Module 3

Tools in AI Development

▼ What are some cluster-based AI solutions for managing large datasets?

- Hadoop
- Spark
- multi-agent systems

▼ Tensorflow

- developed by Google
- an open source ecosystem of tools, libraries, and community resources
- As of March 2020, Tensorflow is used by 15,000+ companies including DeepMind, Apple, and Uber
- Tensorflow Lite allows simple deployment on almost any device, including ARM devices
- has an advantage in hardware utilization: using GPUs, CPUs and TPUs (Tensor Processing Units, specialized ASIC chips to accelerate AI apps), and new forms of neuromorphic chips
- Since Tensorflow version 2.0, Keras can be used inside Tensorflow directly
- like PyTorch, both frameworks operate on tensors and view any model as a directed acyclic graph (DAG)
- you describe a graph statically before a model can run
- allows you to manually fine-tune every operation to run on a specific device, enabling unlimited manual parallelism

- has a clear advantage for deployment of inference
- important to consider what type of compute device you will use for training: your CPU or GPU?

▼ Keras

- an open-source ANN library supported by Tensorflow's core library since 2017

▼ PyTorch

- developed by Meta
- used by Twitter
- a major Tensorflow competitor
- more beginner-friendly than Tensorflow and known for being quick, simple and effective
- is Torch library's Python counterpart written in Lua
- more transparent modeling process than Tensorflow
- uses a "define-by-run" mode by default
- you can define, change and execute nodes without special session interfaces or placeholders
    - this makes debugging easier because you can use Python debugging tools like pdb or PyCharm debugger or Print() statements
- has declarative data parallelism (models can be easily parallelized over batch dimension)

▼ Apache's mxNet

a framework used by AWS, Microsoft and Intel for tasks like speech and handwriting recognition, NLP and forecasting

▼ Deeplearning4j

an open-source DL framework developed by Eclipse written primarily for Java and Java Virtual machine (JVM)

▼ Use of R for ML?

- for use in statistical analysis and machine learning
- it is graphically based and is preferred by statisticians and for work in bioengineering, bioinformatics, and biomedical statistics
- it is well suited to machine learning tasks such as regression, classification, and decision tree formation.

▼ Scala for ML

- Scala is a core language of Apache Spark
- is being utilized in deploying machine learning solutions in the world of big data

▼ Apache Spark

- a comprehensive data platform that provides functionalities for big data processing and machine learning analysis through its MLLIB library

▼ Lisp, Prolog, and Haskell for GOFAI (Good Old Fashioned AI)

- Lisp, Prolog, and Haskell are all logic-based languages that are well-suited for symbolic programming.
- They are common go-tos for development of GOFAI (Good Old Fashioned AI) systems that utilize symbol manipulation.

▼ An ANN in Python will typically have what components?

- an input layer
  - takes input data as a matrix and passes it on after multiplying the input by a set of weights
  - these input parameters are often initialized randomly, or sometimes restored from a previous network checkpoint
- an activation function
  - is nonlinear
  - determines if a neuron has been activated
  - examples: ReLU, Tanh, and sigmoid (easy for beginners)
- feedforward propagation

    - your code will iterate through the matrices given as input data
    - in each iteration, you will loop over the input data as a certain number of times, calculating the derivative (the rate of change of a function with respect to a variable) of the activation function
- backpropagation and loss function
    - a loss function that helps you readjust the input weights and biases by comparing the predicted values (output) from the previous iteration with the actual received value
    - the goal is to minimize the loss function over each iteraction (improve prediction performance)
    - backpropagation is used to adjust the neuron weights by working backwards
    - backpropagation uses an Error Weighted Derivative formula determined by the activation function

▼ How many typical iterations for ANN training process?

is iterated upon no less than 10,000 times and sometimes more depending on the use-case

▼ After training, you can use the model for

inference

▼ Why does the use of Tensorflow often result in higher-dimensional (more layers and nodes) ANNs?

Tensorflow allwos you to process much larger matrices