majority of the images are classified as "No Finding"), and will never be annotated as nicely, probably with much sparsity. Thus, it is imperative to train the model on datasets that will be skewed towards both clusters and object classes to actually have real-world applications on many datasets. Thus, the researchers chose to train DeepCluster on YFCC100M, which is randomly chosen from uncured Flickr images, inherently implying a severe imbalance in both the clustering and classification of the images. This shows that the authors understand the limitations of taking the models at face value and chose to go beyond to improve the performance of the DeepCluster. Being such, they were able to validate that DeepCluster can withstand many changes in classification distribution, as performance outperformed state-of-art models on the YFCC100M dataset. Furthermore, the authors performed validation studies for semantic segmentation on Pascal VOC 2012 which is a small set of images (2500) designed to simulate more of a real-life example. DeepCluster proved to robustly perform here as well, out-performing all other models again.

Linear Activation of Specific Layers: the project also aimed to determine the performance of specific convolutional layers, treating each layer as a step-by-step approach, whilst tying together with the end model performance. By digging deeper into how the DeepCluster updates weights for feature output at each layer, we are able to see where label-specific tasks become redundant and cumbersome (as in traditional CNNs), and where a clustering approach mitigates these supervised problems. By not only comparing performance at the very end output of the model, but in the layers of the model as well, the authors are not treating the DeepCluster as a blackbox approach. They are identifying the exact places where we see performance improvement, which can be used in the future to better train weights and pooling if necessary.

Areas of Improvement

Lack of Background into Image Clustering: while the paper continually references *k-means* and the approaches to *k-means* that have proven to produce the most optimal results, there is little discussion into other approaches to clustering (or even *k*-means) that have performed successfully on image work. I will give the others the benefit of the doubt here given that most classification approaches on images use only CNNs; however, there clustering approaches to validate the specificities of how images can be both over and under segmented. A literature review or brief review of how clustering has worked on image datasets and where the state-of-art lies would have been helpful for a novice reader.

Visual Schematics: to best understand how the clustering approach works on top of the CNN in DeepCluster, it would have been useful to have a visual schematic to see how the weights are updated through backpropagation after clustering and separation into pseudo-labels occurs after the *k-means* approach is utilized. The paper dedicates a decent amount of text space into how the architecture is designed but to those who aren't as familiar with the combination of the approaches utilized here, it would be a very helpful supplement to have a visual representation of the architecture.

Sole use of hard assignments: *k-means* is a hard assignment approach, which means each point is assigned to one and only one cluster. With a soft assignment approach, such as weighed K-means or a Gaussian Mixture Model with Expectation Maximization, each point is assigned to all the clusters with weighted probabilities. I understand the point of using *k-means* because labels here are hard image classes, thus the authors would want to recreate that with the pseudo-labels. However, it would have been useful to see if a probability-based method would have affected performance (either positively or negatively). Given that the assignments were updated at each epoch, backpropagating to the weights of the network to finetune end clustering would have looked much different with soft clustering than hard clustering, perhaps giving the model more flexibility in feature output without assuming the true number of clusters.

Resetting Clusters after each epoch: DeepCluster reassigns the clusters after each epoch, which is very computationally intensive and perhaps inefficient given the number of epochs that the authors use to fully train the model. Perhaps a better approach would be to be compute the sample overlap between old and new clusters (the Hungarian algorithm) and assign the cluster IDs translationally and a full reset would not be necessary. This is mainly a critique of the efficiency of the training method and not a necessarily an improvement on overall performance.

Takeaways

- Learned how clustering can be applied on top of a CNN to produce improved performance in overall classification
- An improved understanding of the challenges of using certain image datasets and how to train and validate models over a variety of datasets that have significantly different characteristics to produce the best performance and allow the model to tactfully handle most if not all inputs
- The limitations with different architectures of CNNs (differences in number of layers, hyperparameters, etc.) and how to best optimize for the task at hand, including handling specific layers

- Learned how to best preprocess image data for architectures that aren’t necessarily proficient in handling color channels

Future Steps

- Future clustering approaches can replace the hard *k-means* with either weighted K-means or GMM-EM (Gaussian Mixture Model with Expectation Maximization) to prevent choosing a hard number of clusters at each point and to allow more distribution in weight updating
- Test on purely unannotated and un-labeled data to see if similar performance can be achieved
- Establish similar performance levels with refined networks such as ResNet and DenseNet for both improvements in efficiency and performance

Conclusion

Overall, I learned immensely from this paper on how to build on top of my current image architecture to produce potentially improved performance in simple image classification. It provided a concise yet informative lens into the approach of building such a model and how they actually verified results. I’m looking forward to applying their techniques towards my maximum deliverable

References:

1. Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. “Deep Clustering for Unsupervised Learning of Visual Features.” arXiv:1807.05520 [cs.CV]. Proc. ECCV (2018).
2. "Pneumonia Can Be Prevented-Vaccines Can Help | CDC." Centers for Disease Control and Prevention. 2018. Centers for Disease Control and Prevention. 20 Apr. 2019 &lt;https://www.cdc.gov/pneumonia/prevention.html?CDC_AA_refVal=https%3A%2F%2Fwww.cdc.gov%2Ffeatures%2Fpneumonia%2Findex.html&gt;.
3. Kirton, Orlando. “Mechanical Ventilation - The American Association for the Surgery of Trauma.” American Association for the Surgery of Trauma, AAST, 2011. 20 Apr. 2019. www.aast.org/GeneralInformation/mechanicalventilation.aspx.