Rethinking ImageNet Pretraining in Domain Adaptation

Junzhi Ning

University of Melbourne

Supervised by Dr Mingming Gong

February 24, 2022

Overview

Timeline of the Project

- Familiar with the research topic and learn how to use HPC. (2 weeks)
- Literature Reading, Design the experiment, Implement and Run the Models.(4 weeks)
- Refine the experiments to obtain the final model results. (1 week)
- Report writing and preparation of presentation (1 week)

Context of our exeriment

- Pretrained Models weights help improve the performance of NN models.
- Usually pre-trained Models trained on the larger dataset.
- In computer vision, ImageNet is a good candidate for pretrained models to train on.

Context of our exeriment

- Domain Adaptation: We aim at learning from a source domain a well performing model on a target domain.
- Unsupervised Domain adaptation: Learned from source domains with labeled data to target domains with unlabeled data only

Related Work

- A recent study [citation] shows that removing a portion of classes in the pretrained model from the ImageNet dataset, up to 20 %, can still achieve comparable or even better performance in transfer learning.
- A research[citation] in 2021 suggests that the SSL methods outperform existing UDA methods on the UDA benchmark and therefore should be promoted as baselines in the future.

Intended goal of the project

- Investigate how ImageNet pretraining affect the unsupervised domain adaptation methods.
- Particularly, look into muting some chosen ImageNet class labels of same or similar types in the UDA benchmark datasets when obtaining the pre-trained model.

Alexnet

- Alex-net is an architecture of Convolutional Neural Network
- Proposed by Alex Krizhevsky in the paper[] of 2012.
- Achieved top-5 error rate of 15.3 % in 2012.
- Consist of 8 convolutional layers and 3 of full connected layers. In total, about 61M parameters.

Alexnet

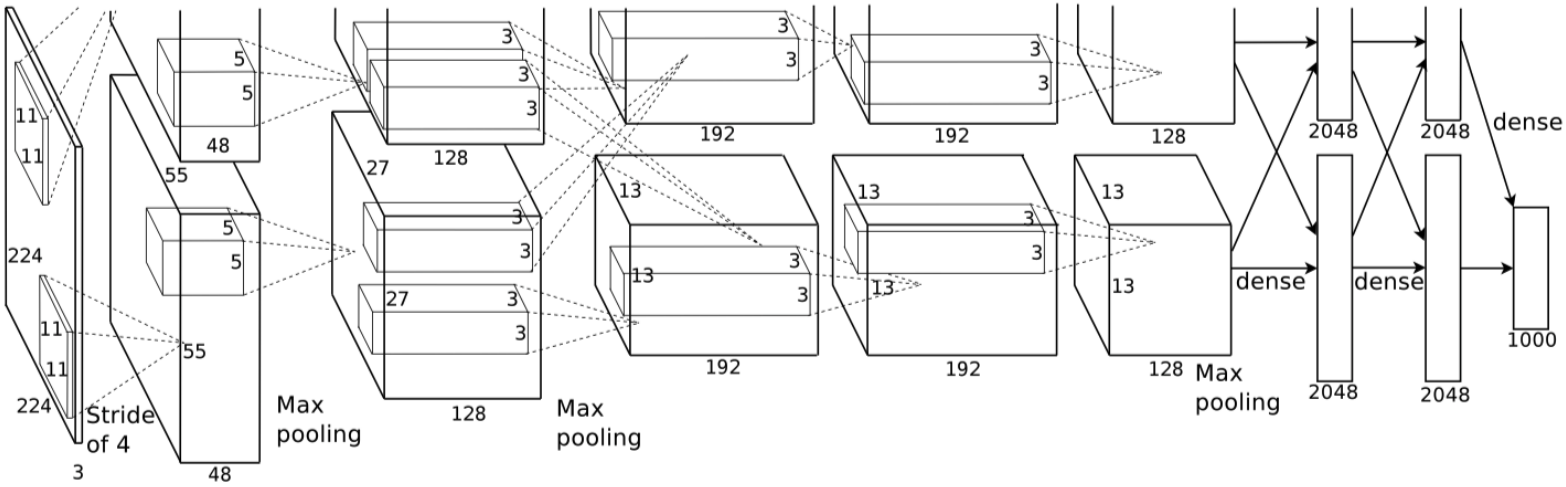


Figure: [citation]Graphical illustration of Alexnet

Concept: Unsupervised Domain Adaptation (UDA)

In a classification task of UDA[],
Given

- n_s labeled samples as a set $D_s = \{x_j, y_j\}_{j=1}^{n_s}$ from source domain with probability distribution of $P_s(X, Y)$
- n_t unlabeled samples as a set $D_t = \{x_j\}_{j=1}^{n_t}$ from target domain with probability distribution of $P_t(X)$

Main goal of UDA: build a classifier C that minimizes the target risk

$$\mathbb{E}_{(x,y')\sim P_t(X,Y)}|C(x) - y'|$$

Concept: Unsupervised Domain Adaptation (UDA)

Two main approaches of UDA:

- **Domain-invariant feature learning.**
- Domain mapping.

Concept: Domain-adversarial Neural Network(DANN)

- DANN was proposed in the paper[3] of 2016 by Yaroslav Ganin.
- Consist of three components, feature extractor, domain classifier and label predictor.
- The feature extractor that generates the feature representation can not be discriminated effectively by the domain classifier.
- The label predictor maps the feature representation by the feature extractor to output the class label regardless of its input domains.

Concept: Domain-adversarial Neural Network(DANN)

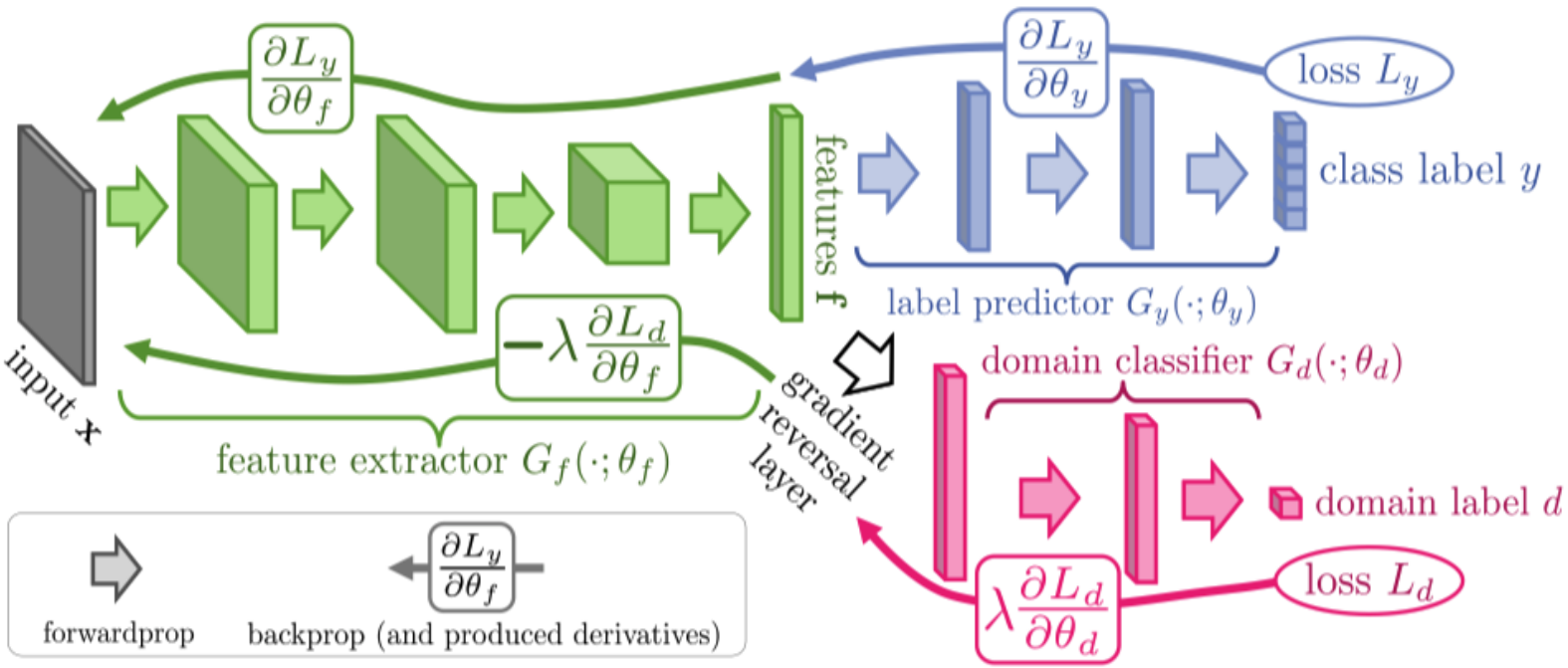


Figure: [citation]Graphical illustration of DANN

Semi-supervised Learning

In a classification of SSL[citation], Given

- n_l labeled samples as a set $D_l = \{x_i, y_i\}_{i=1}^{n_l}$ from $P(X, Y)$
- n_u of unlabeled samples as a set $D_u = \{x_i\}_{i=1}^{n_u}$ from $P(X)$

Main objective of SSL: find the mapping C such that it minimizes the risk

$$\mathbb{E}_{(X,Y)\sim P(X,Y)}|C(x) - y|$$

Pseudo-label Method

- The Pseudo-label method was introduced in the paper of the Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks in 2013.
- During the training iterations, the unlabeled data will being gradually labeled with the class that has the highest probability and confidence.

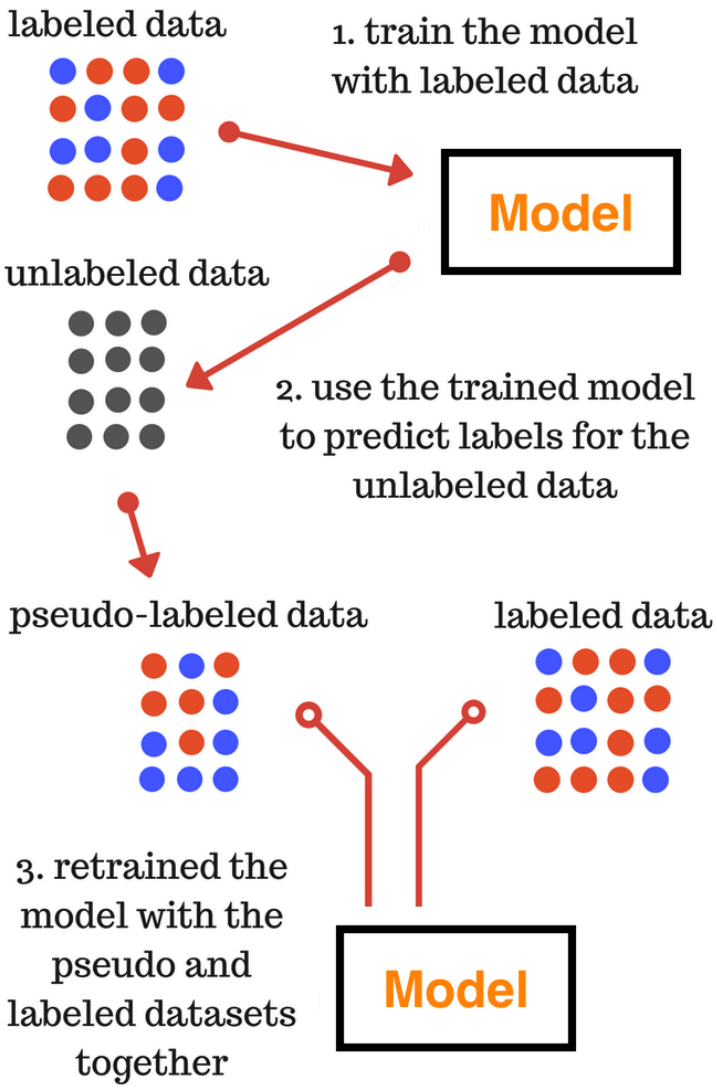


Figure: Graphical illustration of Pseudo-label Method[1]

Datasets

1. **ImageNet (ILSVRC face-blurred)**
   - 1.2 million training labeled images of 1000 categories,
   - Roughly 50 thousand validation images.
   - Used for training our pretrained models.
2. **Office31**
   - Three domains: Amazon, Dslr and Webcam
   - 6 Domain Adaptation Task:
     A $\rightarrow$ W, A $\rightarrow$ D, W $\rightarrow$ D, W $\rightarrow$ A, D $\rightarrow$ A, D $\rightarrow$ W.

Datasets: ImageNet Face-blurred

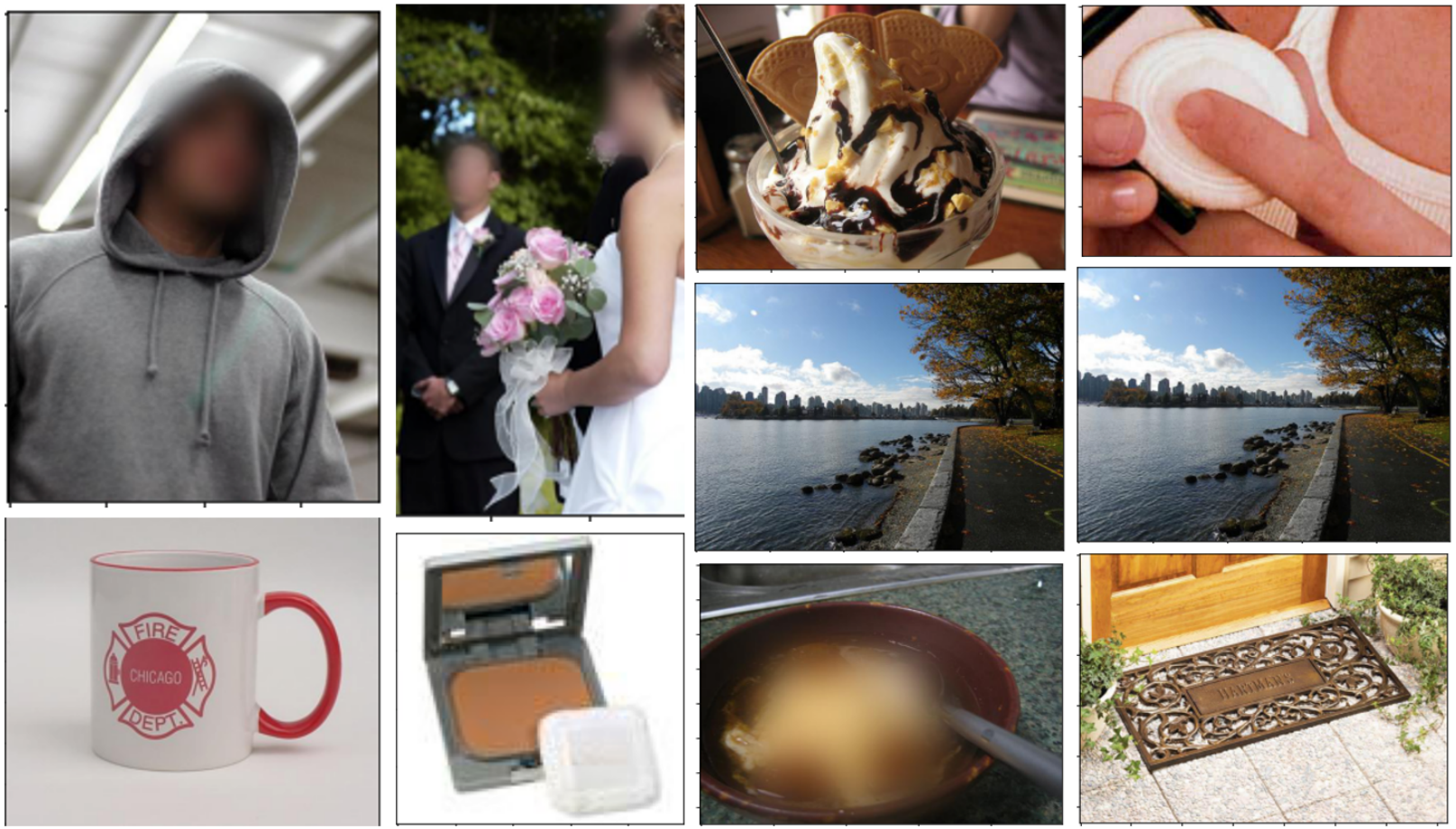


Figure: Example of ImageNet

Datasets: Office31

Figure: Amazon domain

Datasets: Office31

Figure: Dslr domain

Datasets: Office31

Figure: Webcam domain

Setups: Outline

- Create two modified datasets plus the original imageNet:
  1. Dataset with selected masked classes labels.
  2. Dataset with randomly masked classes labels.
- Trained the Alex-net models on the three datasets as pretrained models.
- Use the pre-traiend models as feature extractors to fine-tune the source-only, DANN and Pseudo-label models on six domain adaptation tasks of Office31.

Setups: Pretraind models

Data-sets	No.classes	No.masked classes	No. Training images	No.validation images
Original	1000	0	1281066	49997
Masked	958	42	1226767	47897
Masked Random	958	42	1227158	47897

Table: Three datasets used for training pretraining models

No.Epochs	learning rate	Optimizer	Batch size	Weight decay	LR scheduler
100	0.01	SGD	256	0.0005	Every 30 epochs decay of 0.1

Table: Hyperparameters and setting of pre-trained models.
Note that SGD stands for Stochastic Gradient Descent

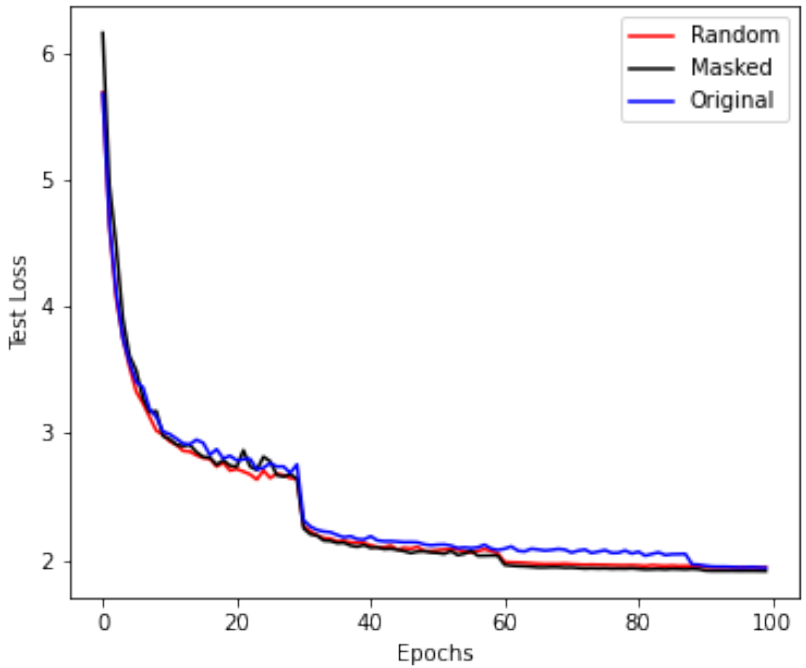
Setups: Fine-tuning

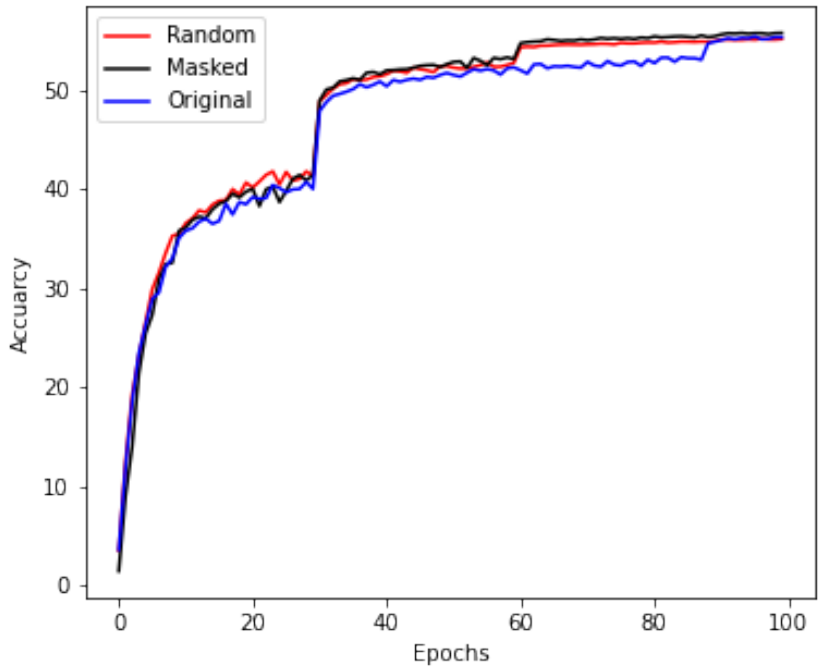
Model types	Source-only	DANN	Pseudo-label
Number of Epochs	50		
Iterations per epoch	1000		
FC layers Learning rate	0.0001	0.01	
Backbone Learning rate	0.00001	0.001	
Optimizer	SGD with a momentum of 0.9		
Batch Size	32		
γ (learning gamma)	0.0003	0.001	
β (learning decay)	0.75		
Learning Rate Scheduler	$LR \times (1 + \gamma \times p)^{-\beta}$		
Weight Decay	0.0005	0.001	

Table: Hyper-parameters and setting of Fine-tuning Models
Note that p is the training process from 0 to 1

Results: Pretraind Models

Figure: Validation Loss and Top-1 Accuracy of Pre-trained Models





Results:Pretraind Models

Table: Results of Pre-trained Models(Alex-net)

Data-set on which Model trained	Best Top 1 accuracy	Best Top 5 accuracy	Test Loss
Original	55.18	78.27	1.949
Masked	55.11	78.93	1.917
Random	55.72	78.41	1.945

Results

Table: Summary of domain adaptation Results (% Accuracy) on Office31(Alex-net)

Dataset of pretrained model	Model Type	$A \rightarrow D$	$A \rightarrow W$	$D \rightarrow A$	$D \rightarrow W$	$W \rightarrow A$	$W \rightarrow D$	Avg.
Original	Source-only	44.4	43	29.7	84.8	31.3	90.2	53.9
Masked	Source-only	43.8	41.4	30.8	83.5	28.9	91.8	53.36
Random	Source-only	46.2	44.8	30.2	81.3	29.7	90.4	53.77
Masked	DANN	49.8	52.2	35.5	94.5	38.4	98.6	61.5
Random	DANN	50.6	55.5	36.4	93.6	42.3	98.8	62.86
Masked	Pseudo-label	41.96	44.65	28.8	90.18	31.55	97.38	55.75
Random	Pseudo-label	43.77	43.52	32.44	91.07	32.62	95.78	56.53

Table: Summary of domain adaptation Results(% Accuracy) at Epoch 1

Dataset of pretrained model	Model Type	$A \rightarrow D$	$A \rightarrow W$	$D \rightarrow A$	$D \rightarrow W$	$W \rightarrow A$	$W \rightarrow D$	Avg.
Masked	Source-only	35.7	34.1	27.4	74.6	22.1	84.3	46.53
Random	Source-only	34.1	35.5	27.2	73.7	25.8	79.5	45.96
Masked	DANN	47.2	47.3	30.4	89.1	31.4	97.0	57.06
Random	DANN	46.8	48.1	30.9	89.6	35.2	98.4	58.17

Dicussions: Pretrained Models

- Pre-trained models of Alex-net converges at 100 epochs with top-1 accuracy of around 55 $\pm$ 0.5% and test loss of 1.94 $\pm$ 0.05 %.
- Our results are still behind that of the original paper, which has the top-1 accuracy of 63.7 %.
- No very substantial accuracy difference between the pretrained model with 1000 classes and the pretrained models (Random and Masked Datasets) with fewer classes.

Discussions: Domain Apdatation

- The domain adaptation tasks perform better of the pretrained model trained on the random dataset.
- The accuracy difference between domains for three methods does not give a consistent pattern.
- These domain adaptation tasks experience a decrease in accuracy in at least one of three methods when the dataset of pre-trained models used changes from Masked to Random.
- Masking some classes in the pretrained model's dataset does not prevent the learning of relevant features in the pretrained dataset associated with Office31.

Dicussions: Domain Adpatation

Possible explanations

- The quality of the pretrained models, DANN and Pseudo-label models underperforms.
- Masking effect of pretrained models on DA tasks can only be reliably estimated through a considerably large number of repeated experiments.
- Omit or fail to mask some or all relevant class labels in the pretrained dataset that are related to categories in Office31.

Limitation of Work and Future Work

- Limited to explore one Neural Network architecture of Alexnet, one relatively small domain adaptation dataset of Office31 and two DANN and SSL methods on pretraining effect.
- Explore more architectures for pretrained models, evaluation of results on various domain adaptation datasets

Acknowledgements

I would like to express my deep gratitude to Dr Mingming Gong, my supervisor, who kindly provided immerse help and timely guidance over the 8-weeks summer course of this project.
This project was undertaken using Spartan platform of High Performance Computing at the university of Melbourne, I am grateful of the University of Melbourne providing such resources to fellow researchers and students like me.
Also, I would like to thank Dongting for providing the help when I faced technical difficulties in using HPC.

The End

References


Shubham, J., 2017. Pseudo Labeling — Semi Supervised Learning. Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2017/09/pseudo-labelling-semi-su [Accessed 23 February 2022].