Practical Data Science Coursera Specialization

by DeepLearning.AI

Course #2 Build, Train, and Deploy ML Pipelines using BERT

Course Site

Made By: Matias Borghi

Table of Contents

Summary

In the second course of the Practical Data Science Specialization, you will learn to automate a natural language processing task by building an end-to-end machine learning pipeline using Hugging Face's highly-optimized implementation of the state-of-the-art BERT algorithm with Amazon SageMaker Pipelines. Your pipeline will first transform the dataset into BERT-readable features and store the features in the Amazon SageMaker Feature Store. It will then fine-tune a text classification model to the dataset using a Hugging Face pre-trained model, which has learned to understand the human language from millions of Wikipedia documents. Finally, your pipeline will evaluate the model's accuracy and only deploy the model if the accuracy exceeds a given threshold.

Practical data science is geared towards handling massive datasets that do not fit in your local hardware and could originate from multiple sources. One of the biggest benefits of developing and running data science projects in the cloud is the agility and elasticity that the cloud offers to scale up and out at a minimum cost.

The Practical Data Science Specialization helps you develop the practical skills to effectively deploy your data science projects and overcome challenges at each step of the ML workflow using Amazon SageMaker. This Specialization is designed for data-focused developers, scientists, and analysts familiar with the Python and SQL programming languages and want to learn how to build, train, and deploy scalable, end-to-end ML pipelines - both automated and human-in-the-loop - in the AWS cloud.

Week 1: Overview of the ML Lifecycle and Deployment

This week covers a quick introduction to machine learning production systems focusing on their requirements and challenges. Next, the week focuses on deploying production systems and what is needed to do so robustly while facing constantly changing data.

Learning Objectives

- **Identify the key components of the ML Lifecycle.**
- **Define "concept drift" as it relates to ML projects.**
- **Differentiate between shadow, canary, and blue-green deployment scenarios in the context of varying degrees of automation.**
- **Compare and contrast the ML modeling iterative cycle with the cycle for deployment of ML products.**
- **List the typical metrics you might track to monitor concept drift.**

Week 2: Select and Train a Model

This week is about model strategies and key challenges in model development. It covers error analysis and strategies to work with different data types. It also addresses how to cope with class imbalance and highly skewed data sets.

Learning Objectives

- **Identify the key challenges in model development.**
- **Describe how performance on a small set of disproportionately important examples may be more crucial than performance on the majority of examples.**
- **Explain how rare classes in your training data can affect performance.**
- **Define three ways of establishing a baseline for your performance.**
- **Define structured vs. unstructured data.**
- **Identify when to consider deployment constraints when choosing a model.**
- **List the steps involved in getting started with ML modeling.**
- **Describe the iterative process for error analysis.**
- **Identify the key factors in deciding what to prioritize when working to improve model accuracy.**
- **Describe methods you might use for data augmentation given audio data vs. image data.**
- **Explain the problems you can have training on a highly skewed dataset.**
- **Identify a use case in which adding more data to your training dataset could actually hurt performance.**
- **Describe the key components of experiment tracking.**

Week 3: Data Definition and Baseline

This week is all about working with different data types and ensuring label consistency for classification problems. This leads to establishing a performance baseline for your model and discussing strategies to improve it given your time and resources constraints.

Learning Objectives

- **List the questions you need to answer in the process of data definition.**
- **Compare and contrast the types of data problems you need to solve for structured vs. unstructured and big vs. small data.**
- **Explain why label consistency is important and how you can improve it**
- **Explain why beating human level performance is not always indicative of success of an ML model.**
- **Make a case for improving human level performance rather than beating it.**
- **Identify how much training data you should gather given time and resource constraints.**
- **Describe the key steps in a data pipeline.**
- **Compare and contrast the proof of concept vs. production phases on an ML project.**
- **Explain the importance of keeping track of data provenance and lineage.**

Week 1: Overview of the ML Lifecycle and Deployment