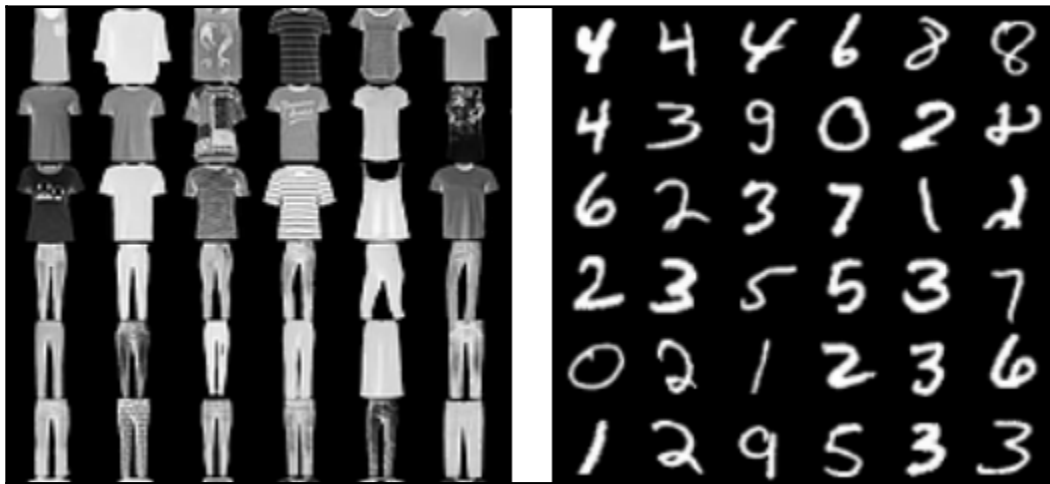
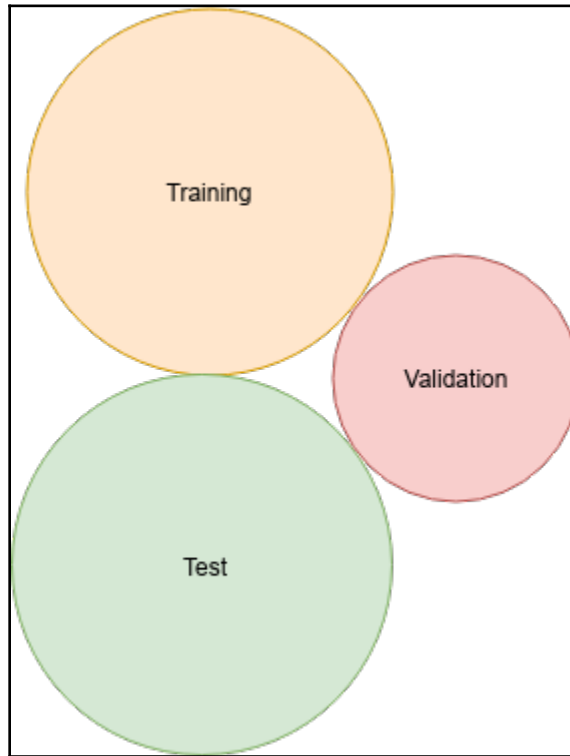
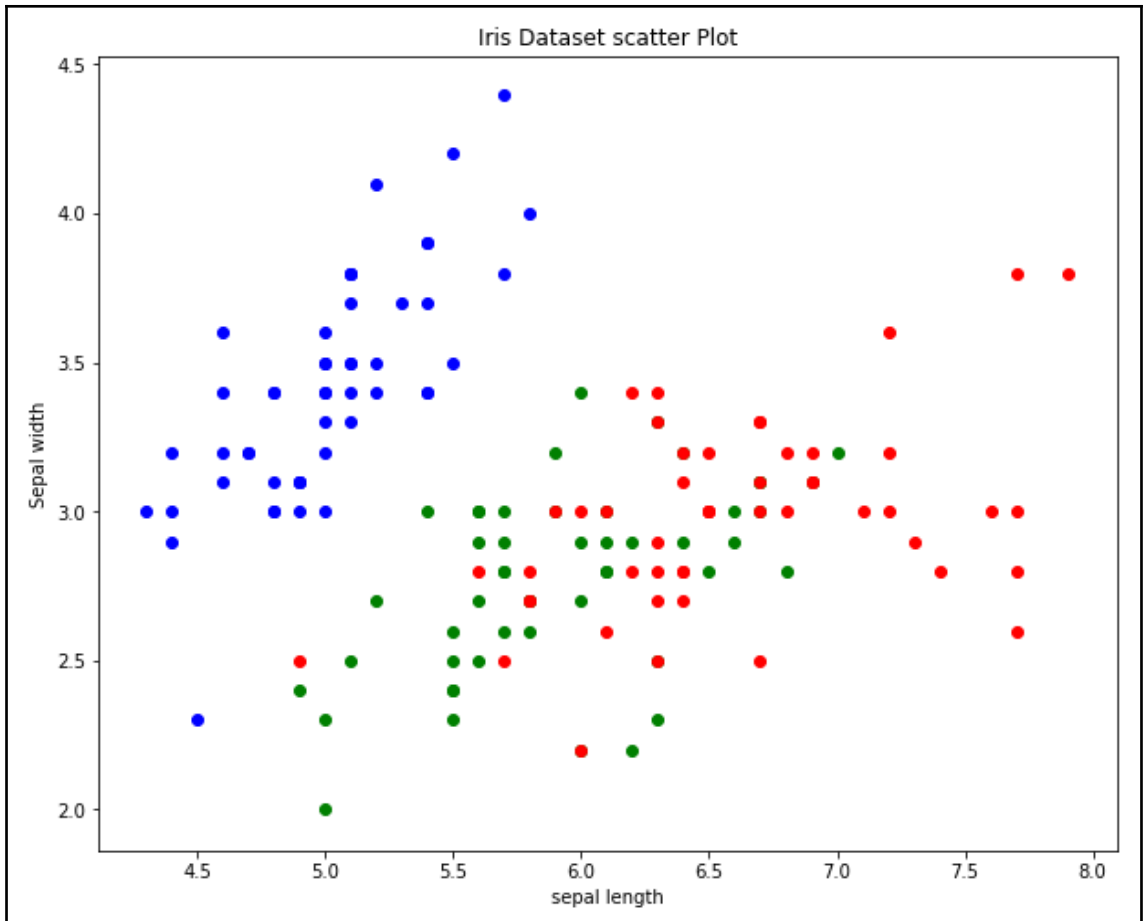
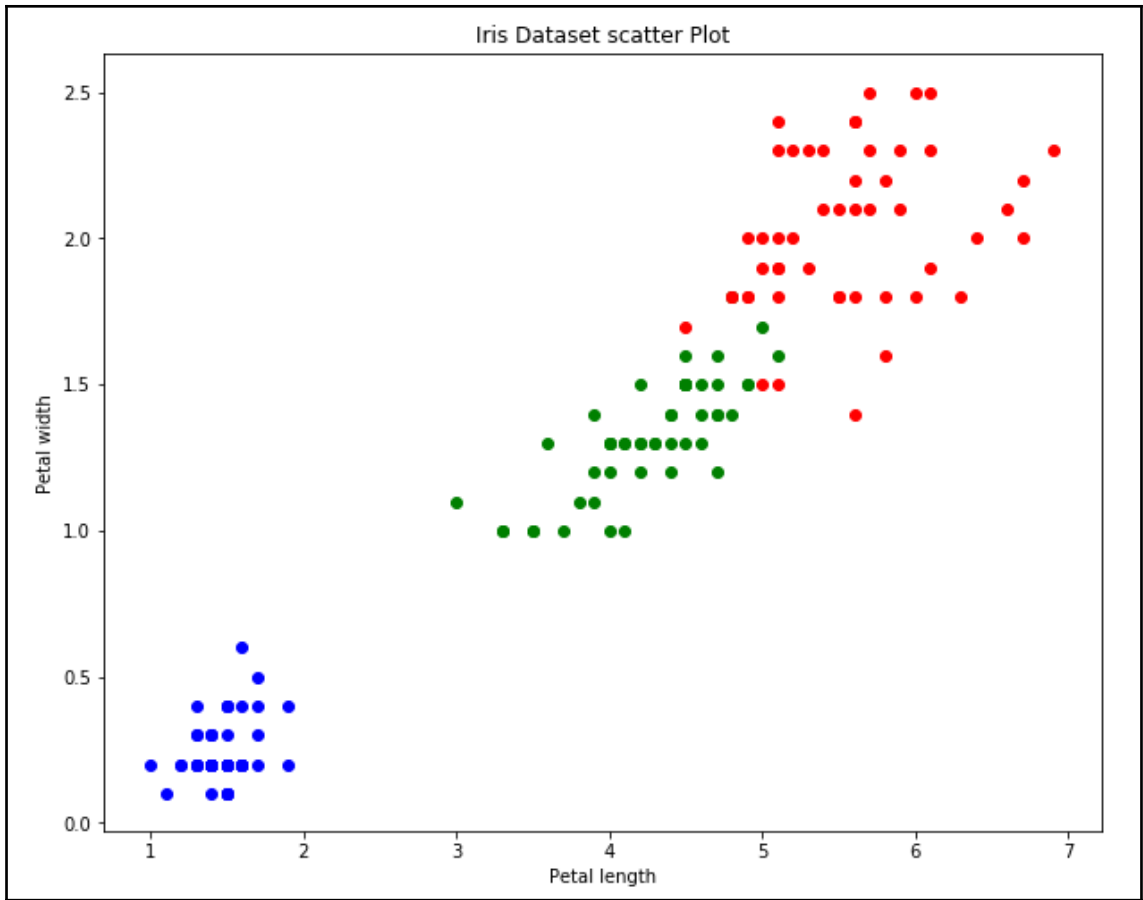
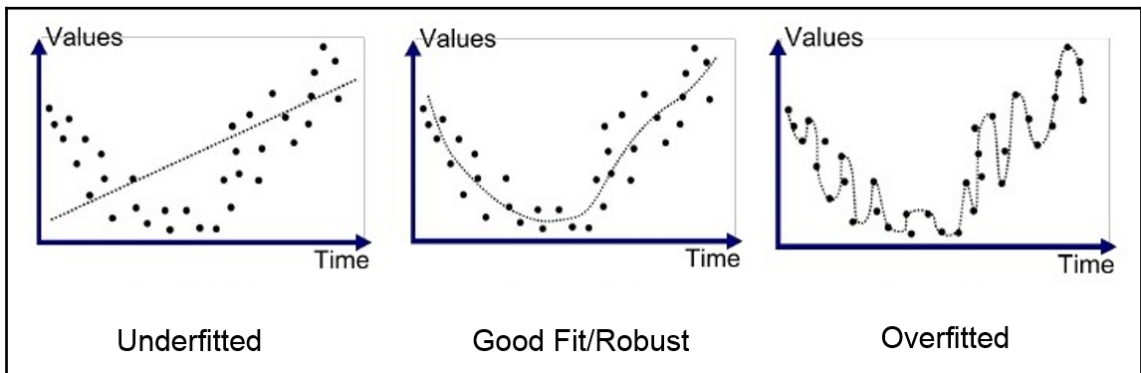
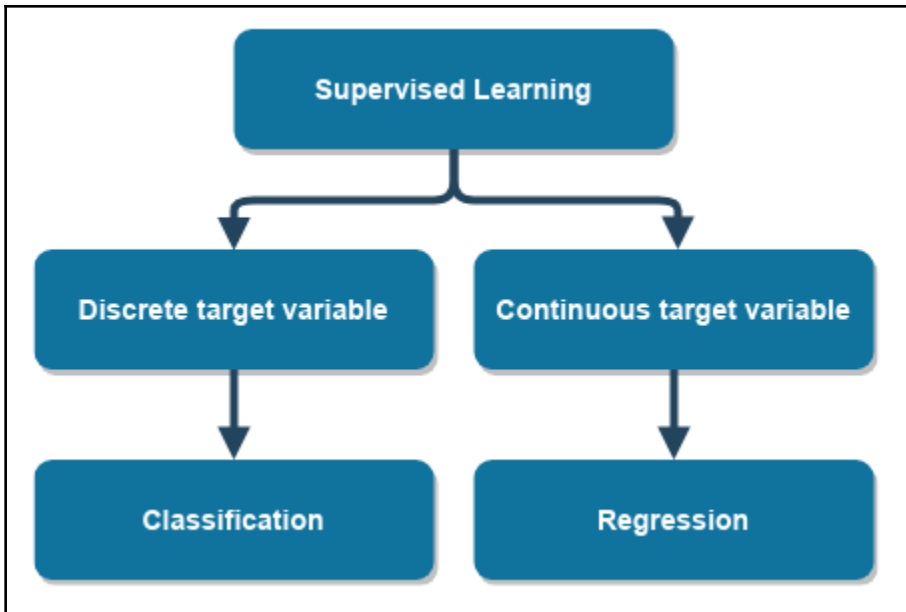


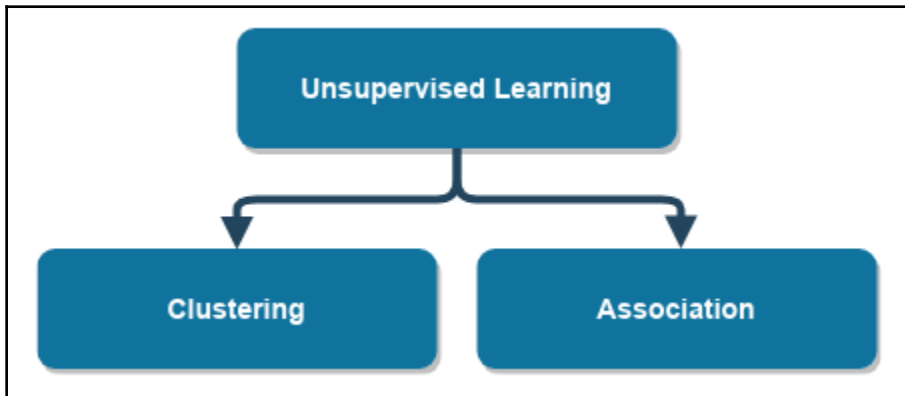
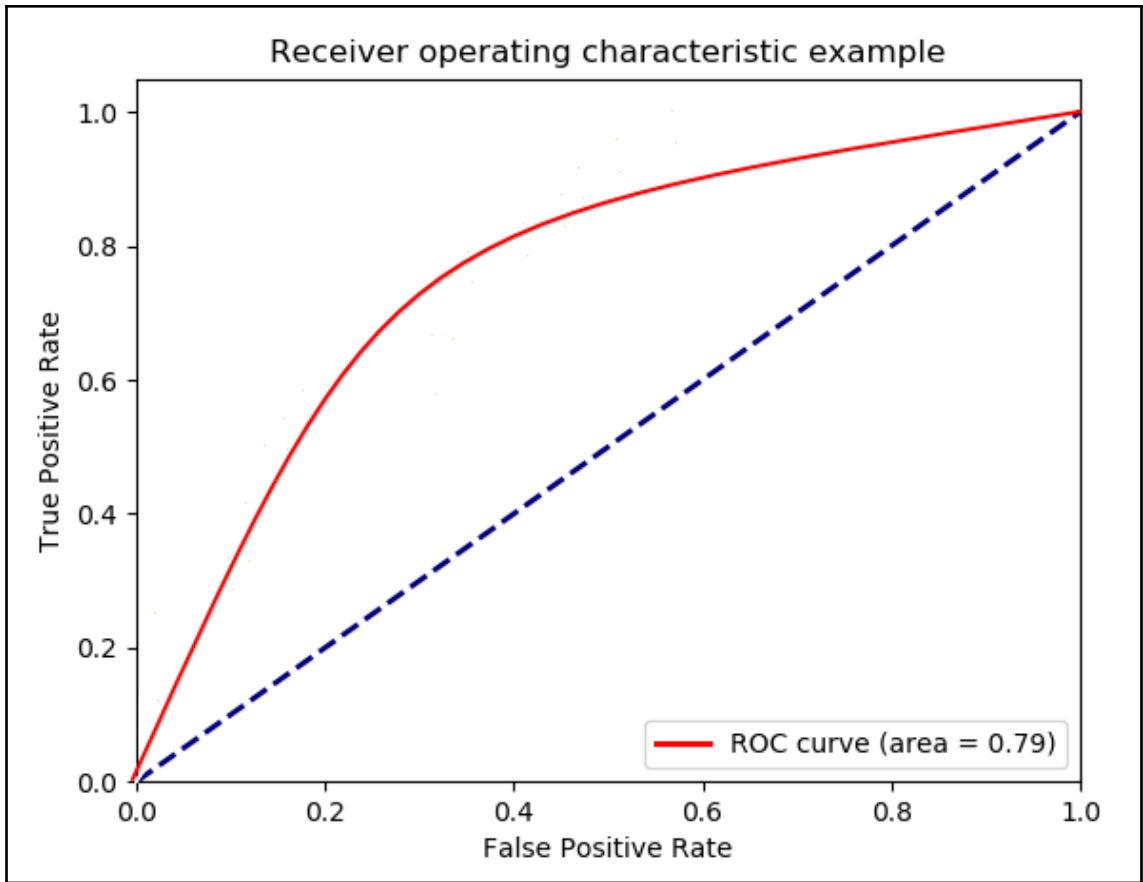
Chapter 1: What is Machine Learning?



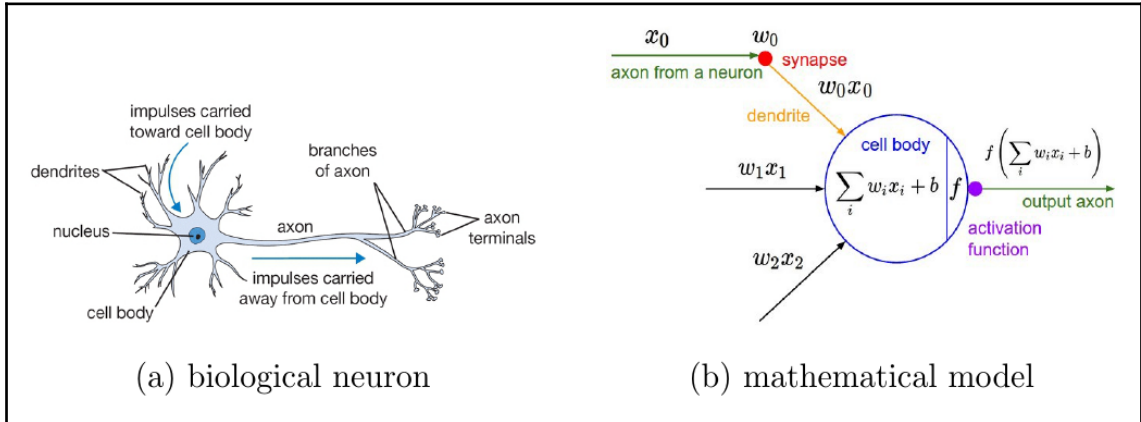


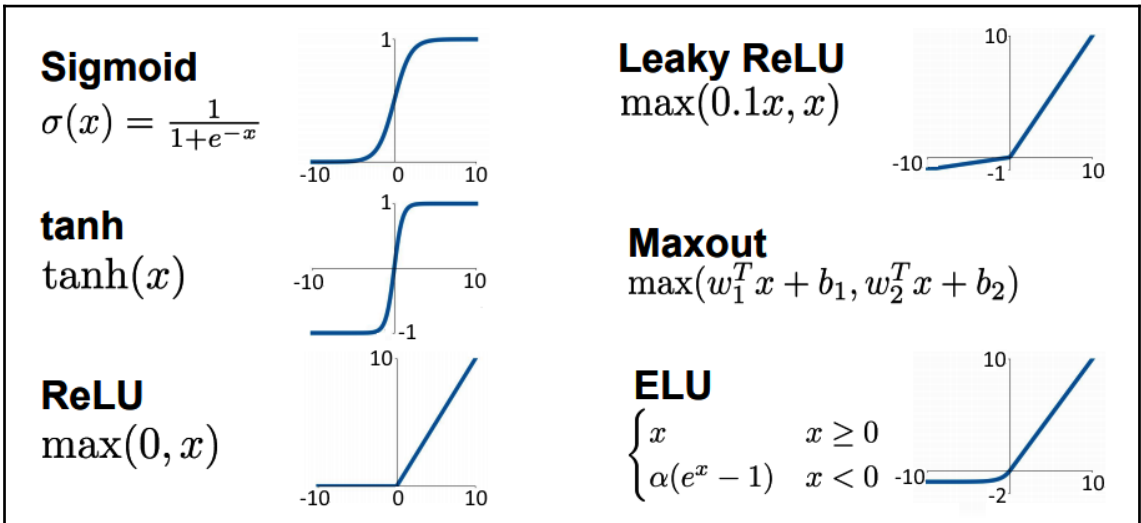
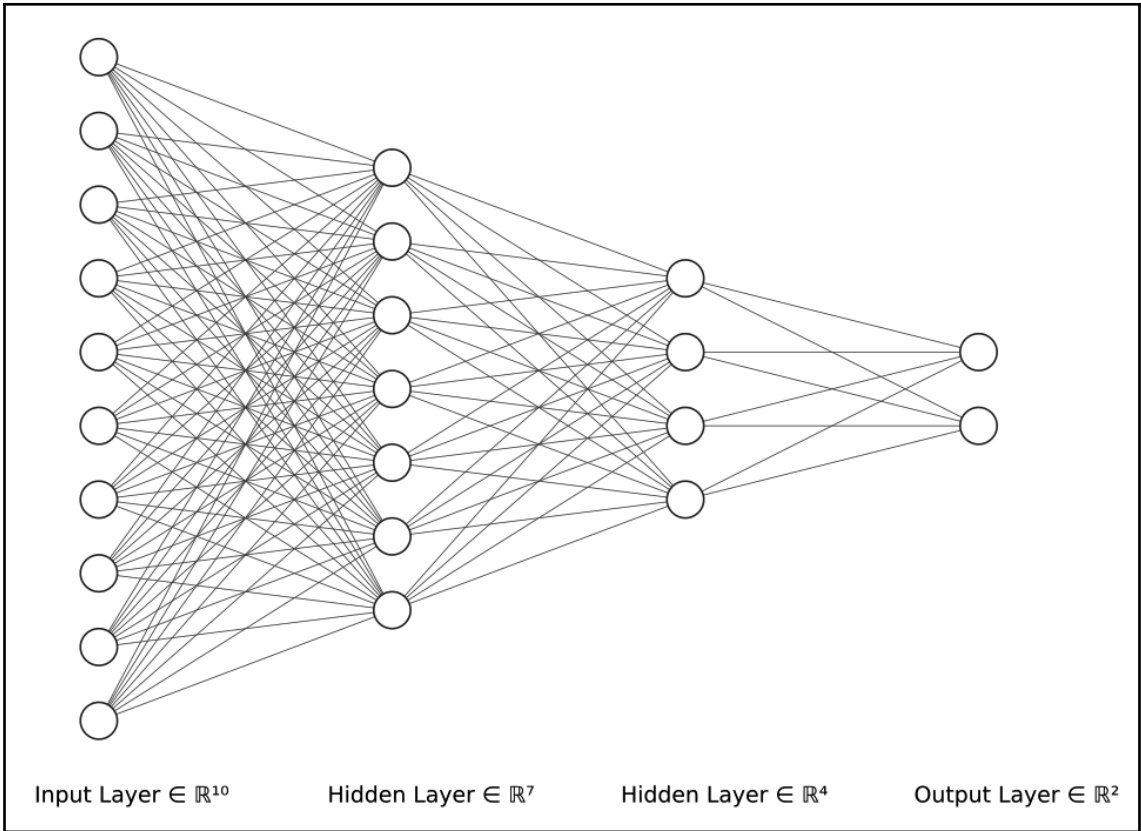


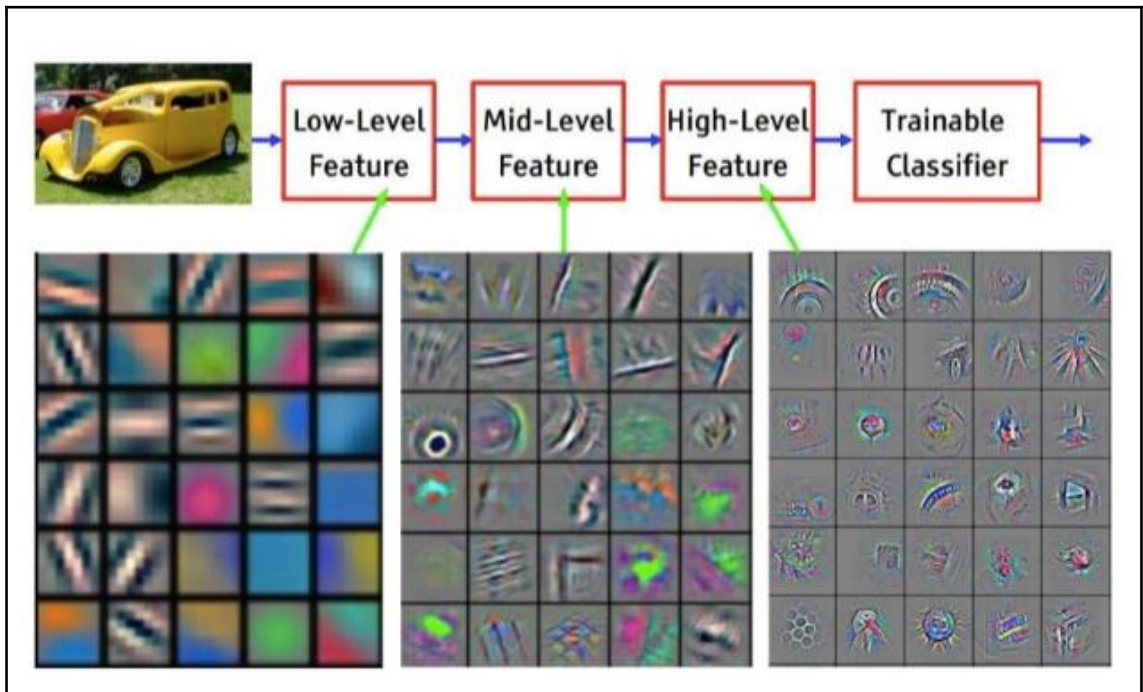
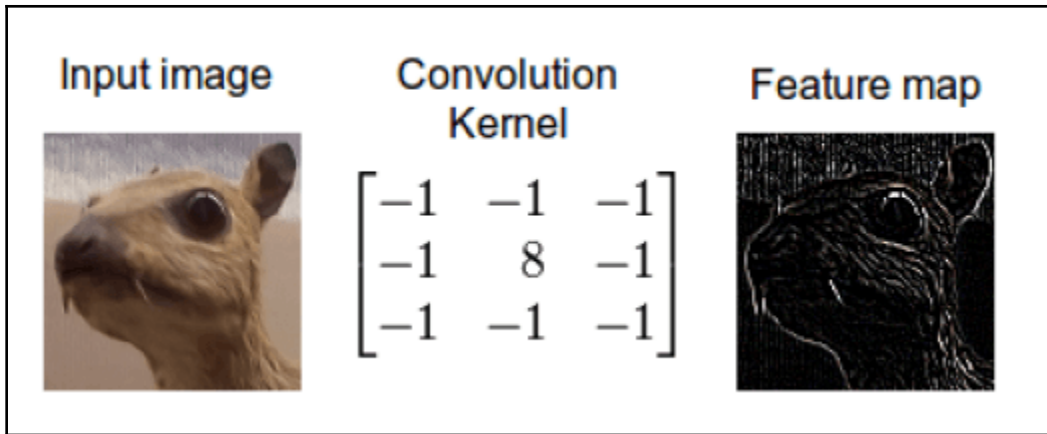


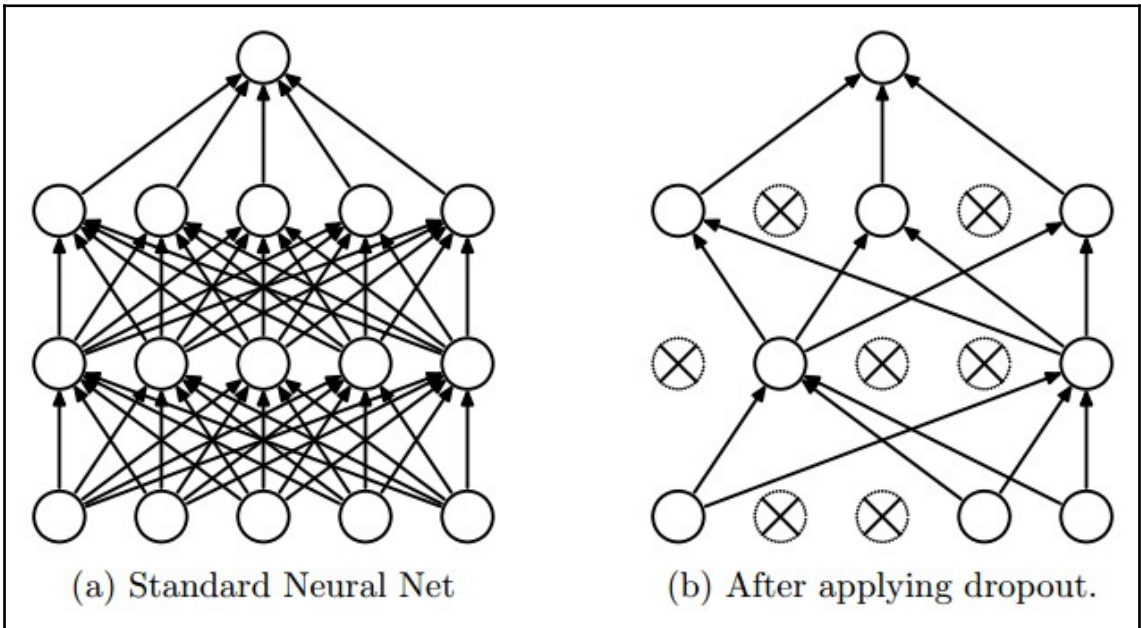
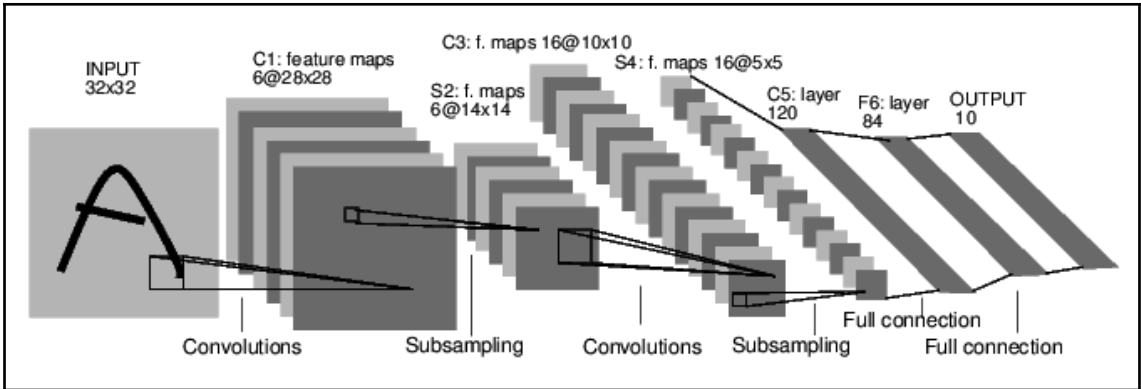


Chapter 2: Neural Networks and Deep Learning









Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1\dots m}\}$;

Parameters to be learned: γ, β

Output: $\{y_i = \text{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m} \sum_{i=1}^m x_i \quad // \text{ mini-batch mean}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m} \sum_{i=1}^m (x_i - \mu_{\mathcal{B}})^2 \quad // \text{ mini-batch variance}$$

$$\hat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \quad // \text{ normalize}$$

$$y_i \leftarrow \gamma \hat{x}_i + \beta \equiv \text{BN}_{\gamma,\beta}(x_i) \quad // \text{ scale and shift}$$

Algorithm 1: Batch Normalizing Transform, applied to activation x over a mini-batch.

Chapter 3: TensorFlow Graph Architecture

The graph shows a **MatMul** node (oval) receiving inputs from **Const** and **Const_1** nodes (circles). An arrow labeled 2×2 points from the **MatMul** node to an **Add** node (oval), which is highlighted with a red border. The **Add** node receives an input from **Const_2** (circle) and another from the **MatMul** node. The final output is labeled **result**.

result
Operation: Add

Attributes (1)
T {"type": "DT_FLOAT"}

Inputs (2)
○ MatMul 2x2
○ Const_2 1x2

Outputs (0)
Remove from main graph

The graph shows a **matmul** node (oval) receiving inputs from **Const** and **Const_1** nodes (circles). An arrow labeled 2×2 points from the **matmul** node to an **add** node (oval), which is highlighted with a red border. The **add** node receives an input from **Const_2** (circle) and another from the **matmul** node.

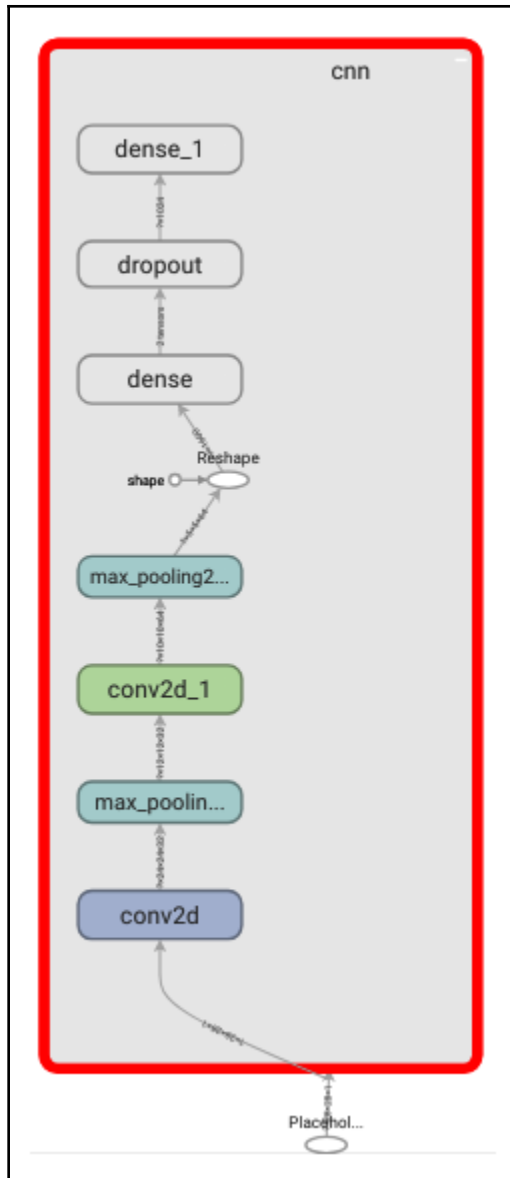
matmul
Operation: MatMul

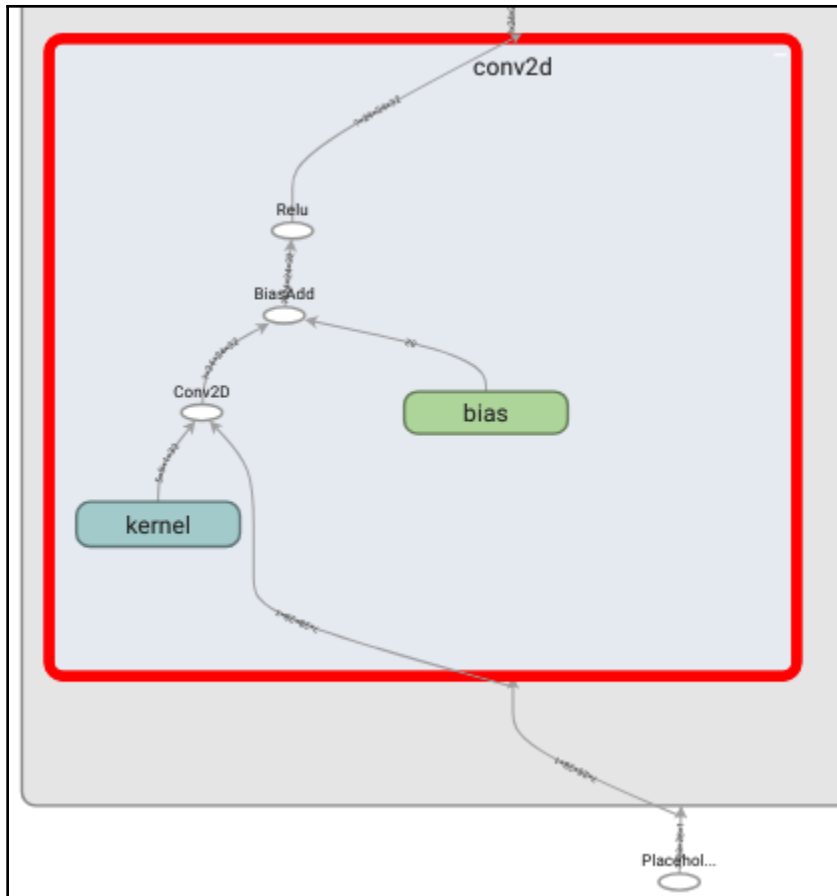
Attributes (3)
T {"type": "DT_FLOAT"}
transpose_a {"b": false}
transpose_b {"b": false}
Device /device:GPU:0

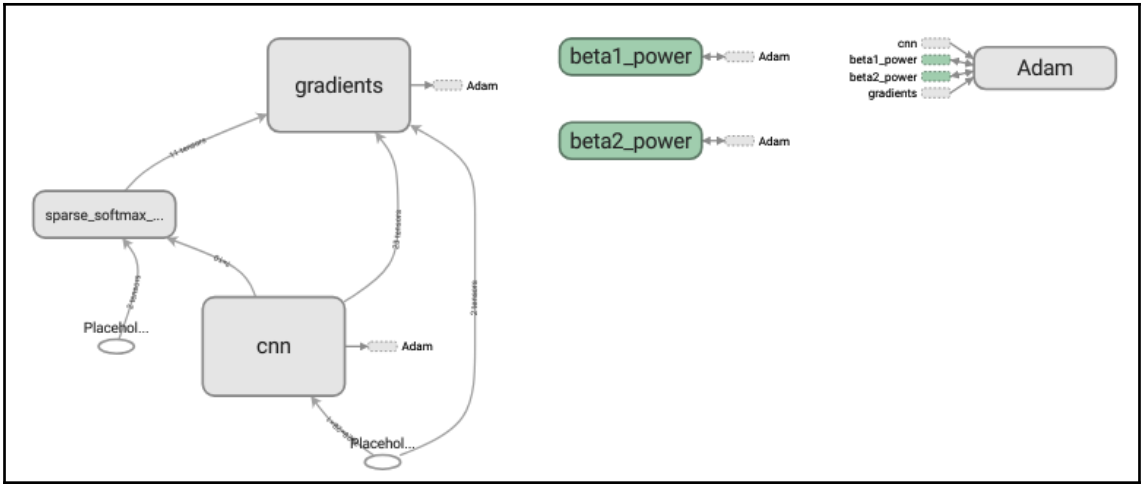
Inputs (2)
○ Const 2x2
○ Const_1 2x2

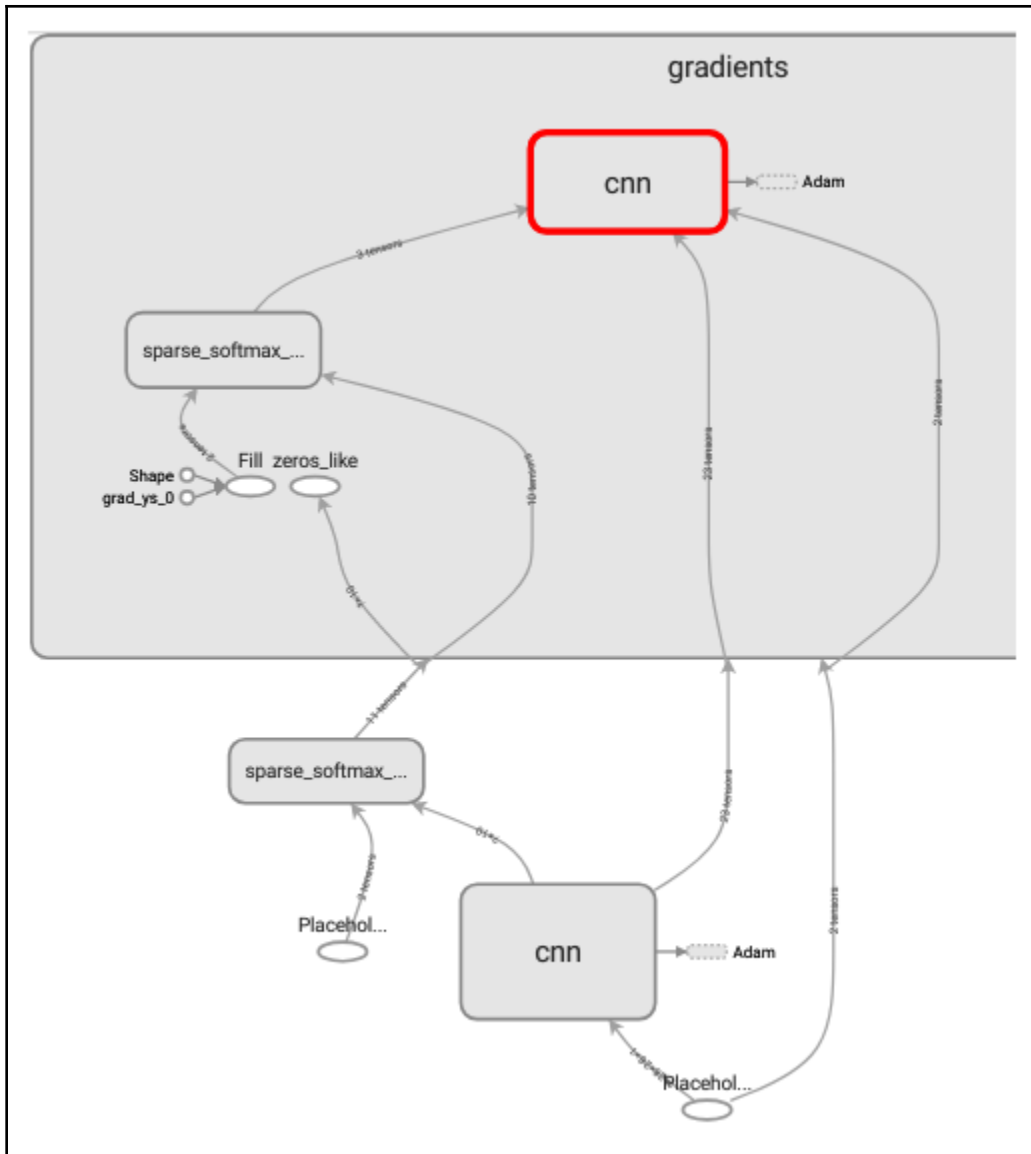
Outputs (1)
○ add 2x2

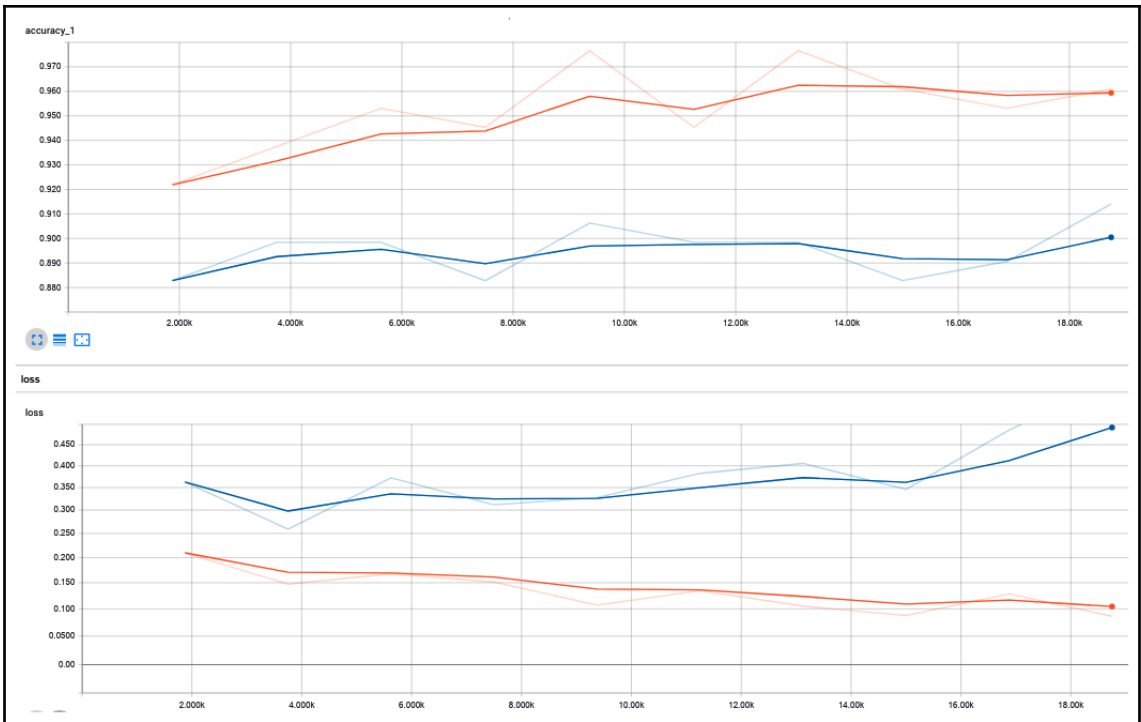
Remove from main graph

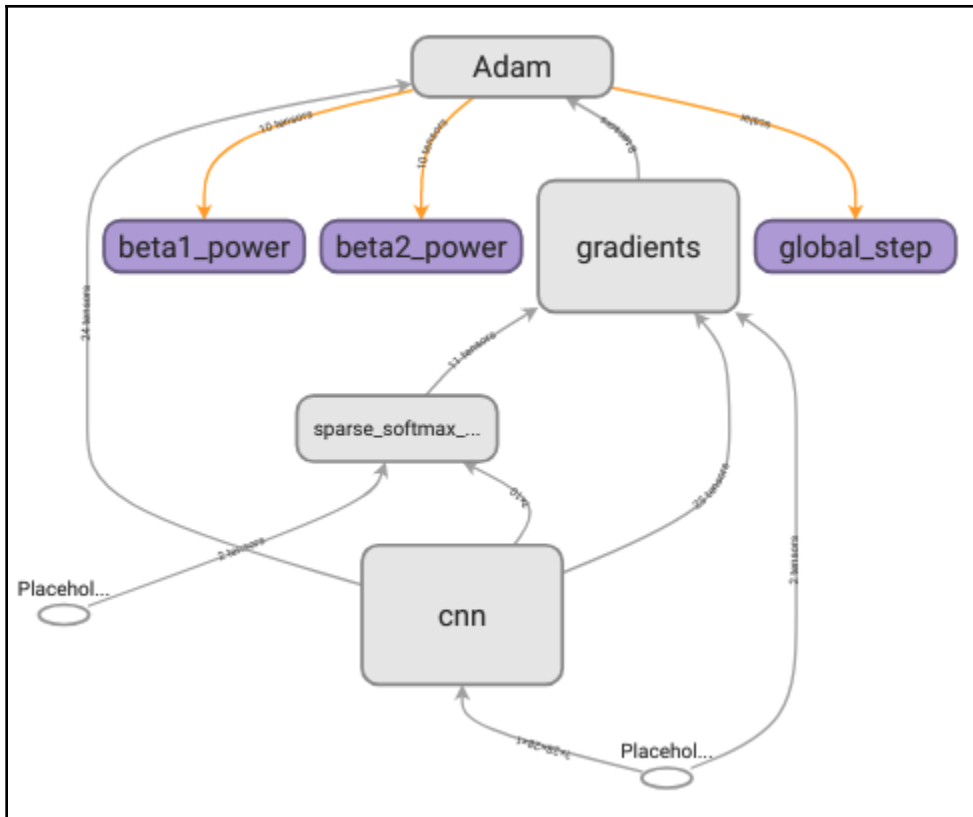




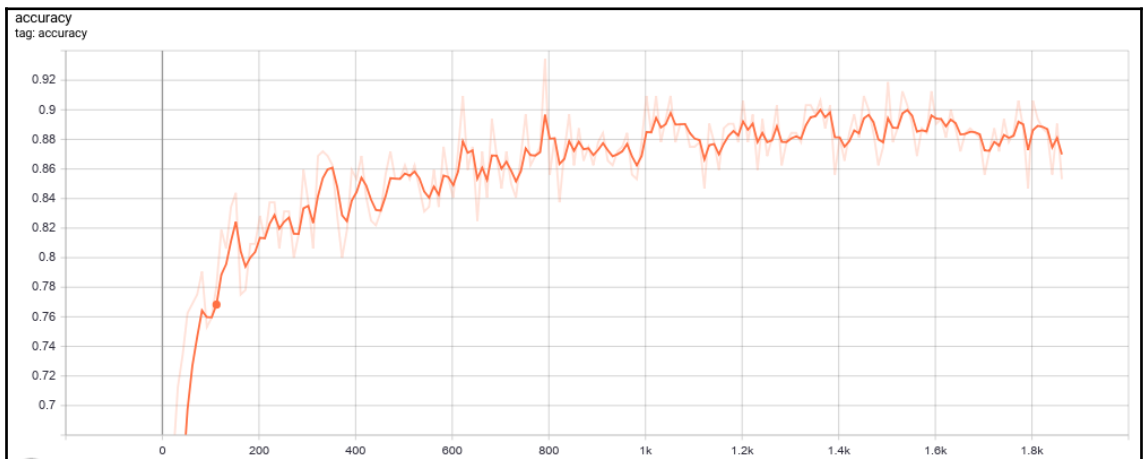
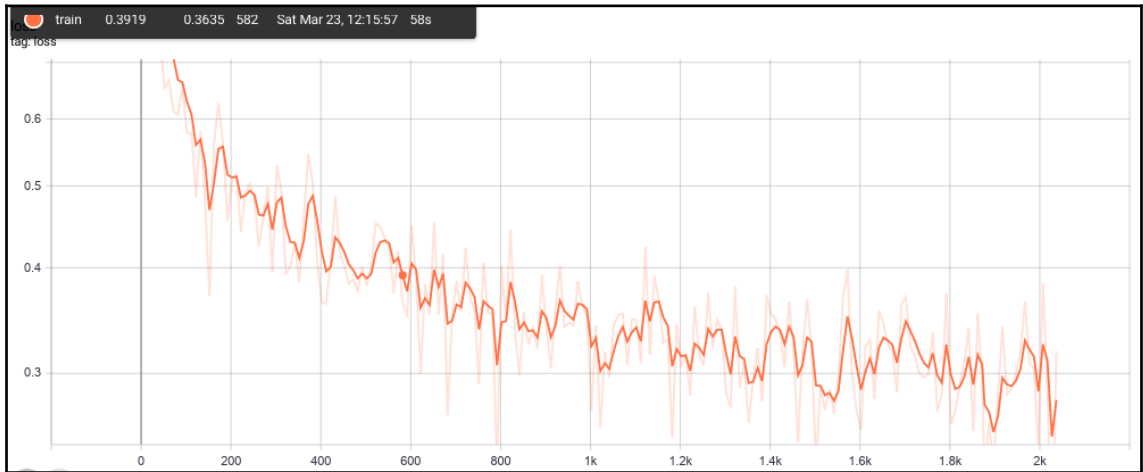


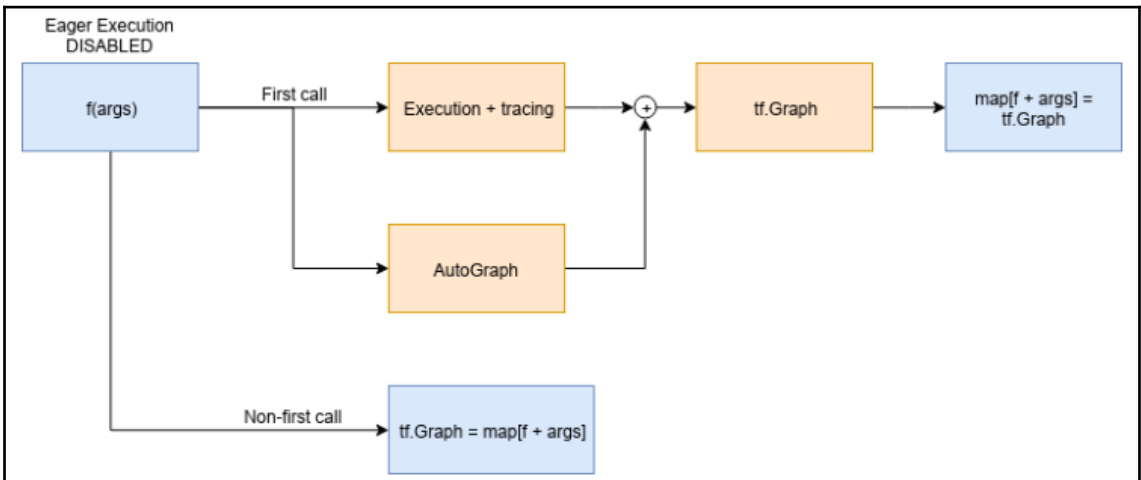
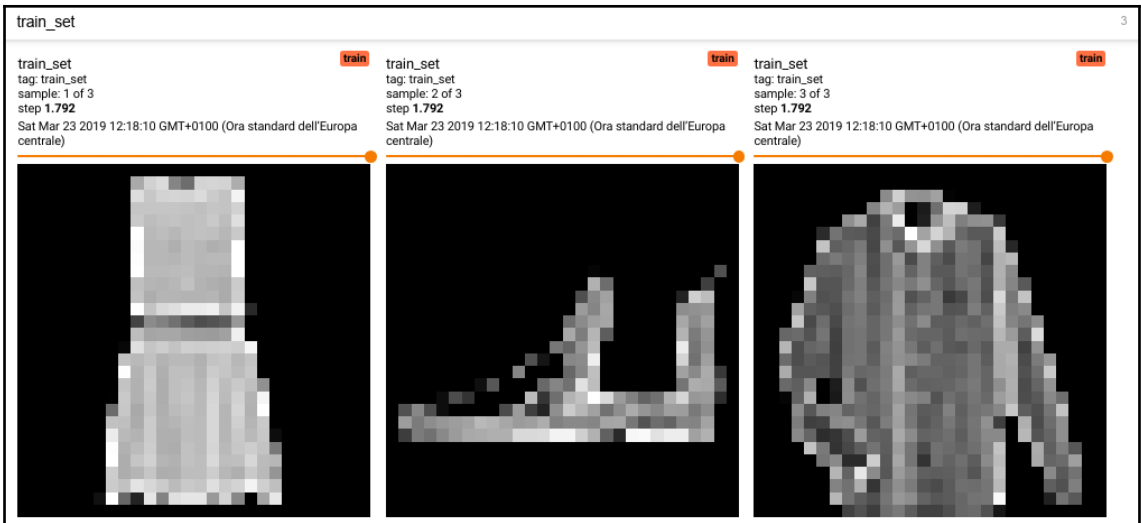




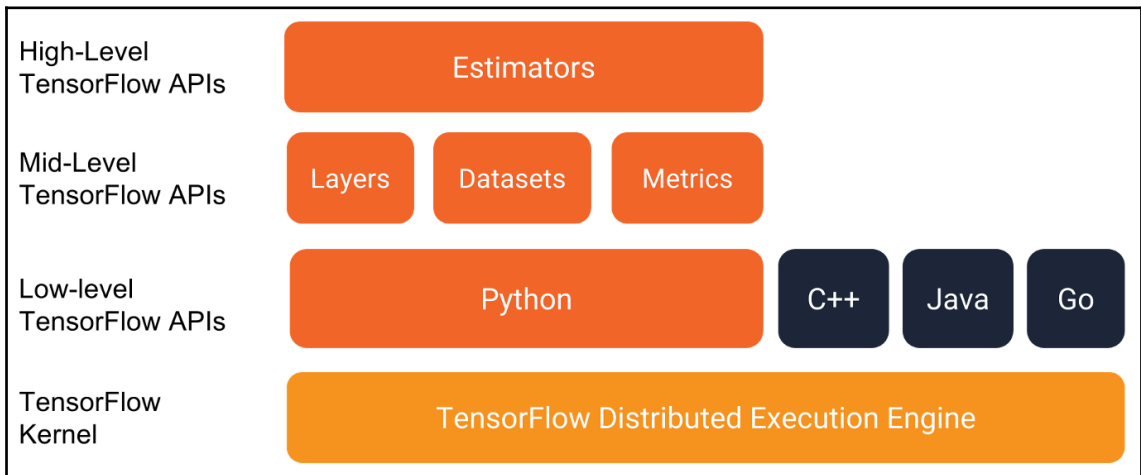
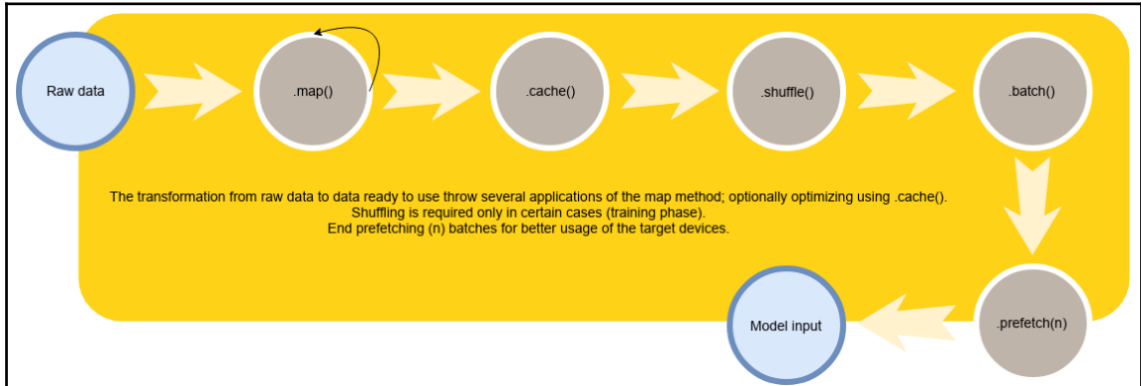


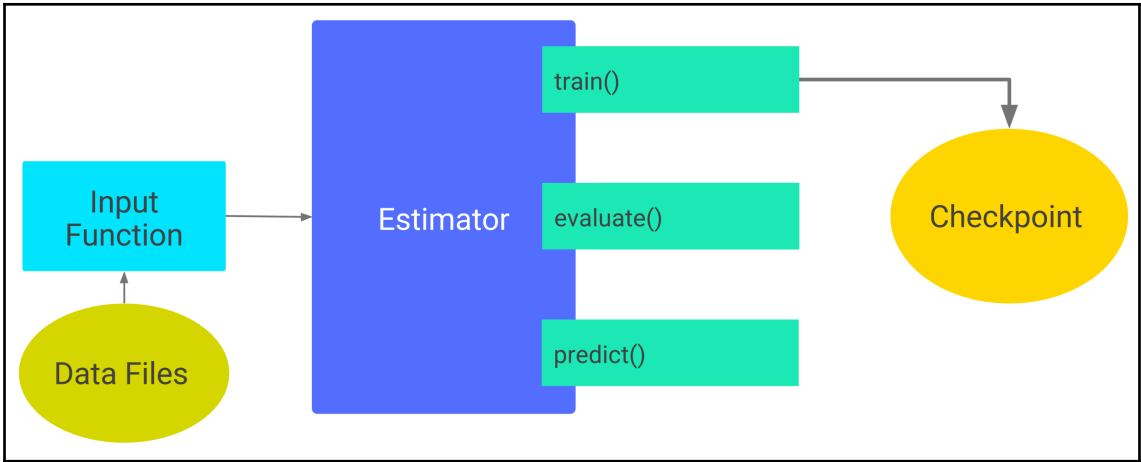
Chapter 4: TensorFlow 2.0 Architecture





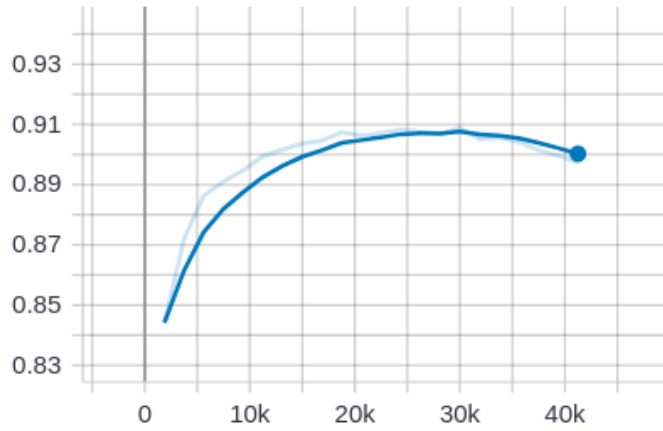
Chapter 5: Efficient Data Input Pipelines and Estimator API





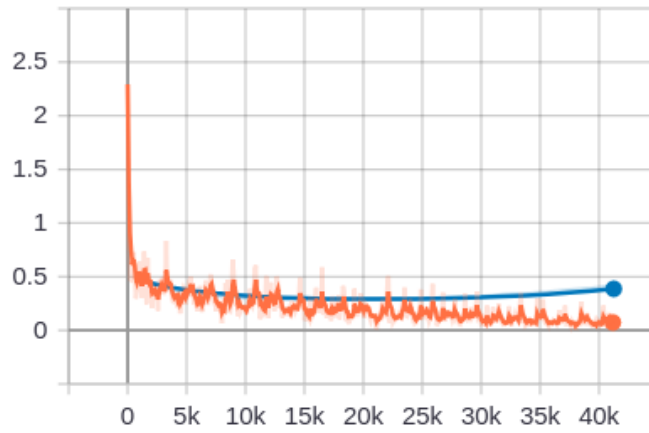
accuracy

accuracy

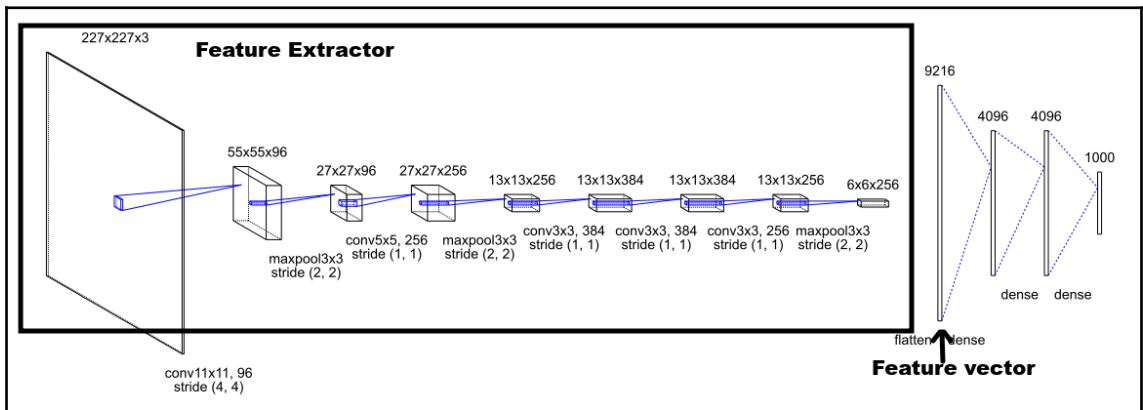


loss

loss



Chapter 6: Image Classification Using TensorFlow Hub



TensorFlow Hub tf2 Q [USER GUIDE](#)

Filter by

Language _____

Network _____

Publisher _____

Dataset _____

Module type _____


 **tf2-preview/mobilenet_v2/feature_vector** By Google

image-feature-vector ImageNet (ILSVRC-2012-CLS) MobileNet V2
[TF2] Feature vectors of images with MobileNet V2 trained on ImageNet (ILSVRC-2012-CLS).



 **tf2-preview/mobilenet_v2/classification** By Google

image-classification ImageNet (ILSVRC-2012-CLS) MobileNet V2
[TF2] Imagenet (ILSVRC-2012-CLS) classification with MobileNet V2.

 **tf2-preview/nnlm-en-dim128** By Google

text-embedding Google News NNLM English
Token based text embedding trained on English Google News 200B corpus.



 **tf2-preview/inception_v3/feature_vector** By Google

image-feature-vector ImageNet (ILSVRC-2012-CLS) Inception V3
[TF2] Feature vectors of images with Inception V3 trained on ImageNet (ILSVRC-2012-CLS).

 **tf2-preview/gnews-swivel-20dim** By Google

text-embedding Google News Other English
Token based text embedding trained on English Google News 130GB corpus.



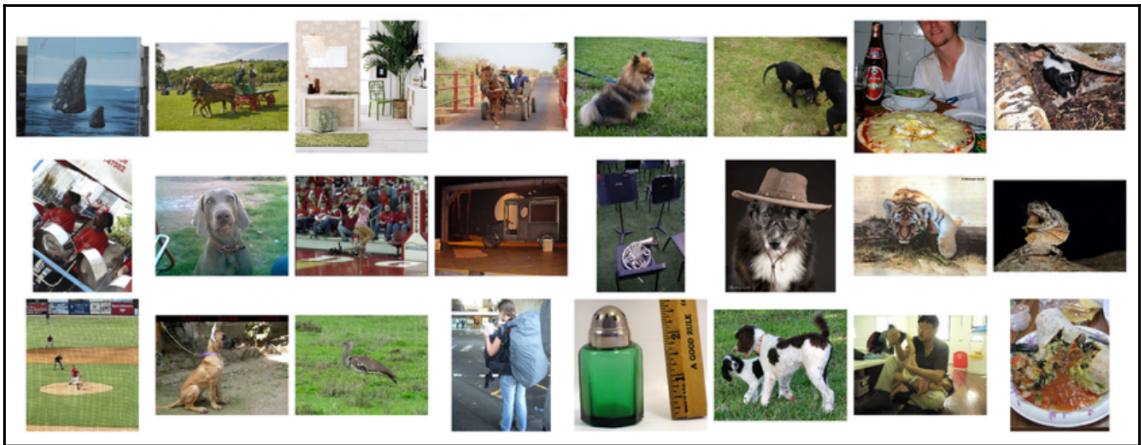
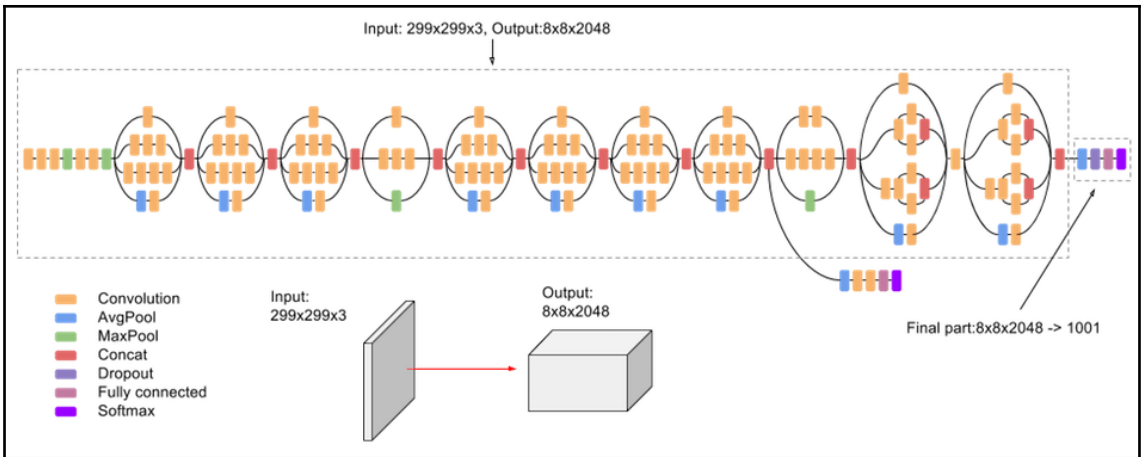
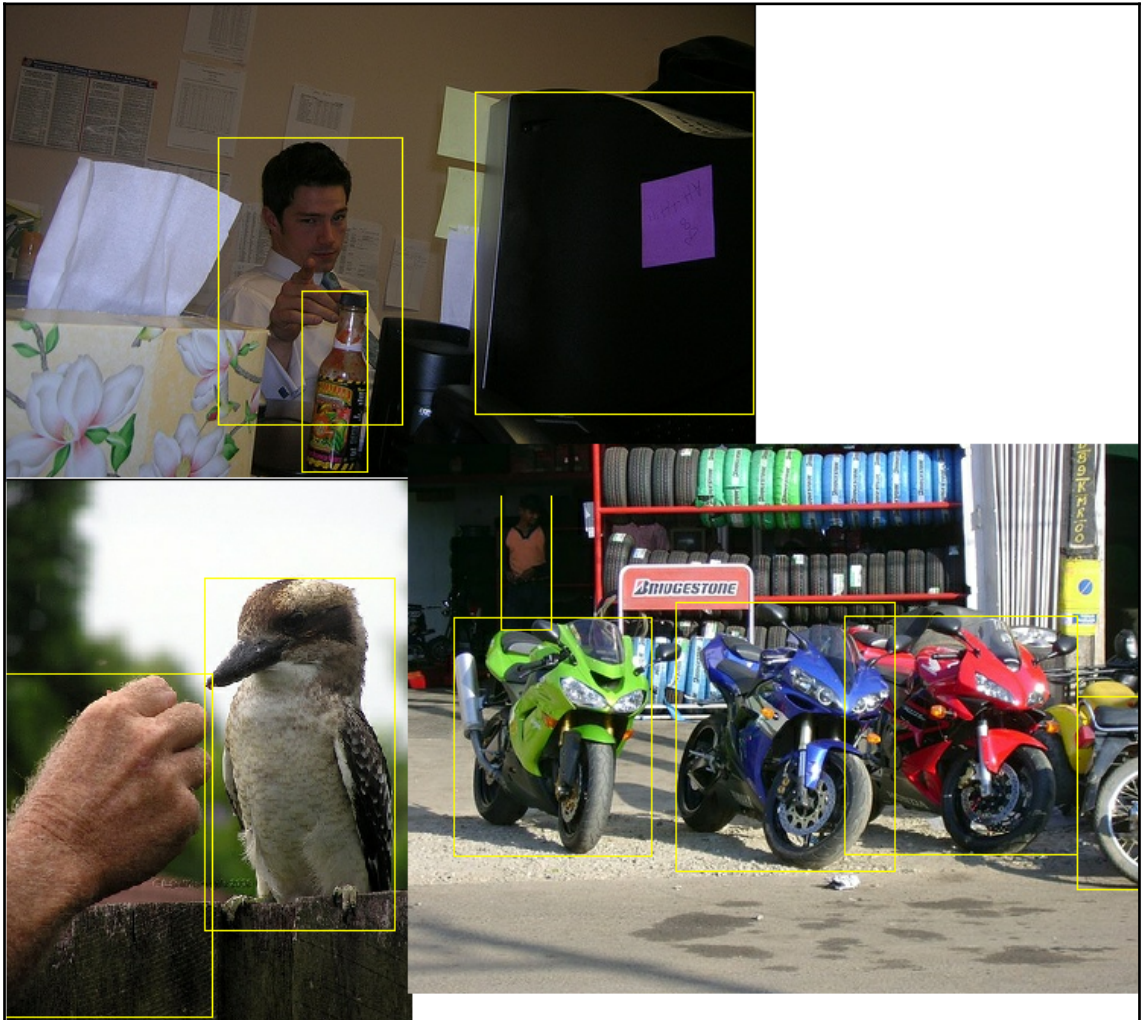
 **tf2-preview/inception_v3/classification** By Google

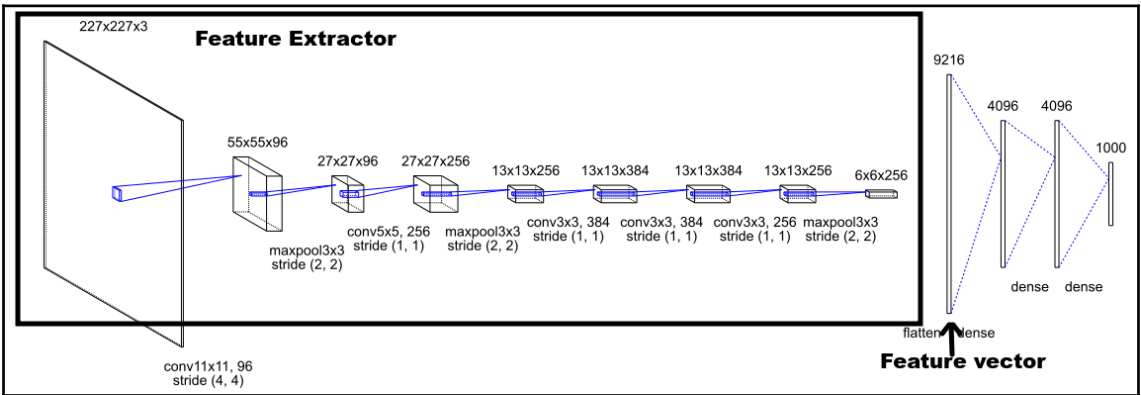
image-classification ImageNet (ILSVRC-2012-CLS) Inception V3
[TF2] Imagenet (ILSVRC-2012-CLS) classification with Inception V3.

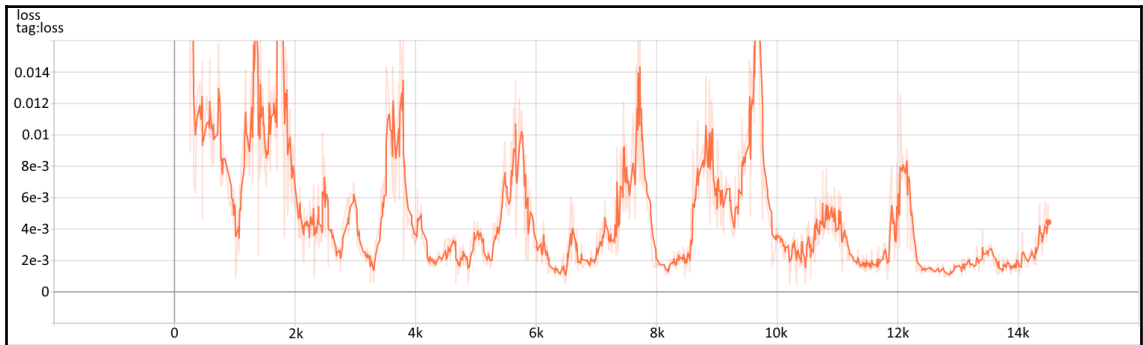
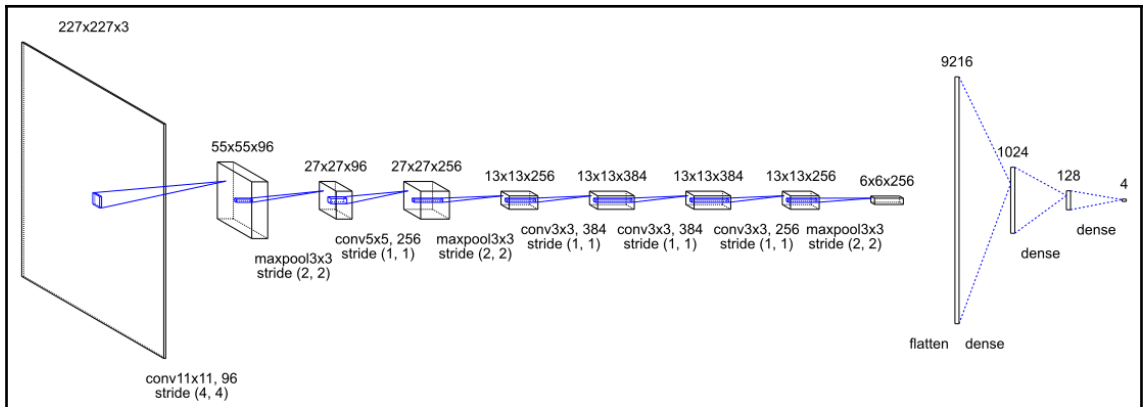
 **tf2-preview/nnlm-en-dim50** By Google



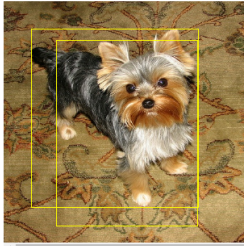
Chapter 7: Introduction to Object Detection







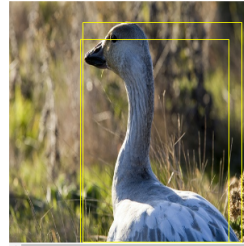
images
tag: images
sample: 1 of 3
step 14,500 Wed May 15 2019 10:36:15 GMT+0200 (Central European Summer Time)



train
images
tag: images
sample: 2 of 3
step 14,500 Wed May 15 2019 10:36:15 GMT+0200 (Central European Summer Time)



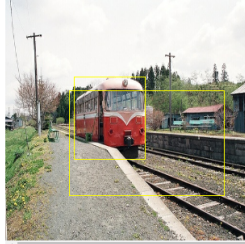
train
images
tag: images
sample: 3 of 3
step 14,500 Wed May 15 2019 10:36:15 GMT+0200 (Central European Summer Time)



validation
images
tag: images
sample: 1 of 3
step 14,500 Wed May 15 2019 10:36:15 GMT+0200 (Central European Summer Time)



validation
images
tag: images
sample: 2 of 3
step 14,500 Wed May 15 2019 10:36:15 GMT+0200 (Central European Summer Time)

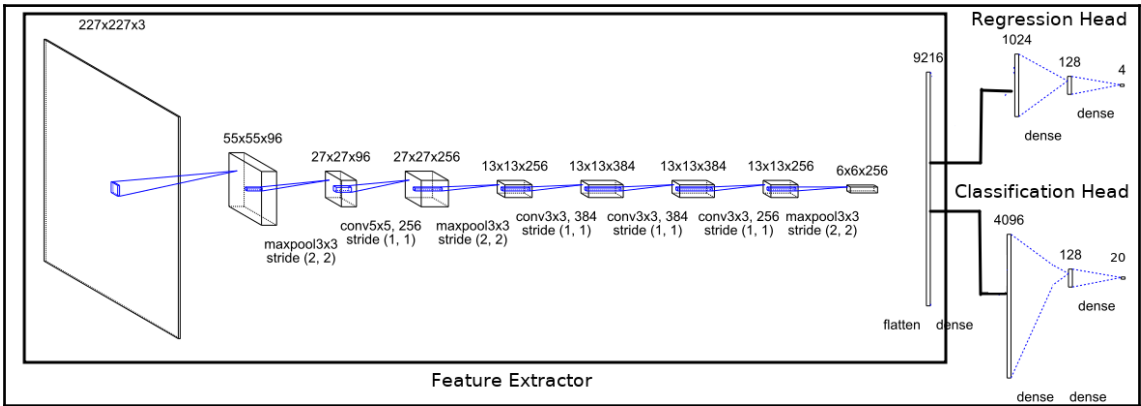


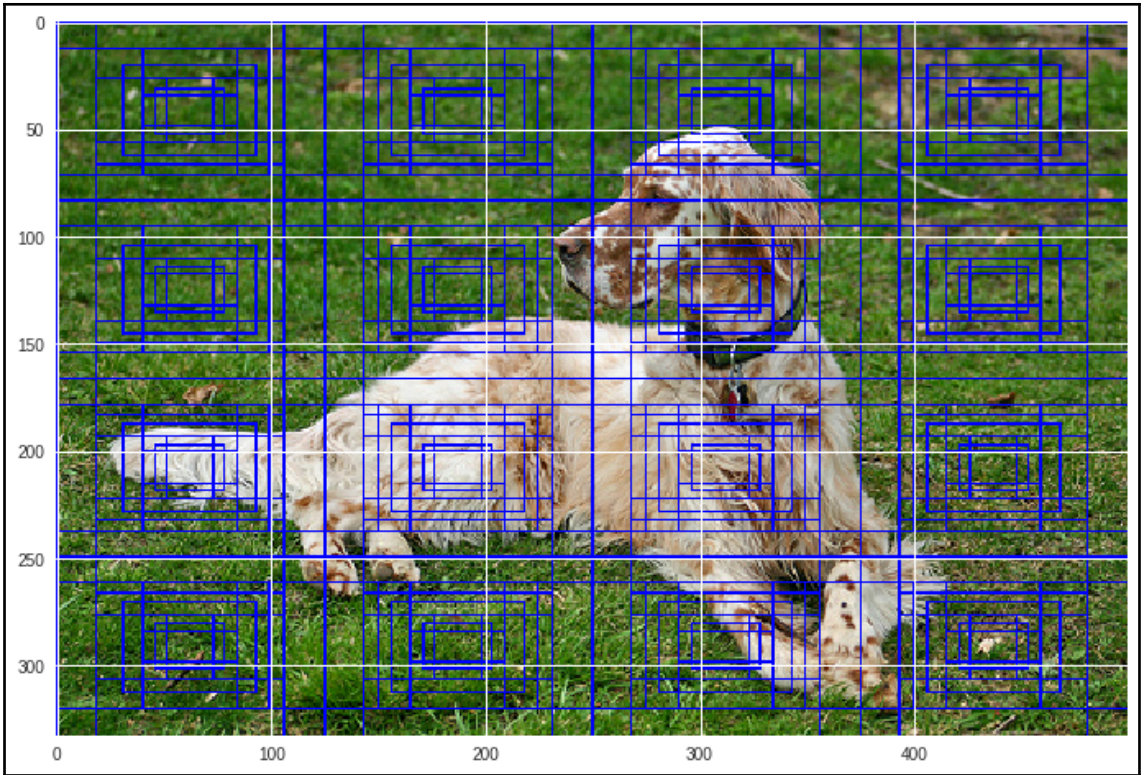
validation
images
tag: images
sample: 3 of 3
step 14,500 Wed May 15 2019 10:36:15 GMT+0200 (Central European Summer Time)

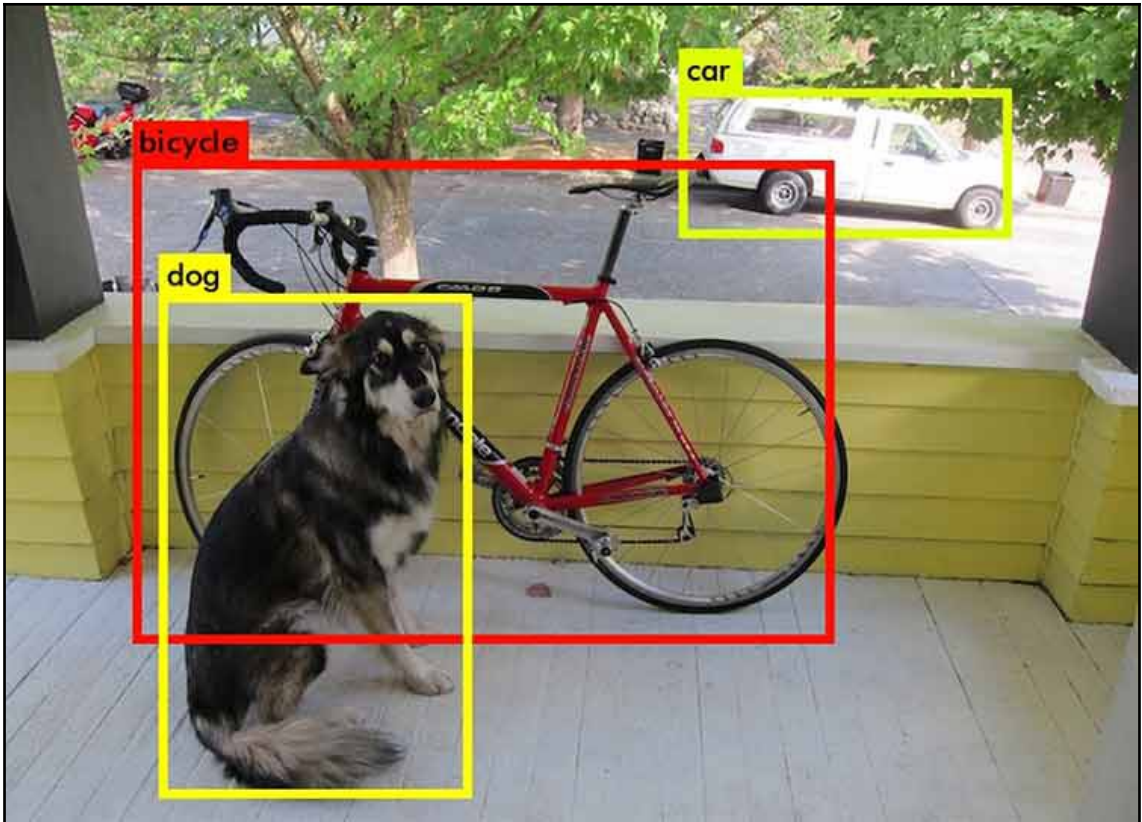




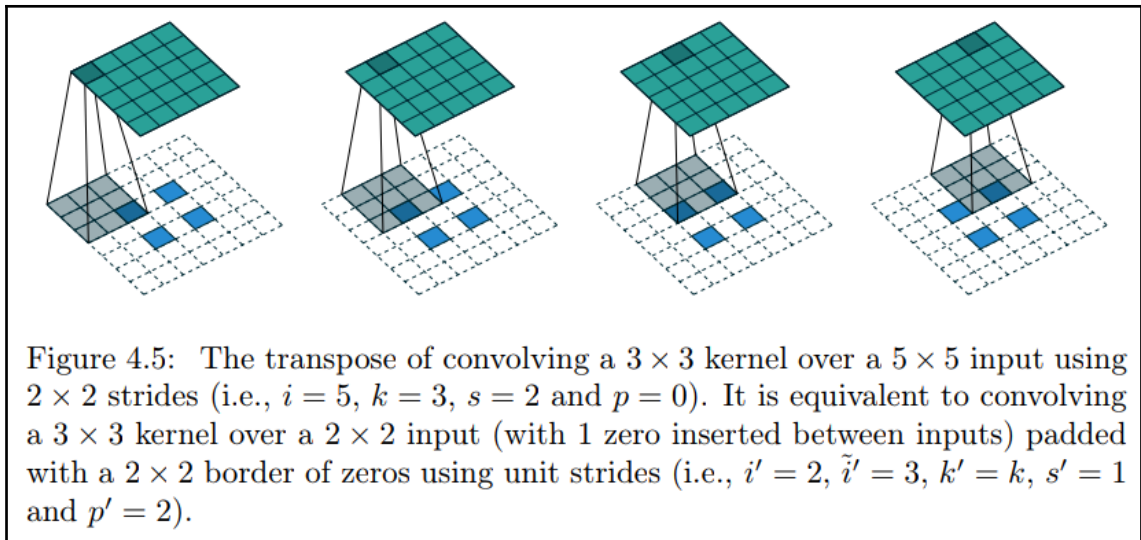
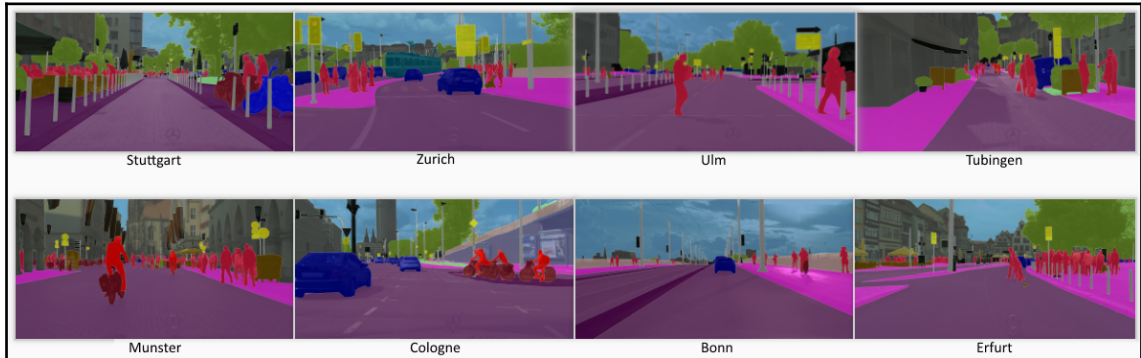
Ground truth
 Prediction

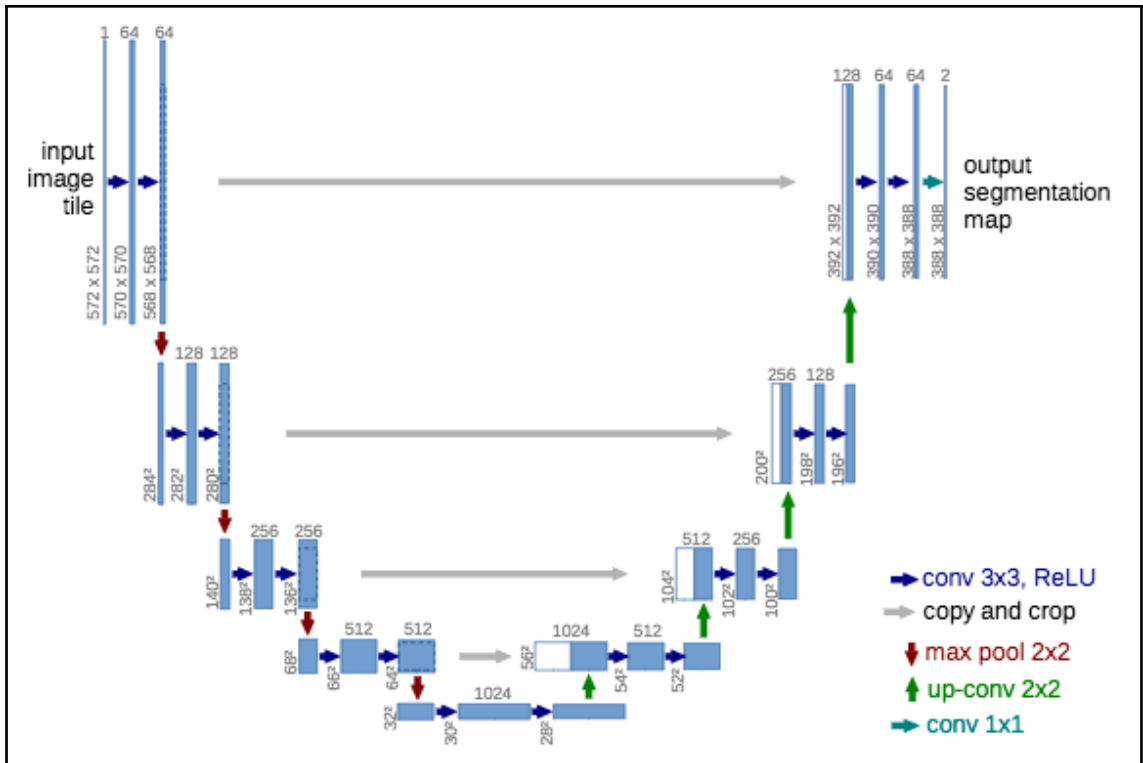


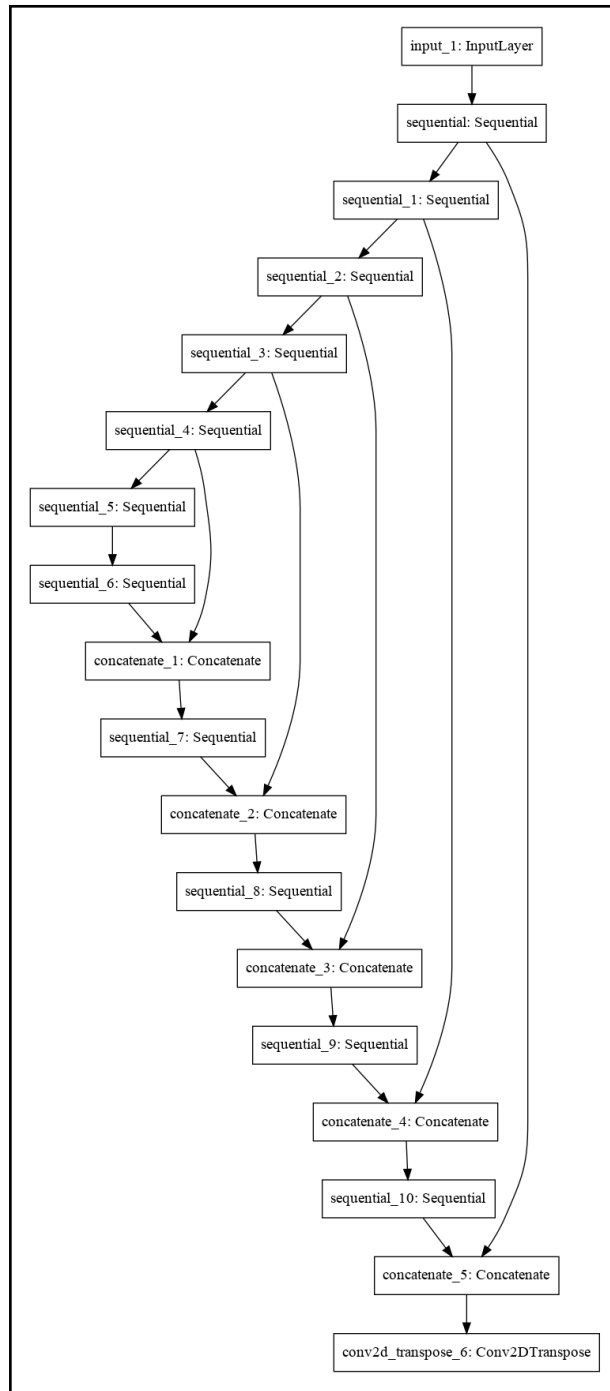


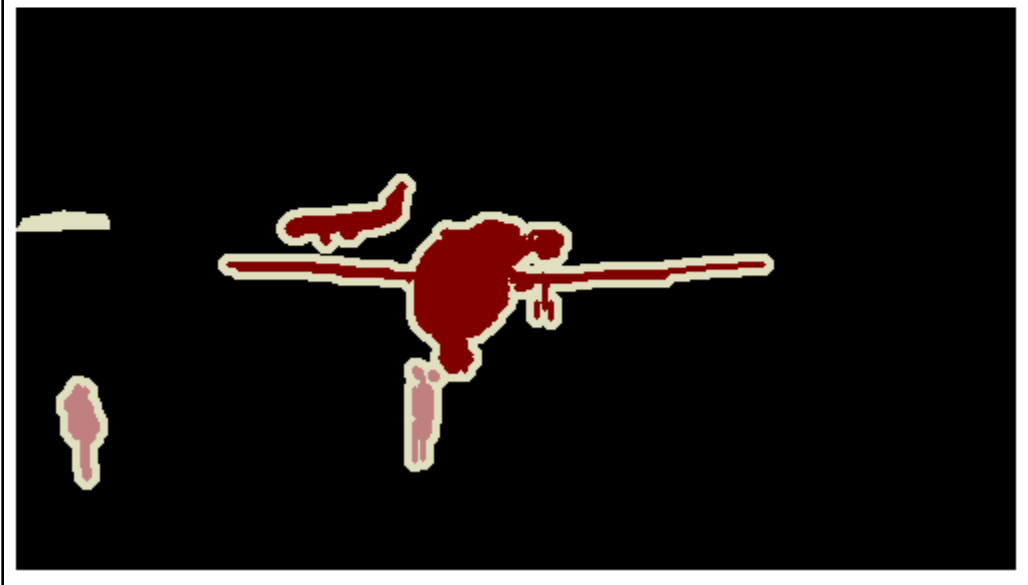


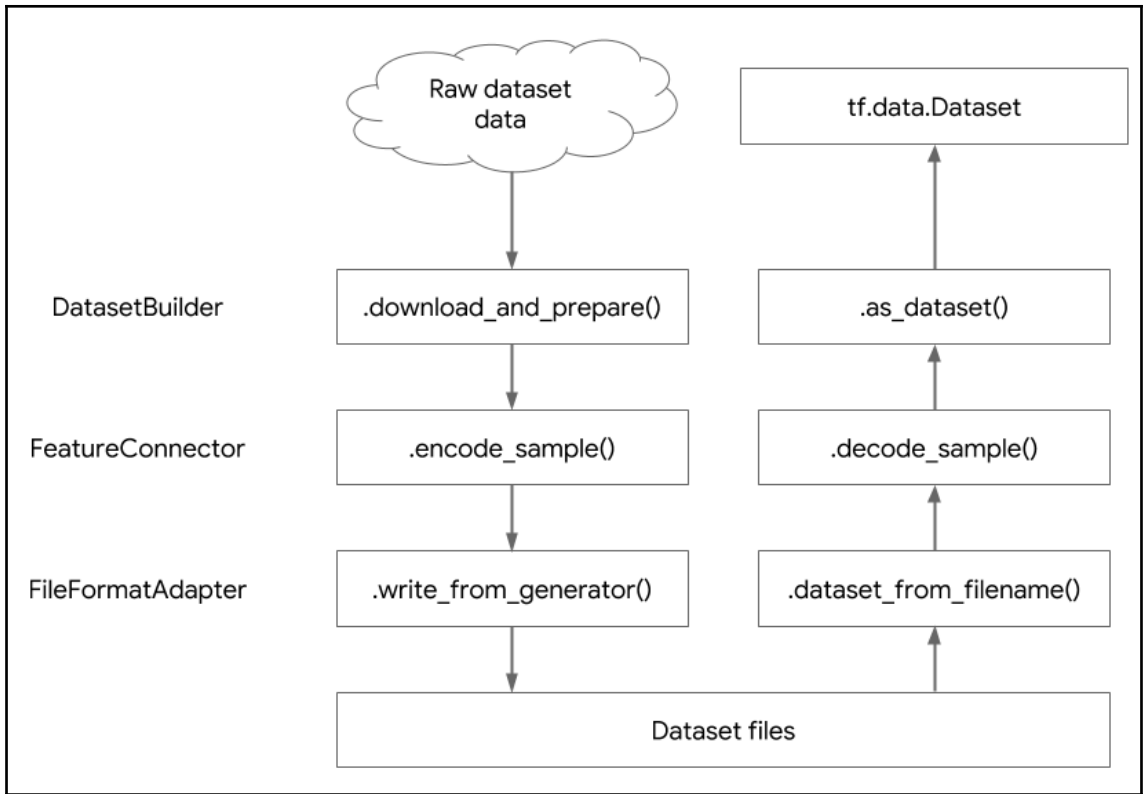
Chapter 8: Semantic Segmentation and Custom Dataset Builder

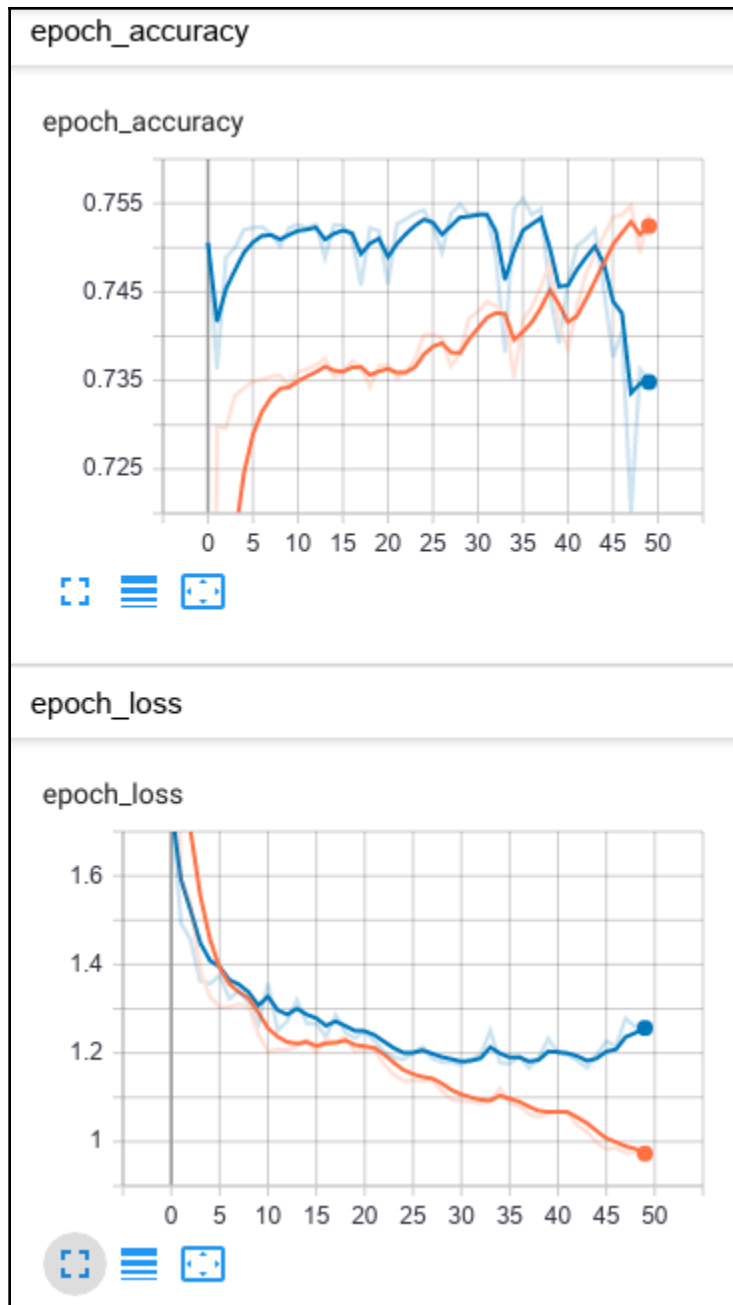








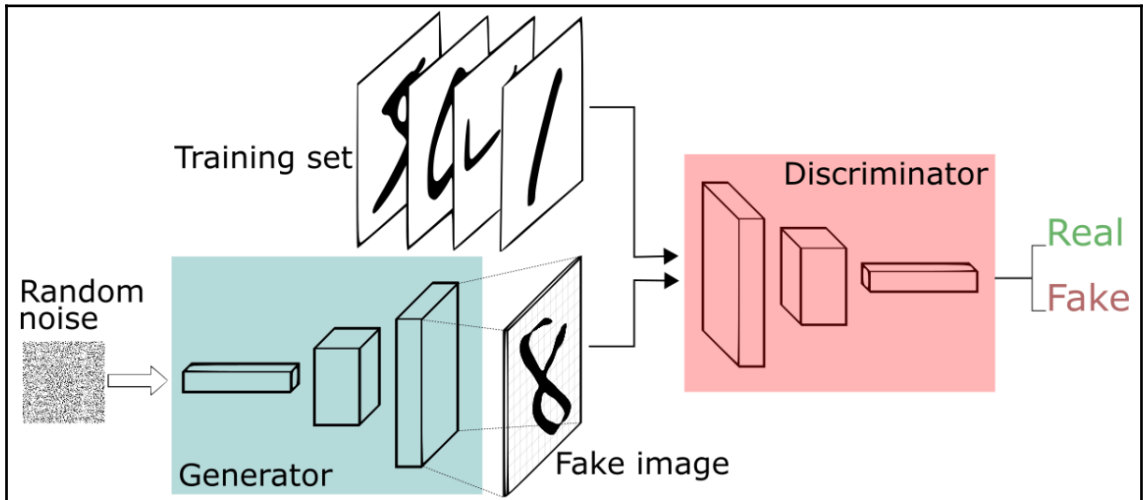


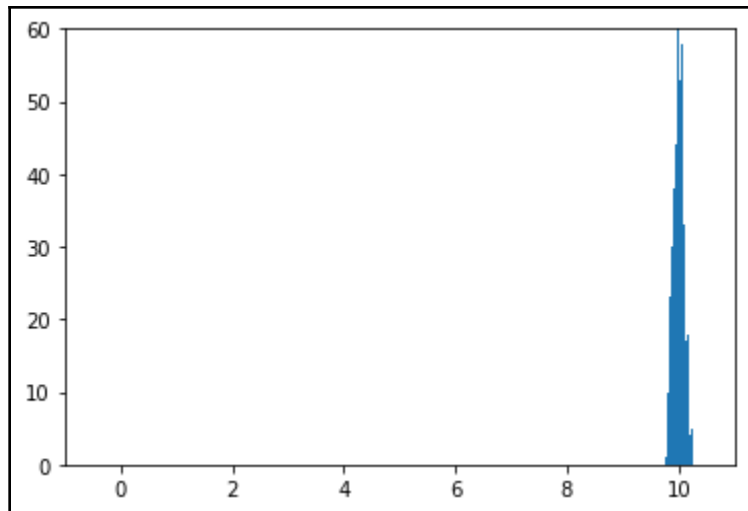
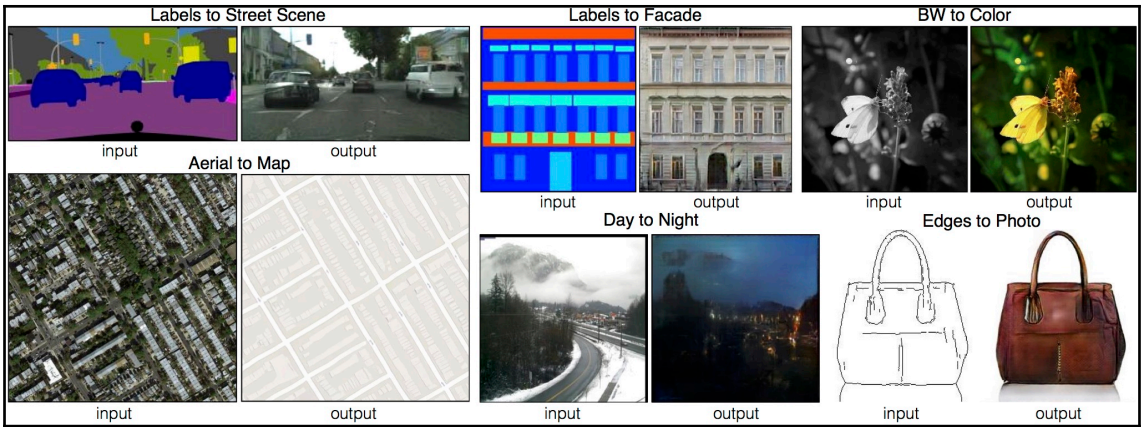


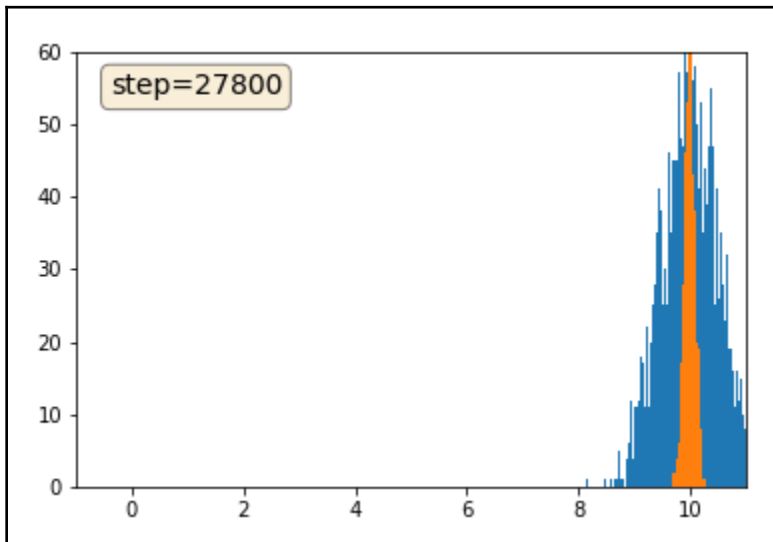
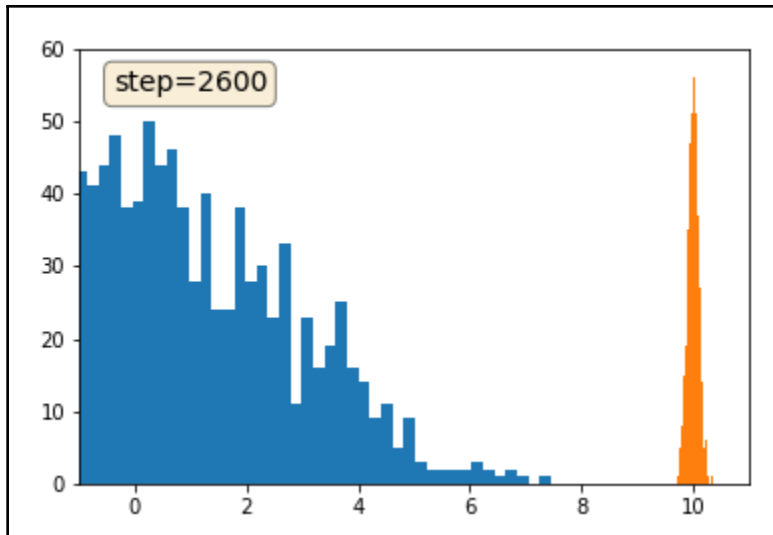


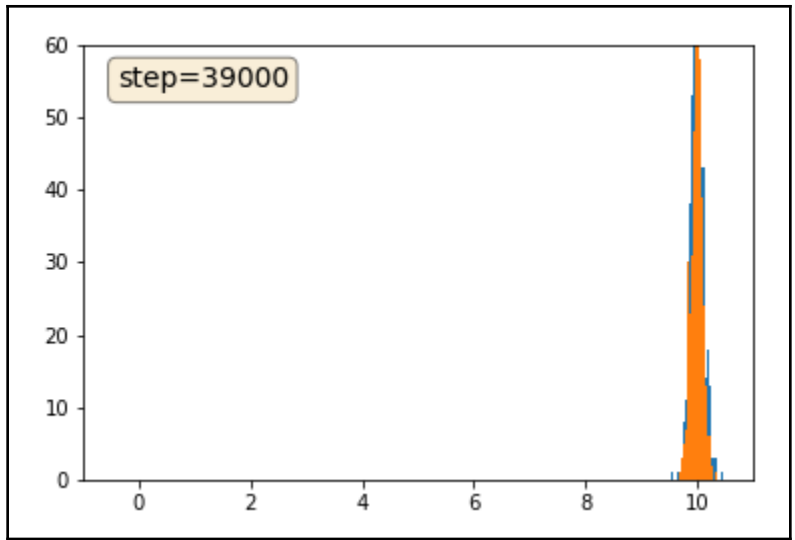


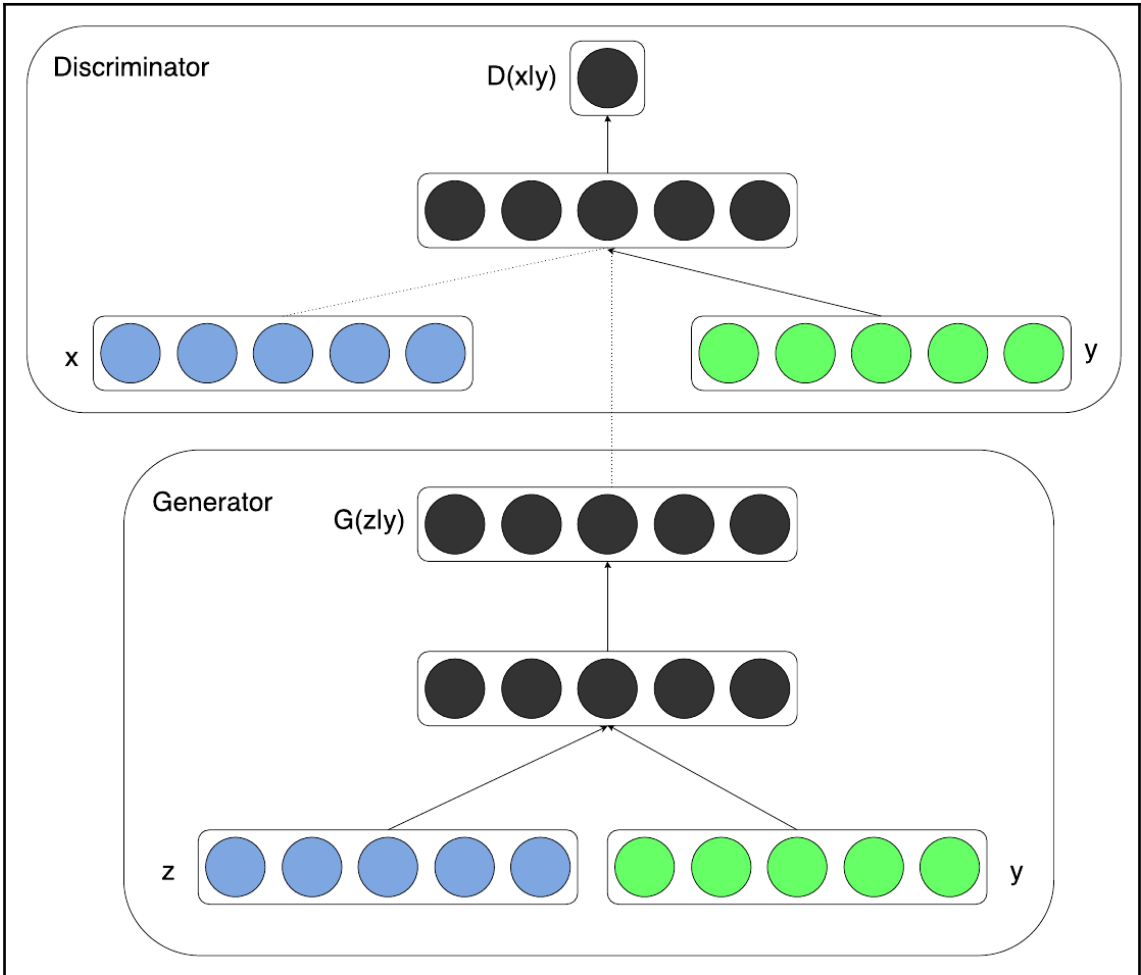
Chapter 9: Generative Adversarial Networks

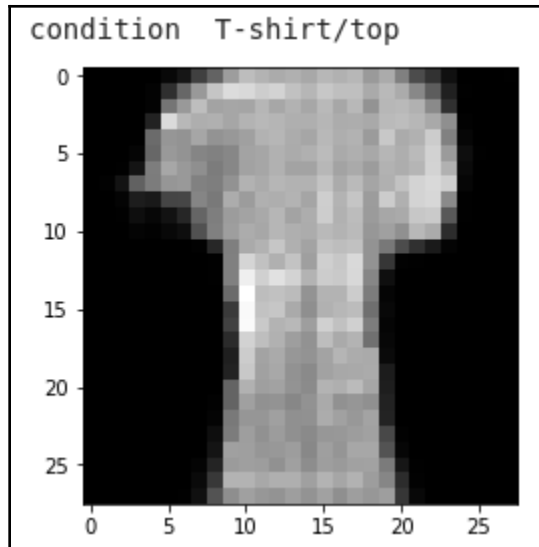




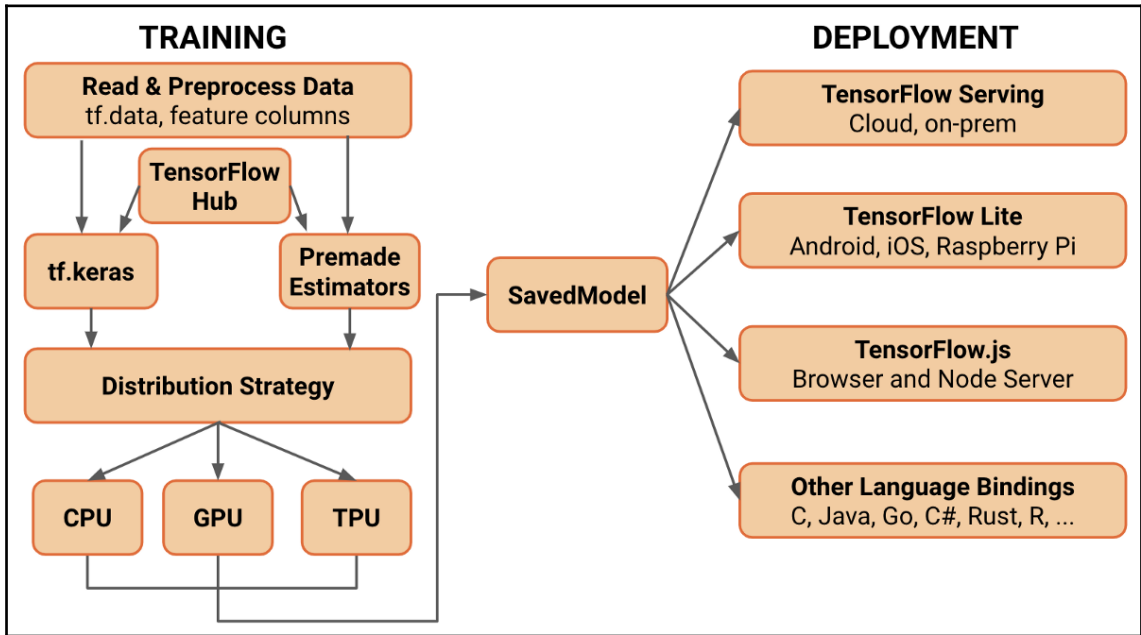








Chapter 10: Bringing a Model to Production



```
x: 5
y: 2
pow
xy: 32
yx: 24.999998092651367
```