Gavin Witsken

CS 590 - Computing in Healthcare

Assignment 3 Responses

1. Shallow neural networks are simple and typically contain only a **single hidden layer** between input and output, but deep neural networks can contain **many more hidden layers** that perform different transformations and operations on the data before reaching the output layer.
2. In neural networks, training occurs as a series of steps. First is the process of **forward propagation**, where data is fed into the input layer, through a series of hidden layers, where weights, biases, and activations are applied. Upon reaching the output layer, the prediction is made and forward propagation has completed.

   Then, the process of **back propagation** occurs where a loss function is used to determine how erroneous the prediction was. The gradient of the loss function is calculated with regard to the weights and biases of the network. Some kind of optimization algorithm like gradient descent, stochastic gradient descent, or Adam is used to update the pervious values of the weights and biases so the network can "learn" from its errors. This process of the forward propagation followed by back propagation will iterate until an acceptable end state is reached.
3. Activation functions are used to introduce some **non-linearity** to the output of the neurons to better represent curves of real data as well as **controlling** which data actually gets sent through the network.
4. Some common hyperparameters include learning rate, epoch, batch size, activation function, loss function, regularization functions/techniques.
5. Overfitting is when the model is too complex and fits too closely to the (usually training) data, meaning that it will not work well on new data and will have high variance (very nonlinear). Underfitting is the opposite, where the model is too biased toward a simple pattern cannot effectively represent nonlinear relationships within the data.
6. Overfitting is often mitigated through techniques like regularization and dropout, to help ensure that the model does not fit too closely to the training data.
7. The vanishing gradient problem is a scenario where the loss gradients get very small, almost to zero, during backpropagation. This happens often in networks with many layers, rapidly shrinking the shape of the data from input to output. This can be mitigated through using activation functions like ReLU to restrict which information passes through the network.

   The exploding gradient problem is the opposite scenario, where the loss gradients get very large, and the model becomes very unstable. Techniques like gradient clipping to clamp the values of gradients help mitigate this unstable growth behavior.

8. Batch normalization is a technique which helps improve the efficiency of the training process of neural networks. This operation is applied to individual layers of the network. Each mini-batch is normalized, then scaled and shifted according to learned scaling and shifting parameters. This allows for efficient learning with higher learning rates, lower internal covariate shift, mitigation of exploding and vanishing gradient problems, and reduced need for other regularization techniques.
9. A Convolutional Neural Networks consists of a series of different types of layers, each type performing a certain role.

   The convolutional layer is the namesake of this type of neural network, which performs the **convolution** operation, which slides some number of kernels (filters) over the input data, creating a feature map. It utilizes parameters like kernel size, stride, and padding.

   The **pooling** layer is another type of layer in a CNN which is used to effectively compress the input data into a smaller representation (output data). This makes computations more efficient and sometimes removes unwanted noise. These are often placed between consecutive convolution layers.

   The **fully-connected** layer is another type of layer which will typically occur near the end of the Network. These work on a flattened version of the previous data, and are used to effectively make the higher-level predictions that we want the network to perform.
10. A convolutional filter in a CNN is often used to detect certain kinds of features in our input. For example, we may want to identify an outline within an image to find certain shapes.
11. Convolutional neural networks can generally perform image processing tasks more efficiently than traditional image processing techniques because they require less pre-processing and are automatically able to learn hierarchical feature representations and can operate largely unsupervised by humans.
12. Different techniques can be used to effectively initialize values in a network. Often, some sort of randomized distribution is a better starting point than just starting everything at zero. However, the proper initialization technique depends heavily on the task at hand and the types of operations the network is performing.
13. Some popular CNN architectures include AlexNet, VGG, and ResNet. Each of these model architectures were designed for different tasks and involve different combinations of convolution, pooling, and fully-connected layers.
14. Batch size is the number of training examples which are used in calculating the loss function gradient before propagating the information back and updating the weights of the model. In theory, larger batch sizes will give a more accurate estimate of the loss function gradient, they will often realistically perform worse in generalization to new data compared to smaller batch sizes.
15. When choosing the optimal batch size, various things must be considered like capability of the hardware, learning stability, and model generalization. Hyperparameter

tuning techniques like grid search or Bayesian optimization are often used to determine this ideal batch size.

16. Some common use cases for CNNs aside from image classification are natural language processing, speech recognition, and time series analysis. We used it in class for DNA accessibility classification.
17. Some techniques often used in hyperparameter tuning are grid search, random search, Bayesian optimization, or simulated annealing. Similar to tuning of hyperparameters themselves, choosing the ideal optimization techniques depends on various factors like hardware capability, necessary operations, and scale of data.
18. Early stopping involves the use of a validation set during training, which is used to mitigate overfitting in the model based on some desired heuristic(s) for which a stopping point is usually defined.
19. Data transformation is the processing of taking some input data and converting into a format which is more easily usable for analysis. This can involve removing unnecessary features associated with the data, enumerating features that can be enumerated, aggregating data into summary measures, and much more. This transformation is often necessary for effective performance in a neural network model.
20. Some common transformation techniques include normalization, scaling, encoding, rotating, discretization, enumeration, aggregation, imputation, and much more.
21. Effective data transformation can help to optimize the data representation into a state which is more meaningful in regard to relationships between features, efficiency in training, and far higher quality for solving particular tasks. Poor transformation can easily do the opposite, adding unnecessary noise or removing potentially useful features.
22. Biases in data are errors in data, generally favoring some certain conclusion or outcome based on some given data which will often not scale effectively when applied to a broader sample range of that of data. Mitigating bias is incredibly important in data science, machine learning, and deep learning. When a model holds too much bias towards certain data, it may fail in adapting to new data which it is intended to make predictions about.
23. A DNA strand is a linear sequence of (ATGC) nucleotides which is very long, tightly packed into a structure called chromatin. Certain regions of this chromatin may be accessible, meaning genetic data encoded in that region can be read and utilized by machinery during various cellular processes such as transcription, replication, and repair. Some other regions of the chromatin will be inaccessible, meaning they cannot be read and utilized by this cellular machinery. This accessibility directly affects overall gene expression.
24. DNase-seq is an experimental DNA accessibility assessment technique with utilizes the DNase I enzyme to selectively cleave DNA where it is not protected by nucleosomes or other proteins (meaning it is an open, accessible chromatin region).

    ATAC-seq is another technique, but it uses a hyperactive Tn5 transposase to

essentially tag open regions of the chromatin which will highlight these accessible regions compared to the protected, inaccessible regions.

ATAC-seq is simpler, faster, and reveals more information than DNase-seq, but it is newer and has fewer comparative studies so there may be more unexplored bias in ATAC-seq compared to the historically-utilized DNase-seq technique.

25. A BED file is a tab-separated-value (TSV) file with a single sequence entry per line. Each entry will contain columns for the name of the chromosome, the start position of the chromosome, and the end position of the chromosome, as well as various other optional fields.
26. To create the negative (inaccessible region) BED file, we simply take the range between the end of the original accessible region and the start of the next accessible region for the same chromosome. We must validate that the values follow the proper bed format rules (no invalid ranges), and then we have generated a valid inaccessible region for our new BED file.
27. Bedtools is used to compare our regions against a reference genome and substitute the regions indicated in our bedfiles with the actual sequence of nucleotides which exist in that range within the genome. This is crucial so we can learn the pattern of the data with our model.
28. Nucleotides are the basic building blocks of DNA. DNA consists of four different types of nucleotides (Adenine, Thymine, Guanine, Cytosine). There are 23 chromosome base pairs in the human genome.
29. As described in the response to question 23, classifying DNA/chromatin region accessibility is vital in understanding gene expression. Understanding gene regulation and other things related to this gene expression helps us to identify various things like diseases which may be present, creating target environments for drug development, and even creating personalized medicine.
30. The coding regions of DNA contain the instructions for protein synthesis, which is used to describe how proteins and cells are to be synthesized, which is the basis for growth and development in organisms.

    The noncoding regions of DNA are regions that do not contain instructions for protein synthesis. One potential function these regions serve are as regulatory elements which may influence coding regions further upstream in the gene they are regulating. They are not necessarily located very close to the genes they regulate and can very helpfully increase the rate of transcription.
31. In genomics, biased data would be data that may not be a good representation of the general population (sampling bias), data where some regions are sequenced more extensively or accurately than others (coverage bias), or other reasons which cause certain regions of the genome to be particularly overrepresented or underrepresented compared to others.
32. Data must be converted to a tensor for use in pytorch because the pytorch library is designed to effective handle parallelized computations, typically leveraging the power

of the GPU's many simple, focused cores to perform computations. The tensor data structure allows for computation to be vectorized, and thus parallelized for great efficiency compared to standard sequential computation. These kinds of calculations are incredibly common in deep learning.

33. If we plotted our model's loss with respect to the epoch, we would ideally notice a decreasing trendline (negative slope). This would indicate that our model is performing better (less loss, better accuracy) as we run consecutive training iterations.
34. In deep learning, a validation set is useful for many kinds of tuning and optimizations that we want to perform with our model. First and foremost, the validation set is useful for hyperparameter tuning. By checking performance of different hyperparameter values on a test data set and comparing the results to a validation set, the hyperparameter values can be tweaked according to this performance discrepancy to achieve more desirable results from the model.
35. Generally, 32-bit floating-point precision will be used with training in deep learning (at least in our use cases for the course). During training or testing, we could lower the floating-point precision if we desired. This could significantly reduce the memory footprint of the data and our model, depending on how much raw data we want to process. Lowering the precision may also potentially allow for more calculations do be done more quickly (though this certainly depends on the hardware). Lower-precision calculations will be more energy efficient and less costly overall. Deciding on the precision to use will depend heavily on weighing these costs and benefits.