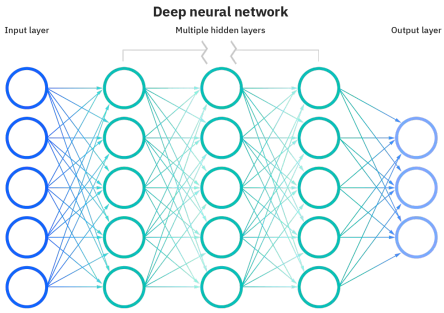
Overview of TensorFlow

TensorFlow is one the the most popular, open-source deep learning frameworks that exists today. Deep learning described by IBM is "a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain allowing it to "learn" from large amounts of data"[1]. As shown in the picture, essentially the deep learning model takes in inputs and maps them to outputs through layers of neural network (typically more than one layer) which are just algorithms.



Neural networks learn to do things by learning from examples, like our brains. An example would be training a model to recognize whether a picture is that of a cat or a dog. Other applications include facial recognition, language translation, and fraud detection [2]. TensorFlow is an open source framework developed by Google researchers to facilitate machine learning [3]. There are two versions of TensorFlow, 1.0x and 2.0x which has Keras as the high-level API, making it easier to build models, and easier for users to get started with TensorFlow. It also has a more compatible TensorFlow ecosystem meaning as soon as the model is completed it can be deployed to other TensorFlow platforms . Keras provides various model building APIs such as sequential, functional and subclassing. Sequential API, is just a list of layers, which has one input and output stacks of layers. Functional API is more flexible, and allows for the model to break out into two different models and is more complex than sequential API. Subclassing lets you implement everything from scratch, suitable for research. TensorFlow's core component is known as a Tensor. Tensor is a vector or a matrix of n-dimensions which represents all types of data. They are ranked based on the number of dimensions, Scalar would be rank-0, Rank-1 tensor would be a list that is 1 dimension. Rank-2 would be a matrix with two dimensions and rank N tensor would have n-dimensions [4]. TensorFlow performs computations with the help of dataflow graphs. It has nodes or neurons that represent the computations in your model. The wights from the computations in the model lead to the final decision.

In order to better understand TensorFlow, we can look at an application of TensorFlow such as natural language processing (NLP). By going through the steps of NLP, it will become apparent how efficiently a user can build a model in TensorFlow. To start with NLP we first need

to understand how a computer processes words. This is called Tokenization, it is the process of splitting sentences into tokens, like into words, so the model understands them. This is done using a tokenizer function built into the Keras API from TensorFlow. Which returns back a dictionary of the tokens, which are most common words and their indexes 7. Next it is important to remove words which do not provide any value to the mode such as “and”, “the”, and ”to”. This is done to ensure the accuracy of the model is not skewed by irrelevant words. Next, it is important to understand sequencing, we convert these sentences into sequences using the generated tokens. This is done by using the texts_to_sequences method in Keras. One problem we may encounter when we tokenize unseen words, words that were not part of the vocabulary when tokenization is done. In order to tackle the problem of tokenizing, we add “<OOV>” to our tokenizer and use a method called padding. The <OOV> stands for out of vocabulary, which tokenizes words from sentences that were not part of the initial vocabulary. In order to handle the sequences not being the same length, we use the pad_sequences method from module TensorFlow.Keras.preprocessing. This will add padding to the beginning of the sentence to make room for the tokens we have not seen before. Another method that is used to make text more understandable is stemming, by reducing words to their stem. Such as taking “waits”, “waiting” and “waited” and reducing it to wait. Lemmatization is a process of switching any kind of word to its base root. It is done to group different words of the same root, to have the same meaning. Another important technique to better train the model is topic modeling, that groups different texts under different subjects. After importing, we can start building the NLP model. The dataset is separated into training, validation and testing sets. The training data will be calibrated into the tokenizer, followed by the sequencing method. We use the padding methods, to make all the sequences the same shape [5]. In order to build a simple model we can add two hidden layers to the model. This is done by calling twice the model.add() method and passing the tf.kears.layers.Dense() which takes in the dimensionality of the output and the activation function. We compile the model and its loss, optimizer, and measure its accuracy. We fit the model by passing the training data, then we can evaluate the model by simply model.evaluate() method by passing testing data.

NLP has experienced rapid growth and advancement, and TensorFlow gives users the tools needed to create and deploy models efficiently. It allows users to generate models from complex unstructured data without having to worry about the details like which algorithms to implement. This abstraction is valuable because it allows the developer to focus on the overall logic of the application. A key characteristic of TensorFlow is the CPU and GPU support, since deep learning applications can be very complicated and the process requires a lot of computations. TensorFlow supports GPUs which significantly speed processing time. Another valuable asset of TensorFlow is the documentation by google, aside from the documentation, because of the sheer popularity of TensorFlow there is a huge community that almost always has answers that you're looking for. Aside from NLP, TensorFlow has tons of applications, making it easy to use it for whatever use-case your project needs. The ease of use, robust APIs, community

support and broad range of use cases, make TensorFlow one of the best Machine learning and artificial intelligence tools.

Citations

1. By: IBM Cloud Education (no date) *What is deep learning?*, *IBM*. Available at: https://www.ibm.com/cloud/learn/deep-learning (Accessed: November 6, 2022).
2. By: IBM Cloud Education (no date) *What are neural networks?*, *IBM*. Available at: https://www.ibm.com/cloud/learn/neural-networks (Accessed: November 6, 2022).
3. Vaughan, J. (2018) *What is tensorflow? - definition from whatis.com*, *SearchDataManagement*. TechTarget. Available at: https://www.techtarget.com/searchdatamanagement/definition/TensorFlow (Accessed: November 6, 2022).
4. About Sakshi Gupta Sakshi is a Senior Associate Editor at Springboard. She is a technology enthusiast who loves to read and write about emerging tech. She is a content marketer and has experience working in the Indian and US markets. (2022) *Tensorflow 1.0 vs. tensorflow 2.0: What's the difference?*, *Springboard Blog*. Available at: https://www.springboard.com/blog/data-science/tensorflow-1-0-vs-tensorflow-2-0/ (Accessed: November 6, 2022).
5. *Introduction to the tensorflow Models NLP library : text* (no date) *TensorFlow*. Available at: https://www.tensorflow.org/tfmodels/nlp (Accessed: November 6, 2022).