



Odisee
DE CO-HOGESCHOOL

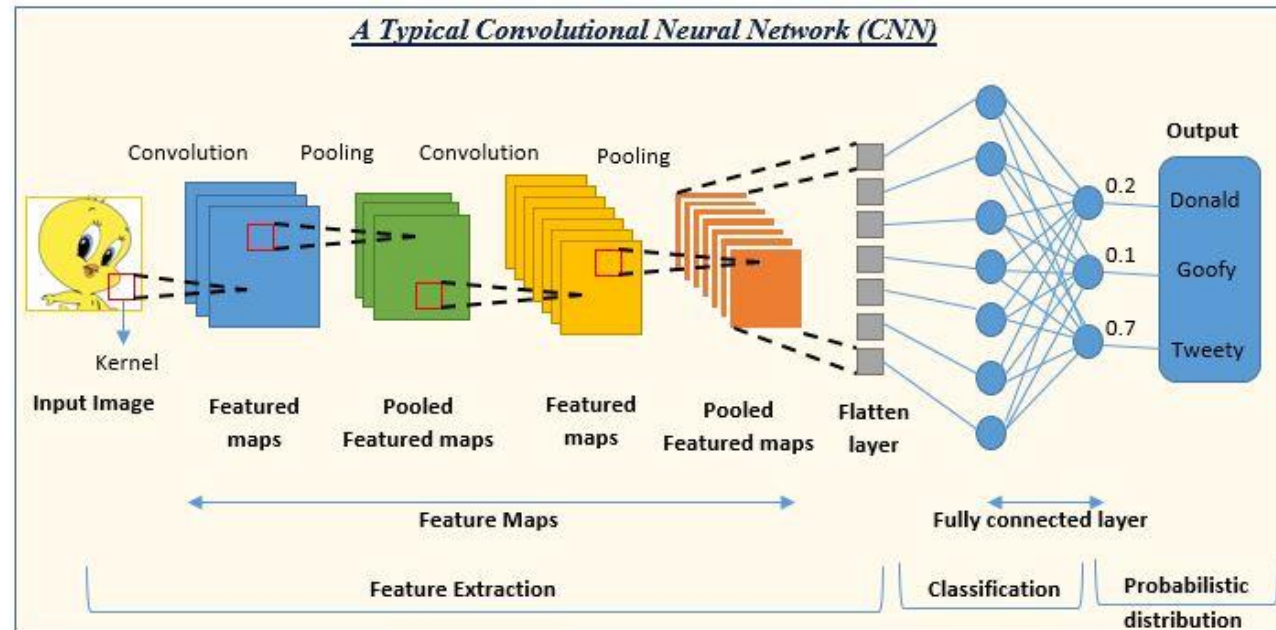
Machine learning – Week 5



Convolutioneel neuraal netwerk


Bestaat uit 2 delen


- Convolutionele + Pooling lagen -> Feature extraction
- Neuraal network -> Features to output





Pretrained networks

- 
- ▣ Netwerk dat niet door jou getrained is met data die meestal niet beschikbaar is voor jou
 - ▣ Wat moet je weten om het te gebruiken?

- 
- ▣ Netwerk dat niet door jou getrained is met data die meestal niet beschikbaar is voor jou
 - ▣ Wat moet je weten om het te gebruiken?
 - Welke lagen zijn er aanwezig
 - Type, kernels, strides, type pooling, aantal neurons, activation functions, ...
 - Alle gewichten van alle lagen
 - Kernels en neurons
 - Input dimensions en output dimensions

Waar vind je pretrained networks?

- ▣ <https://www.tensorflow.org/hub>

- Bevat een groot aantal pretrained networks voor o.a. computer visie en tekstverwerking
 - Voor afbeelding van verschillend formaat
 - Voorbeeldcode hoe het te gebruiken

- ▣ https://www.tensorflow.org/api_docs/python/tf/keras/applications

- Vooral veel gebruikte computer visie applicaties

- ▣ Github / Google Search

- <https://github.com/tensorflow/models/tree/master/research/audioset/yamnet>

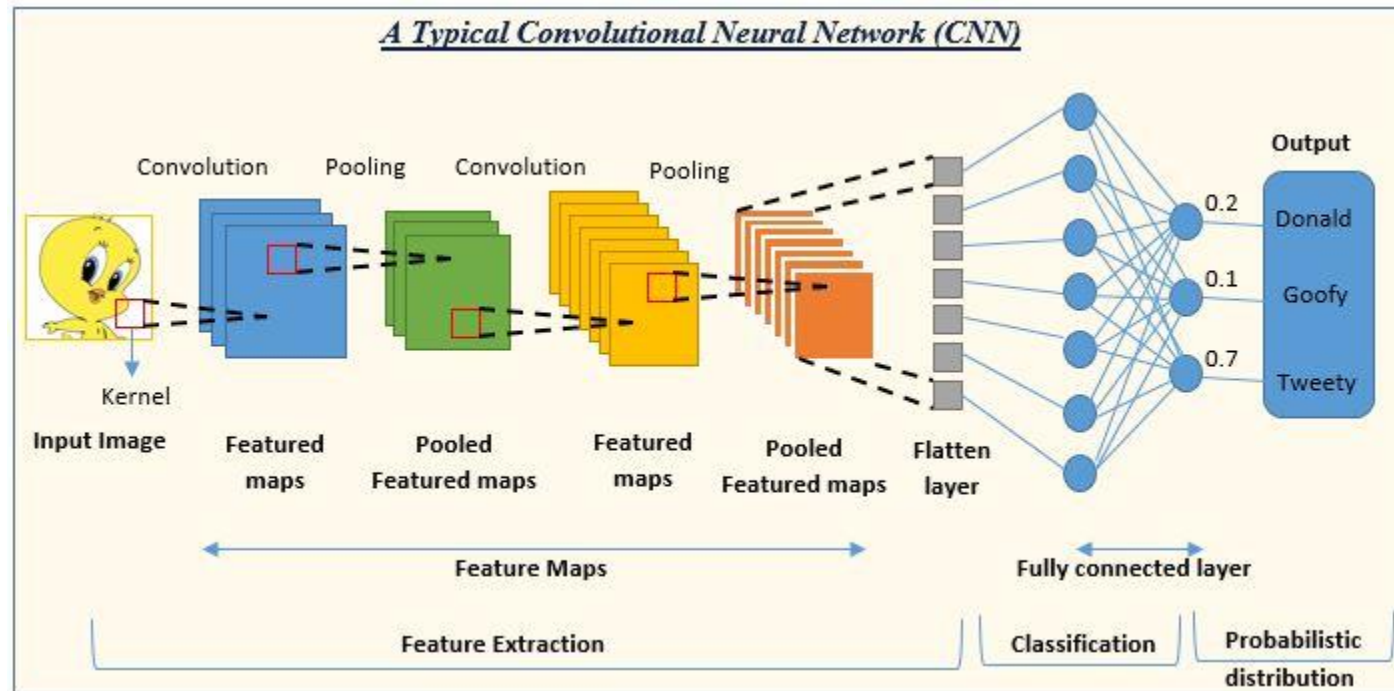


Oefening

- Ga naar https://tfhub.dev/tensorflow/collections/object_detection/1
 - ▬ Hoeveel klassen zijn er in de dataset waarvoor de modellen getraind zijn?
- Selecteer het SSD MobileNet v2 320x320 model
 - ▬ In welke output zit de beste klasse voor elke detectie
- Open de Colab Notebook en bestudeer de code
 - ▬ Besteed hier voldoende aandacht aan. Het uitvoeren van een pretrained netwerk is deel van de huistaak

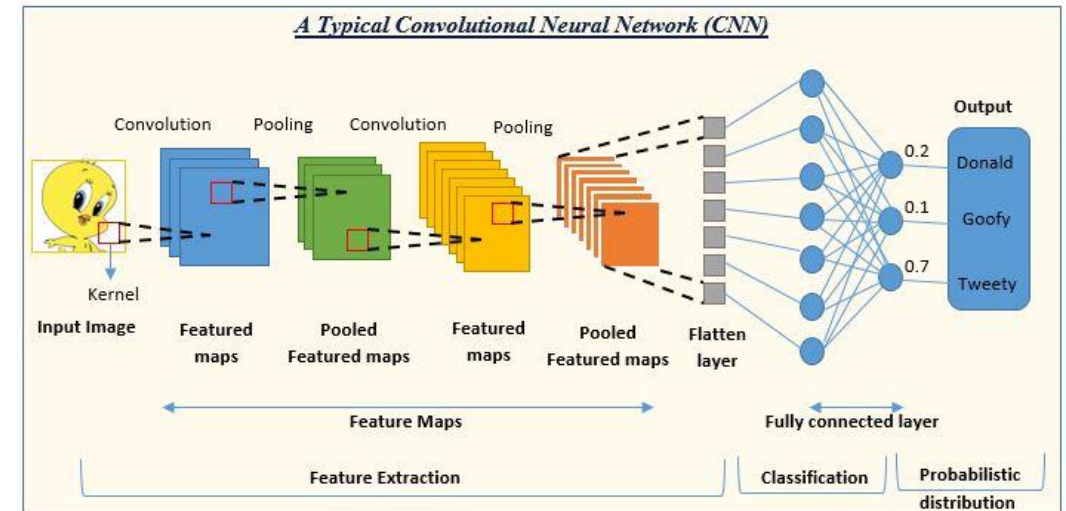
Wat als er geen exact netwerk bestaat?

- ▣ Kan een bestaand netwerk aangepast worden?
 - Hoe zou je dit aanpakken?



Wat als er geen exact netwerk bestaat?

- ▣ Behoud convolutionele en pooling lagen
- ▣ Pas (gedeeltelijk) neural netwerk aan
 - Laatste laag/lagen aan te passen
 - Voeg laag toe
 - Output kan zijn wat we nodig hebben





Transfer learning



Wat is transfer learning?

- ▣ Bij transfer learning nemen we het grootste deel van het pretrained netwerk over
- ▣ Enkel de laatste lagen worden vervangen om je probleem op te lossen
 - Enkel de gewichten van de vervangen lagen worden getraind
 - De gewichten van de niet aangepaste gewichten blijven hetzelfde



Hoe vervang je de laatste lagen?

- ▣ Baseer je op de volgende guide:

https://www.tensorflow.org/guide/keras/transfer_learning

- ▢ Heel vaak is er een argument: `include_top=False`
 - Top is het fully-connected gedeelte na de flatten
- ▢ Anders: `layers` property is een list
 - Pop/slice de nodige layers eruit
 - append nieuwe layers of maak een model aan met de nieuwe lagen eraan toegevoegd



Fine tuning





Fine tuning

- ▣ Uitbreiding op transfer learning
 - Gewichten van het reeds bestaande model worden niet gefreezeed
 - Nieuwe data om verder te trainen om te fine-tunen naar wat je wil bereiken
- ▣ Voorbeeld: algemeen netwerk meer data geven over hondenrassen om de accuraatheid voor dat problem te verbeteren.



Belangrijke opmerkingen

- ▣ Voer `compile()` uit na aanpassen van de trainable property van het model/lagen
 - ▬ Anders wordt de wijziging niet doorgevoerd
- ▣ BatchNormalization
 - ▬ Bevat twee interne gewichten (mean en variance)
 - ▬ Na unfreezen voor fine-tuning moeten deze lagen terug op `trainable=false` gezet worden
 - ▬ Anders pas je de verdeling aan waardoor het reeds geleerde verwijderd wordt

How to participate?




Click on the projected screen to start the question

 [Copy participation link](#)

wooclap

 100 % 

16 



 You cannot vote anymore



In welke volgorde moet je een Neuraal Netwerk trainen met tensorflow



✓ The correct combination was:

Kiezen tussen sequentieel model / functional API

Toevoegen van preprocessing lagen

Opstellen van de architectuur van het Neuraal netwerk

Compileren van het model met keuze loss-functie en learning rate optimizers

Trainen van de gewichten in het neuraal netwerk

Evaluation

Click on the projected screen to start the question

wooclap



100 %



14% correct

7 / 16



Go to **wooclap.com** and use the code **DXPKPK**



Wat is GEEN hyperparameter van een convolutionele laag?



1

Stride

17%

2



2

Aantal kernels

58%

7



3

Loss functie

17%

2



4

Activatiefunctie

8%

1



Click on the projected screen to start the question

wooclap



100 %



12 / 16



Go to **wooclap.com** and use the code **DXPKPK**



Welke lagen worden NIET gebruikt in het convolutioneel gedeelte van een ..



1

Convolutionele laag

38%

5



5

Dense Layer

46%

6



2

Random Rotation

92%

12



6

Embedded Layer

46%

6



3

Max Pooling

31%

4



7

Batch Normalization

69%

9



4

Average Pooling

69%

9



Click on the projected screen to start the question

wooclap



100 %



13 / 16



Go to **wooclap.com** and use the code **DXPKPK**



Transfer learning houdt in dat

1 je een pretrained network volledig verder trained

9%

1



2 je de structuur van een ander network neemt en vanaf het begin begint met trainen

9%

1



3 je een deel van een reeds getrained network behoudt en een deel vervangt

82%

9



Click on the projected screen to start the question

wooclap



100 %



11 / 16



Go to **wooclap.com** and use the code **DXPKPK**



Fine tuning houdt in dat



1

je een pretrained netwerk verder traint met nieuwe data

91%

10



2

je een deel van een pretrained netwerk behoudt en een vervangt en enkel het vervangen gedeelte hertrained

9%

1



3

je een pretrained netwerk volledig hertrained.

0%

0



wooclap



100 %



11 / 16





Huiswerk





Huistaak over convolutionele netwerken en pretrained netwerken

- ▣ Opgave zie Toledo
- ▣ Maak gebruik van een pre-trained netwerk met transfer learning om een ML-probleem op te lossen.