Semi-Supervised Learning

Marcus Bloice

Medical University Graz

13th April 2021

Introduction

This mini-project relates to **semi-supervised** learning in **computer vision**

1. We will first briefly discuss the topic
2. Then briefly discuss a recent paper
3. Finally the project will be described

Semi-Supervised Learning

What is semi-supervised learning?

- This is a type of middle ground between **unsupervised** and **supervised** learning
- It is a broad term for techniques where you may have both **labelled** and **unlabelled data**
- Normally, it is for when the **amount of labelled data** $\ll$ **amount of unlabelled data**

Semi-Supervised Learning: An Example

Imagine you had a dataset of **a million cats and dogs**, that you scraped off the internet and therefore **you have no labels**

Semi-Supervised Learning: An Example

Imagine you had a dataset of **a million cats and dogs**, that you scraped off the internet and therefore **you have no labels**

Labelling a million images is not possible so you **label maybe 1,000 of them**

Semi-Supervised Learning: An Example

Imagine you had a dataset of **a million cats and dogs**, that you scraped off the internet and therefore **you have no labels**

Labelling a million images is not possible so you **label maybe 1,000 of them**

You then train a network on these **1,000 labelled data**...

Semi-Supervised Learning: An Example

Imagine you had a dataset of **a million cats and dogs**, that you scraped off the internet and therefore **you have no labels**

Labelling a million images is not possible so you **label maybe 1,000 of them**

You then train a network on these **1,000 labelled data**. . .

Then, you use your freshly trained model to **label** the remaining 999k images!

Semi-Supervised Learning: An Example

Imagine you had a dataset of **a million cats and dogs**, that you scraped off the internet and therefore **you have no labels**

Labelling a million images is not possible so you **label maybe 1,000 of them**

You then train a network on these **1,000 labelled data**. . .

Then, you use your freshly trained model to **label** the remaining 999k images!

Finally you train a last model on your 1 million **labelled data**!

Semi-Supervised Learning: An Example

Imagine you had a dataset of **a million cats and dogs**, that you scraped off the internet and therefore **you have no labels**

Labelling a million images is not possible so you **label maybe 1,000 of them**

You then train a network on these **1,000 labelled data**. . .

Then, you use your freshly trained model to **label** the remaining 999k images!

Finally you train a last model on your 1 million **labelled data**!

Such approaches are now getting **close to the results of purely supervised methods**!

SimCLR

- There are actually many approaches to semi-supervised learning. . .
- So we will discuss one particular approach used by Google Brain in a recent paper[1]
- The paper discusses **SimCLR**, a **semi-supervised approach for image classification**
- Your **mini project is to implement this approach** on a dataset of your choosing

Now we will describe the SimCLR workflow, which is performed in 3 stages. . .

[1] **Big Self-Supervised Models are Strong Semi-Supervised Learners**, Chen et al., *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*

SimCLR: Stage 1 of 3

- In the first phase of the SimCLR workflow, a purely unsupervised neural network (ResNet) is trained
- You discard your labels completely!
- You feed the network **similar** and **dissimilar** images, so **that it learns to differentiate them**

SimCLR: Stage 1 of 3

- In the first phase of the SimCLR workflow, a purely unsupervised neural network (ResNet) is trained
- You discard your labels completely!
- You feed the network **similar** and **dissimilar** images, so **that it learns to differentiate them**

But how do we do this if we do not have any labels?!?

SimCLR: Stage 1 of 3

- In the first phase of the SimCLR workflow, a purely unsupervised neural network (ResNet) is trained
- You discard your labels completely!
- You feed the network **similar** and **dissimilar** images, so **that it learns to differentiate them**

But how do we do this if we do not have any labels?!?

- Augmentation is used! We can create **pairs of similar images from a single image**
- If we take another **random image from the dataset**, we can say this is **less similar than the augmented pair**
- Slowly the network learns what makes images similar and what makes them different

SimCLR: Stage 1 of 3

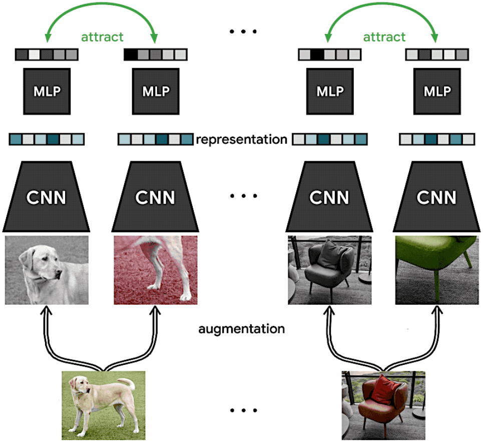

Figure 1: Similar images attract

SimCLR: Stage 1 of 3

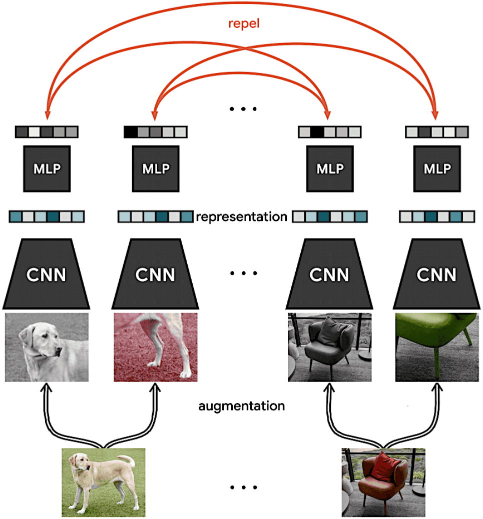

Figure 2: Dissimilar images repel

SimCLR: Stage 2 of 3

So, the first part is pre-trained and the network can somewhat classify images.

In the second stage we now use our labelled data!

- Basically, we **fine tune our network from stage 1**
- We take our **labelled subset and update the network** in the standard way you'd fine tune a pre-trained network
- Once this second stage of training is complete we now have even better performing network, however there is a further stage. . .

SimCLR: Stage 3 of 3

This is the **self-supervised** phase

- You take your trained model from Stage 2 and pass all your data through it
- It outputs predictions, so for the image $x^{(i)}$ we get prediction $\hat{y}^{(i)} = [0.9, 0.1]$
- This image $x^{(i)}$ plus the label $\hat{y}^{(i)}$ is used to train a new network!

This trained model is then the final output of the workflow.

SimCLR: Results

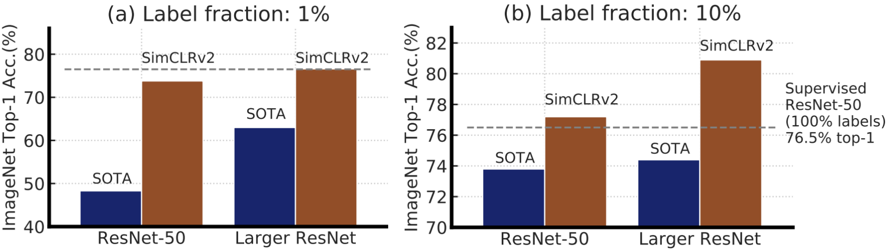

Figure 3: SimCLR versus supervised SOTA, 1% and 10% labels

SimCLR: Summary

So to summarise the approach:

1. Purely unsupervised, pre-training phase, using augmentation
2. Supervised, fine tuning
3. Self-supervised training

Pros & Cons

- **Pro**: We can achieve SOTA accuracy with a fraction of the labelled data
- **Pro**: Useful if collecting labels is expensive, like medicine
- **Con**: The first stage, the purely unsupervised stage, is time consuming and quite slow

Mini-Project

So, SimCLR forms the basis for your mini-project.

Your tasks:

1. **Choose a dataset**—easiest is to start with a labelled dataset and just discard the labels as required, or create your own dataset by labelling yourself!
2. **Implement SimCLR**—source code is available for TensorFlow! Use **Google Colab** and prepare a notebook to submit—free GPU access!
3. Compare supervised, versus 1% labels, versus 10% labels!
4. Submit your **single Colab notebook** as your assignment!

Conclusion & Resources

Any Questions?

Organisational:

- Discord server: https://discord.gg/W9se9Rxw
- My e-mail: marcus.bloice@medunigraz.at
- Google Colab https://colab.research.google.com

SimCLR Method:

- Paper full text: https://arxiv.org/abs/2006.10029
- SimCLR source code: https://github.com/google-research/simclr
- Paper video walkthrough: https://www.youtube.com/watch?v=2lkUNDZld-4