Planning Confident Predictions for Semi-Supervised Learning

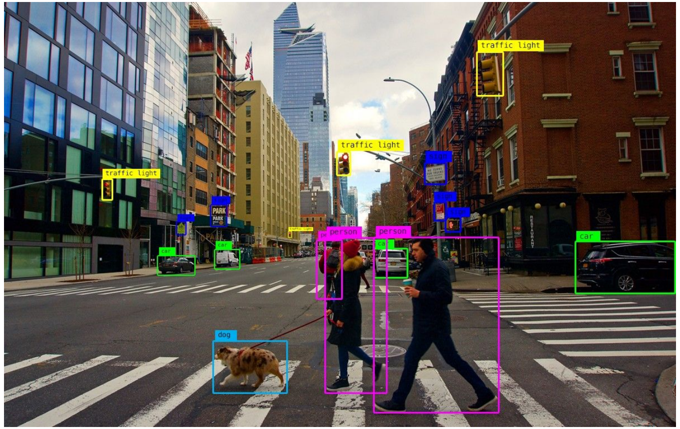
Eduardo H. P. Pooch
PhD Student in Computer Sciece

Automated Planning 2021 - PUCRS

Introduction

Deep learning models usually need a lot of labeled data to achieve good results

Data annotation might be expensive and time-consuming if the people annotating the images need to be experts or if the annotation demands a lot of time

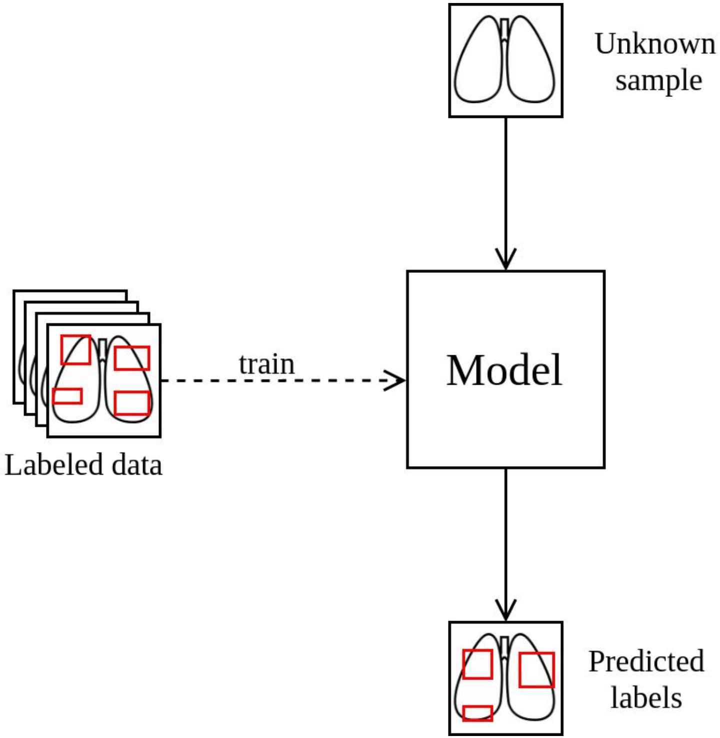


Introduction

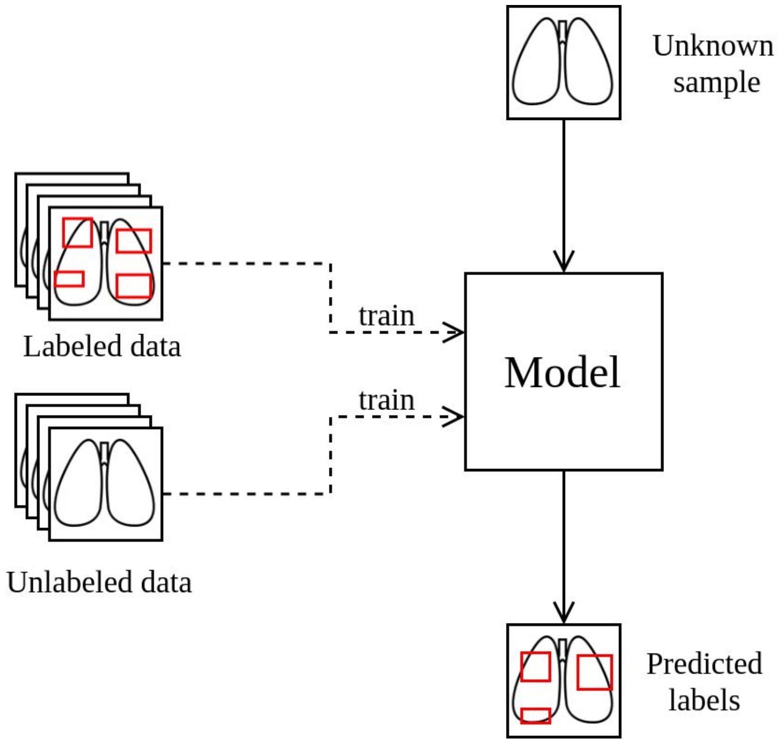
- Recently, SSL methods had a considerable progress[1]
- SVHN dataset (73,257 samples)[2]
  - Supervised (All labels): **2.59% error**
  - UDA[3] (250 labels): **2.72% error**

1. Qi, G.-J.; Luo, J. "Small data challenges in big data era: A survey of recent progress on unsupervised and semi-supervised methods", arXiv preprint arXiv:1903.11260 (2019)
2. Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; Ng, A. Y. "Reading digits in natural images with unsupervised feature learning". (NIPS 2011)
3. Xie, Q., Dai, Z., Hovy, E., Luong, M.-T., & Le, Q. V. "Unsupervised Data Augmentation". (2019)

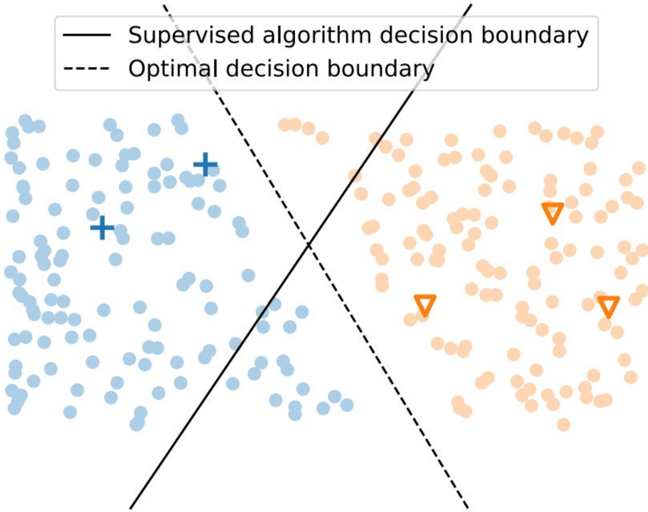
Supervised learning



Semi-supervised learning



- Smoothness assumption:
  - Two close samples have the same labels
- Low density assumption:
  - The decision boundary lies in a low density region



Source: van Engelen, J. E.; Hoos, H. H. “A survey on semi-supervised learning”, Machine Learning, (2019)

Semi-supervised learning

- Perturbation-based methods include a **consistency loss** term on the optimization function

$$\mathcal{L}_{comb} = \mathcal{L}_{task} + \gamma \mathcal{L}_{cons}$$

- Entropy minimization methods encourage the model to make **confident predictions**

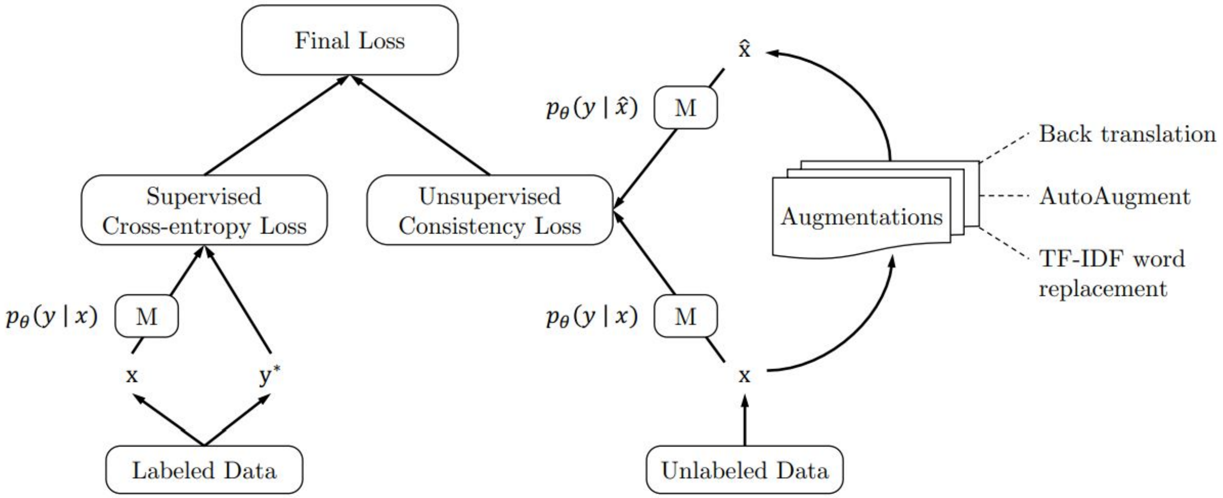
Unsupervised Data Augmentation (UDA) [1]

- One model receives strongly augmented and non-augmented data
- The consistency loss is a KL-divergence between both predictions

$$\mathcal{L}_{cons} = log\frac{m(\phi(u))}{m(u)}$$

1. Xie, Q., Dai, Z., Hovy, E., Luong, M.-T., & Le, Q. V. “Unsupervised Data Augmentation”. (2019)

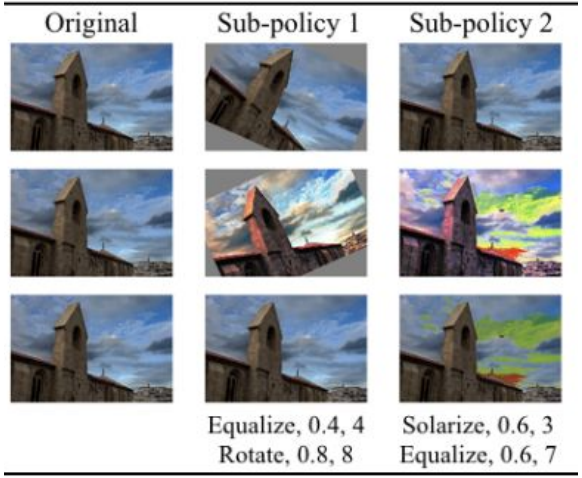
Unsupervised Data Augmentation (UDA) [1]



1. Xie, Q., Dai, Z., Hovy, E., Luong, M.-T., & Le, Q. V. "Unsupervised Data Augmentation". (2019)

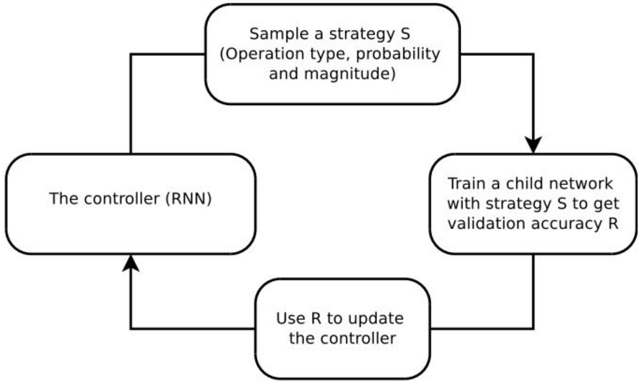
AutoAugment

- Designs a **search space** for image augmentation
- A policy consists of many **sub-policies** , one is randomly sampled for each image
- A sub-policy consists of two operations (e.g. translation, rotation), their **probability** and **magnitude**



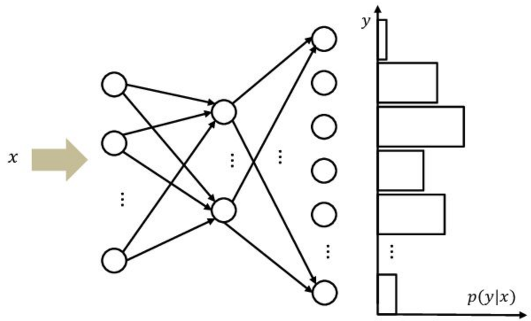
AutoAugment

- It samples a data augmentation policy S, containing the operation type, probability and magnitude
- Train a small network using S, and get the validation accuracy R
- Use R to update the controller
- Since R is not differentiable, the controller is updated by policy gradient methods



Goal

- To develop a novel augmentation technique
- Instead of using the accuracy as the reward, it will reward a maximum output probability score
- This way, it enforces confident predictions and rests on the low-density assumption to achieve a better augmentation policy for semi-supervised learning



Technical approach

- We aim to develop a Markov Decision Process (MDP) to search optimal augmentation policies.
- In this MDP, the **states** are the probability and magnitude values of each of the 16 image operation functions on the Python Image Library (PIL).

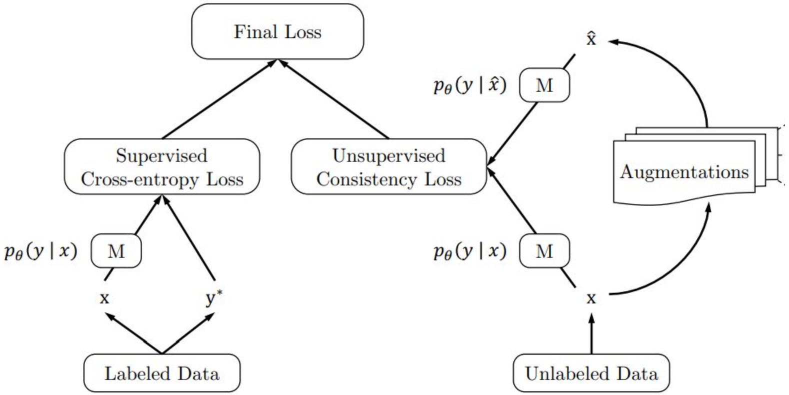
Technical approach

- The possible **actions** are to lower or increase the values of probability and magnitude of each image transformation function.
- Since we want to increase the confidence of the model, the **reward** would be the sum of the maximum output score probability of the model on a subset of unlabeled data

$$\sum_i^n \max(m(\phi(u_i)))$$

Technical approach

- This policy can be learned at each iteration of the model, based on a batch of samples
- Or, learned offline on a subset of the data and then used during multiple iterations



Schedule

- Week 1: The first step of the project is to reproduce the UDA framework using a policy learned by AutoAugment;
- Week 2: Code the proposed MDP environment and study the best search algorithm for the proposed approach;
- Week 3: Implement the search algorithm on the proposed MDP
- Week 4: Include the augmentation policy on the UDA framework, and evaluate the performance of the approach;
- Week 5: Report the results on the final paper.

Conclusion

- At the end of this work, we expect to develop a novel semi-supervised learning approach that better encourages **both consistency regularization through the UDA framework and entropy minimization through a novel augmentation mechanism**.
- We hope that this method can accomplish better results on semi-supervised learning, helping to advance the state-of-the-art in machine learning with limited supervision.

Thank you