Annotation and Benchmarking on Understanding and Transparency of Machine learning Lifecycles (ABOUT ML)

Supplement: Machine Learning Primer

The goal of this document is to provide enough preliminary background and definitions to enable the reader to follow along with the main body of the ABOUT ML recommendations. In the future, PAI may add further resources for people who want to learn more about the topic of machine learning.

What is AI?

Artificial intelligence (AI) is any computational process or product that appears to demonstrate intelligence through non-biological/natural processes. With recent advances in different branches of AI technology, AI can now do important tasks, including identifying and deciphering the objects in images ("computer vision (CV)"), interpreting text in various ways (“natural language processing (NLP)”), and controlling robots or game agents via strong feedback loops (“reinforcement learning”). AI can read handwriting and generate stories, can identify faces as well as successfully play games like chess or Go.

What is machine learning?

One mainstream approach to building AI is machine learning (ML) - a statistical and mathematical modeling approach to approximate the patterns between input and output variables using data. Machine learning is a method to uncover statistical correlations within a dataset and could range from simple linear regression to more complicated algorithms. On the more complex side, examples include models that use a multi-layered network structure (not unlike neurons in a brain) to map inputs to outputs, called "deep learning" (DL), and its many variations, such as convolutional neural nets (CNNs), Long/Short Term Memory Networks (LSTMs), etc. Many of these variations have become buzzwords in recent years as advances in ML technology have lead to rapid breakthroughs.

A machine learning model, like any other mathematical function, is attempting to map inputs (“features”) to an expected output (“label” or "prediction"). This could mean matching a specific color, texture, and shape in an image to the label “dog”, or a specific combination of notes to the assignment of a song genre. Ideally we want to be able to send inputs the model has never encountered before and get an output that closely matches the expected relationship from past experience and data. In machine learning, we achieve this by describing that expected relationship with constant terms derived from these past experiences, and using

those constants to define a model function. The hope is that if future examples closely match past ones, then the model will do a good job giving the proper output even for an input it has not been previously exposed to, as it has already defined a relationship between inputs and outputs based on examples from past experiences.

These models are defined by a combination of data and an algorithm (see Figure 0.1 below). The algorithm is effectively a recipe - the set of steps required to define the constants in the final model. It dictates the mechanism through which the relationship between the inputs and the outputs is determined, and reflects the functional form of the final model.

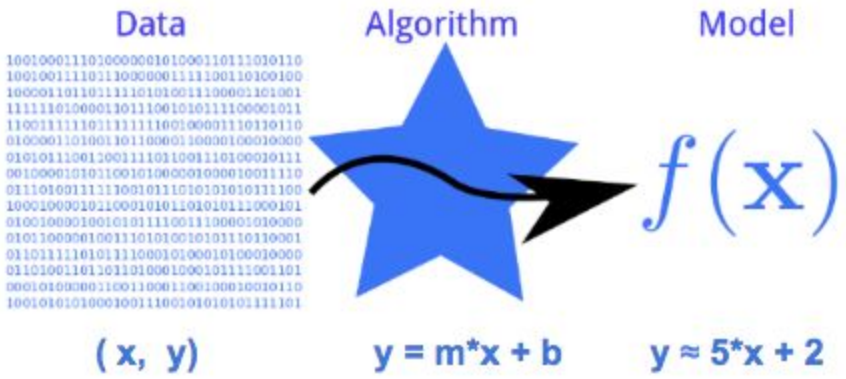

Figure 0.1 : Overview of a general machine learning model structure

In order to define the specific relationship between inputs and outputs that we hope to get from the machine learning model, we go through a process of training the algorithm on data (see Figure 0.2). This process involves providing many examples of the relationship between inputs and outputs, which then, through the set of steps defined by the algorithm, define the unchanging, consistent patterns (ie. “constants” or “weights”) of that relationship - effectively setting a concrete worldview of the relationship between features and labels, based on the dynamic found in the training data.

Training a machine learning model can take anything from a couple of seconds on a single computer to weeks and months on thousands or more machines. The training process can involve calibrating the constants (or “weights”) of the final model based off of anything from thousands to billions (or more) of training data examples. If the training is set up like a matching game - showing the algorithm several examples of features and their correct labels during training to estimate the weights, then we call this process “supervised learning.” If the training data does not involve information on the desired predicted output (“labels”) of the model, and uses other techniques to define the function based on the implicit relationship between the inputs and outputs of past experiences, then this model is known as “unsupervised.” Though these are general techniques, other methods exist as well - for instance “reinforcement learning” when the model defines the weights from adjusting its model based on a feedback loop based on interactions with its environment.

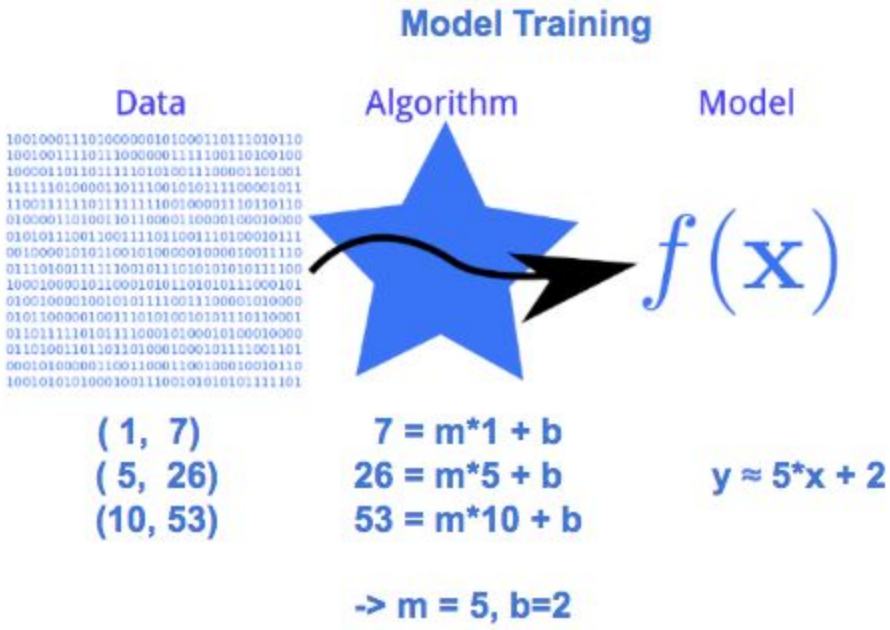

Figure 0.2 : Overview of training process for a machine learning model

Once trained, the machine learning model is a defined function that encodes the estimated relationship between inputs and outputs. This trained machine learning model will give an approximate output prediction from any new input, based on what it has estimated as the relationship of the input and output pairs it was trained on implicitly or explicitly in the past. In order to update the model, and upgrade its worldview to encompass a new context or domain, the model must be retained to infer a new relationship between input and output, likely using the same mechanism (algorithm) but with a new set of training data, more representative of that new context.

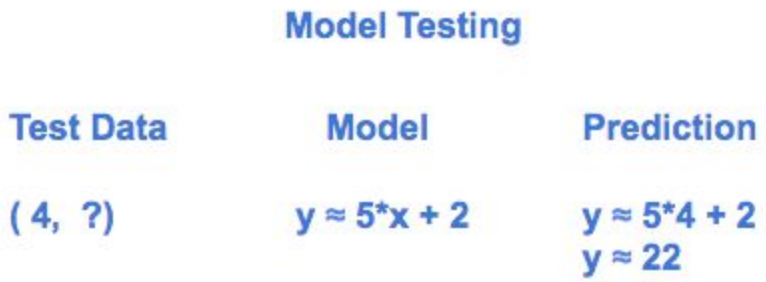

Figure 0.3 : Overview of testing process for a machine learning model

What are different contexts of use for data?

[This section is meant to define what it is, how it's used in machine learning, typical sources (collection methods etc)]

Training data is data analyzed through an algorithm in order to generate a machine learning model (the data from which the model 'learns'). The resulting model is thus an encoding of the relationships found in the data and reflects the biases of the training dataset.

Testing data is the dataset used in the evaluation of the model. The biases in this dataset can prevent a realistic understanding of the model's performance in its deployment context.

Validation data is similar to the testing data, except it is used in the process of system development, and revisited as engineers work to improve the model, with changes to either algorithm or training data. This means that model development can continue after evaluating model performance on the validation set, making it integral to supporting engineers in determining when the model is ready for release and informing engineering design decisions for improvement.

What is a model?

A model is a defined input-output function that takes in a set of inputs (ie. "features") and provides a prediction for the expected output (i.e. "label") for that input, based on the learned past relationship of previous input-output pairs it has been exposed to directly or indirectly.

What is an ML system?

A machine learning (ML) system can take on multiple forms, from a simple defined function embedded in a computer program to be run, to a deployed Application Program Interface (API) to an ecosystem of models packaged into a software package and presented via a Graphical User Interface (GUI). This system involves some form of automated decision making to accomplish a wide range of tasks from classification and prediction to database search and recommendation. The models in the system are defined through a process known as machine learning (ML) , and an ML system incorporates one or more of models developed through this method in order to obtain a final result.

The reference to ML stakeholders refers to the entire population of those interacting or engaging with the system in any way, either as end users of the system, the public affected by the systems decisions, those involved in the procurement and distribution of the system and those engaged in the development and specification around the system.

Those responsible for the creation of the system - usually engineers or researchers either at a company, academic institution or other organization - often develop ML models to address a relatively narrow use case (i.e. facial recognition, natural language processing, etc.), but can

also be trained for broader impact if the data is available. The scope of a model's effective operation is often limited by the training data available to develop a model to operate in that context, as a machine learning model will often make unreliable predictions for new input data that does not conform to the data that it was trained on.

An ML system can involve a combination of machine learning models, chained to feed into each other (ie. predictions of one model are the input of the next), or interact with a greater system in some way. The overall decision and prediction made by the system is learned from previous training data (ie. examples from past experience), but in a system, that prediction may also be influenced by other factors before being incorporated into the final result.