Course: Deep Learning

Class: Setting the Scene

Subject: Discussion Topics

1. **Explain, with a clear example for each, the 5 V's of Big Data**

Volume: the amount of data created

Velocity: The speed at which data is generated, processed and analyzed.

Veracity: verifying the data is trustworthy

Value: the usefulness of the data

Variety: the different types and formats of data.

2. **When dealing with Big Data, one has to make architectural design decisions: briefly explain in this context the principles of horizontal versus vertical scaling of databases, sharding and replication, and the CAP theorem.**

In the beginning developers used monolithic architectures to make applications. When they got more users they upgraded the servers running the applications (Vertical Scaling). After, when they got even more users and they couldn't upgrade the servers anymore they started using multiple servers (Horizontal Scaling). Sharding means they split up the data into different databases. With replication you have a master database and some slaves, you can only write to the master database, this database is replicated to the different slaves. You can only read data from slave databases. The CAP theorem explains how you should make databases in terms of Consistency, Availability and Partition-Tolerance. It makes sure the database is balanced for different loads.

3. **Explain the concept of a Data Lake, versus a Data Warehouse. Make the distinction between on-prem versus cloud solutions. What is a Data Lake House, and a Data Mesh?**

A Data Lake is big repository where you can store all types of data, structured and unstructured. A Data Warehouse is a big repository optimized for structured data. A cloud solution means you use a cloud provider, where you store all the data. A on-prem solution means the company owns and maintains their own databases. A Data Lake House is a hybrid between the Data Lake and Data Warehouse, it means you have a Data Lake and you do some governance for the data in the lake. A Data Mesh solution means you have a data lake house, but the governance is split up to different owners, e.g. Sales, Marketing, etc.

4. **Where do the concepts of Hadoop, Spark, Qlik (BI), BigQuery (Data Analytics), and TensorFlow/PyTorch (Data Modeling) fit in when looking at the concepts of data lake / data warehouse?**

Data lake: Hadoop, Spark, TensorFlow/PyTorch.

Data warehouse: Qlik, BigQuery.

5. **Can you explain the (buzzword) concept of Data Governance in the context of Data Lakes? Can you link it to the Veracity part of Big Data (5V's of Big Data)? And to the concept of a Data Swamp?**

Data Governance means you have some rules in the data lake, this makes sure your data lake doesn't turn into a data swamp. In a data swamp the data isn't usable and too unstructured. With governance you can insure the Veracity of the data.

6. **With the rapid and explosive growth of AI, the principles of Trustworthy and Ethical AI have become major concerns. Can you explain what the principles of human in-, on-, or out the loop mean in this context?**

Human in the loop: A human can see everything the model is doing.

Human on the loop: A human checks the model regularly.

Human out the loop: There is no human that checks the model(autotomy)

7. **One of the cornerstones of Trustworthy AI, is explainability (XAI). Discuss what this concept of XAI embodies, and why it might be considered a challenge when dealing with deep learning**

It means you can explain what the model is doing. With deep learning there are many layers that nobody can explain (latent space), because the model is working in millions of dimensions.

Class: Deep Learning Adv

Subject: Discussion Topics

1. **A multilayer perceptron is considered an universal approximator. Explain this concept by explaining the 2 main parts of a neuron (the affine transformation, and the activation). How do tensors, and the power of GPU's, fit in this picture?**

A multilayer perceptron is considered a universal approximator due to its ability to approximate any continuous function given enough neurons and layers.

- The **affine transformation** in each neuron applies a linear operation to the input data using weights and biases.
- The **activation function** introduces nonlinearity, allowing the network to model complex, nonlinear relationships.
- **Tensors** allow neural networks to represent inputs, weights, and activations efficiently and handle operations in a scalable way.
- **GPUs** accelerate the training process by handling parallel computations needed for tensor operations, making it feasible to train large MLPs that serve as universal approximators.

2. **Explain how a neural network learns from labeled data (use these terms: forward pass, error/loss function, gradient descent, learning rate, backpropagation)**

Neural networks learn from labeled data by adjusting their parameters to minimize the difference between their predicted output and the true output.

- **Forward Pass**: The input data is fed through the network, and predictions are generated.
- **Loss**: The error between the predicted output and the true label is calculated using a loss function.
- **Backpropagation**: The gradient of the loss with respect to each weight is computed by propagating the error backward through the network.
- **Gradient Descent**: The weights are updated based on the gradients using the gradient descent algorithm.
- **Learning rate**: It controls how large the steps are that the models takes in the direction the backpropagation calculated.

3. **When optimizing our weights in a neural network, we try to find the minimum of the loss function, but we often get stuck in local minima. Discuss different ways to deal with this problem**

We need to use some randomness. The randomness makes sure we explore the loss landscape a bit more, so we don't get stuck in a local minimum. We can do this in a few different ways:

- Use batches, so the model doesn't have a full view of the loss landscape directly near the data point.
- Use different learning rates, so the model can skip over local minima.
- Use a momentum optimizer, so the model has a "momentum"

4. **We use the chaining rule to let the gradient of the loss function propagate backwards from the last layers to the first. In doing so, we encounter the problems of vanishing gradients and covariance shift. Briefly discuss these problems, and how we could fix these**

Vanishing gradient means some neurons stop training because the activation function doesn't fire. A good fix is to use ReLu or ReLu based activation functions.

5. **Deep Neural Networks are extremely flexible and can adapt to any kind of problem. So flexible, that we always run a major risk of overfitting. That's why we use regularization techniques: we want to keep our extreme flexibility, but we want to keep overfitting in check. Discuss different regularization techniques.**

1. Weight Decay: make sure all the weights are being used by penalizing big weights.
2. Early Stopping: stop training sooner rather than later, so the model can't overfit itself.

3. Dropout: sometimes random neurons are shut down so they can’t learn, this makes sure there are no “star” neurons. It makes sure all neurons are used the same.

4. Ensemble method:

6. **Adjusting the decision threshold in a binary classification model affects the confusion matrix. Discuss the trade-offs between precision and recall when changing the threshold. How does this impact the overall performance of the model?**

   If your threshold favors precision it will predict more false negatives than false positives. If your threshold favors recall, it will predict more false positives than false negatives.

7. **Discuss why the Precision-Recall (PR) curve is more informative than the ROC curve for evaluating models on unbalanced datasets. Define precision and recall, and explain how the PR curve helps in understanding the performance of a classifier in such scenarios. What does the area under the PR curve indicate?**

   The Precision-Recall curve is better for unbalanced datasets because you can choose how you predict. You can choose to be more precise, (rather have false negatives than false positives) or be “safe”, (rather have false positives than false negatives). The area under the PR-curve tells you how good your classifier actualy is.

Class: Computer Vision

Subject: Discussion Topics

1. **Explain why a fully connected feed forward neural network is not very well suited to model a computer vision task. And how a convolutional neural network (cnn) solves these problems.**

   You have to much inputs/parameters. CNN is better because it already searched features. And they use smaller filters to convolute the images.

2. **Explain the concept of a convolutional operation using the terms ‘kernel’, ‘local receptive field’, ‘stride’, ‘padding’, ‘feature map’, and ‘2D convolution’. Can you think of any use cases of a 3D convolution?**

   The kernel is a small matrix of weights that slides over the input image.

   The local receptive field refers to the specific region of the input image that a particular neuron in a convolutional layer is connected to.

   Stride refers to the number of pixels the kernel moves across the image during the convolution operation.

   The padding adds extra pixels around the border of the image.

   A feature map is the output of the convolution operation after applying a kernel to the input image.

2D convolution refers to the convolution operation performed on two-dimensional data, such as images. The kernel slides over the height and width of the input image, computing the dot product at each position.

3D convolution: Video, AR and VR

3. **Explain a typical CNN architecture for image classification (convolution + relu layers, pooling layers, flatten or GAP layer, fully connected feed-forward layer, batch normalization layer, and a softmax layer to do cross-entropy loss calculations)**

   The first part are convolution layers, layers where the network tries to find the best convolutions for feature extraction. Convolutions are filters that “slide” over the image and highlight certain patterns. The relu layers are the actual neurons behind the convolution layers. Pooling layers make sure you don’t have parameter explosions by pooling the output of the layer in front. This means they downsize the images. The Flatten or GAP layer flattens the output from a layer to a 1D vector. A fully connected feed-forward layer is layer used to do object detection, classification. A batch normalization layer makes sure are the weights are normalized around the sweet spot of the neurons in the network and it stops neurons from becoming “Star players”. A softmax layer is a layer on the end of the network that makes the output readable to humans. It predicts each class relative to eachother.

4. **Explain the concept of transfer learning to do computer vision modelling, and how we go about training such a network. Focus on the difference between feature extraction and fine tuning. Why was Imagenet such a big game-changer in this context?**

5. **The Imagenet Large Scale Visual Recognition Challenge caused some real innovative shifts in the deep learning CNN community. Explain following innovative architectural enhancements: the ReLU’s and Dropout from Alexnet, the use of smaller and smaller kernels from VGG (why is this better?), and combining small and larger kernels with the identity convolution into a module, from Inception/GoogLeNet, and finally the use of skip connections from resnet and densenet architectures.**

6. **Explain some advanced tweaking options from the fastai library, such as the learning rate finder, discriminative learning rates on top of the one cycle learning rate, and the use of fp16 mixed precision to optimize learning. And why is the ‘timm’ library such a useful addition to the standard fastai options?**

7. **Explain how convolutional neural networks for vision might be applied in non-traditional domains, such as sound classification, fraud detection, and malware detection. Would you still advise to use a pretrained resnet (trained on imagenet data) to be used in this context?**

Class: Computer Vision – Object Detection

Subject: Discussion Topics

1. Discuss the evolution of object detection methods from naïve template matching to HOG features and SIFT. What were the main limitations of these methods, and how did each subsequent method attempt to address them?

   Object detection evolved from basic template matching, which compared image patches to predefined templates but was sensitive to changes in scale, rotation, and lighting, to more advanced methods like Histogram of Oriented Gradients (HOG) and Scale-Invariant Feature Transform (SIFT). HOG improved detection by focusing on local gradient orientations, making it more robust to lighting changes and small positional shifts, though it still struggled with larger variations in scale and rotation. SIFT went further by detecting keypoints and computing descriptors that were invariant to scale and rotation, but it remained computationally intensive and less suited for real-time or dense object detection tasks. Each method aimed to address the limitations of its predecessor, paving the way for more robust and efficient approaches.

2. How do Convolutional Neural Networks (CNNs) improve feature extraction in object detection compared to earlier methods? Discuss the roles of regression and classification heads in these networks.

   Convolutional Neural Networks (CNNs) significantly improve feature extraction in object detection by learning hierarchical and abstract features directly from data, unlike earlier methods like HOG and SIFT that rely on manually designed features. CNNs automatically capture low-level to high-level patterns, making them more robust to variations in scale, rotation, and illumination. They extract features that preserve spatial relationships important for object detection, and their end-to-end training optimizes both feature extraction and task-specific objectives. In object detection, CNNs use classification heads to predict object classes and regression heads to determine precise bounding box coordinates, enabling them to accurately identify and localize objects in images.

3. What are the main challenges in object detection, such as scale, aspect ratio, and occlusion? How do techniques like Region of Interest (ROI) identification and non-maximum suppression help mitigate these challenges?

   Object detection faces challenges such as scale variations, aspect ratio differences, and occlusion, which complicate accurate object identification. Small and large objects, varying shapes, and partially hidden objects make detection difficult. Techniques like Region of Interest (ROI) identification help by focusing on regions likely to contain objects, improving efficiency and allowing the model to handle diverse object sizes and shapes. Non-Maximum Suppression (NMS) reduces redundant detections by selecting the most confident bounding box when multiple overlap, helping address issues like occlusion and cluttered scenes, ensuring clearer, more accurate object localization.

4. Compare and contrast two-stage detectors (e.g., R-CNN family) and one-stage detectors (e.g., YOLO). What are the advantages and disadvantages of each approach?

5. Discuss the importance of evaluation metrics in object detection. How do metrics like Intersection over Union (IoU) and mean Average Precision (mAP) help in assessing the performance of object detection models?

   Evaluation metrics are essential for assessing object detection models, with Intersection over Union (IoU) and mean Average Precision (mAP) being key measures. IoU calculates the overlap between the predicted and ground truth bounding boxes, helping to evaluate localization accuracy, where a higher IoU means better overlap. mAP provides an overall measure of model performance by combining both precision (accuracy of positive detections) and recall (ability to find all objects). mAP averages the precision over different IoU thresholds and object classes, reflecting both classification accuracy and localization precision, making it a comprehensive metric for evaluating object detection models.

6. How does the threshold for non-maximum suppression affect the performance of an object detector?

   The threshold for non-maximum suppression (NMS) significantly impacts the performance of an object detector by influencing the balance between precision and recall. NMS eliminates redundant bounding boxes by retaining only the one with the highest confidence score when multiple boxes overlap for the same object, with the overlap determined by the Intersection over Union (IoU) threshold. A low NMS threshold (e.g., 0.3) may aggressively suppress boxes, potentially leading to missed detections when objects are close together, thus lowering recall. Conversely, a high NMS threshold (e.g., 0.7) allows for more overlap, reducing missed detections but may retain multiple boxes for the same object, thereby decreasing precision due to redundant detections. Therefore, tuning the NMS threshold is critical to ensuring accurate object localization while minimizing false positives.