Contents

1. Introduction:

Data storage is widely used and in high demand in today's interconnected business ecosystem. For example, each day Walmart stores potentially several petabytes of data due to the millions of transaction records, updates of inventory stock, information of new customers, etc. Merchants are always required to choose the best storage solution to avoid unnecessary overhead costs and provide more comfort services for customers. Their solution is storage management systems using cloud, tape, and disk.

Currently, cloud storage seems to be the best option for many companies. It is affordable compared to traditional disk storage since it doesn't need to be operated by the business itself and pays for itself in terms of ownership, maintenance, and operation of servers. Merchants can rent these servers provided by companies like Amazon and Microsoft. Most importantly, customers can quickly access data stored on the cloud at any time, anywhere, with many different types of devices.

Disk and tape storage are the other two forms of storage systems that IBM Spectrum protect uses. Disk is one of the better storage solutions for quickly accessing data. It is quite expensive compared to cloud or tape but if a customer wants to access their data immediately then disk would be the right way to go. Tape on the other hand is a cheaper form of storage however it takes longer to access by the customer therefore it is mainly for data that doesn’t need to be accessed immediately.

Our project is sponsored by Daniel Boros who is a staff software engineer in the IBM Spectrum Protect project. IBM Spectrum Protect is a generalized monolithic server with cloud capabilities, which is designed to simplify protection for large amounts of data hosted in physical, virtual, software-defined, and cloud environments for all customers. Also, Spectrum Protect simplifies backup administration, improves efficiencies in the backup process, and enable scalability to an entire enterprise of inputs. Our project is committed to improving secure enterprise level storage for IBM by providing a good solution for classifying data into the appropriate storage tier.

As of now the data that’s being stored by Spectrum Protect is often miscategorized. An example would be data that should be stored in tape is now stored in disk. This means that after a period of time it will get demoted to a

lower tier by policies in place to help categorize this data. These policies take some time to take action and during the time that this data is getting demoted, overhead costs for IBM increase the longer the data is in an incorrect tier. If the opposite happens and data that should be on disk is now on tape, customers might not be able to access their data quickly enough and depending on the situation could waste time and money for them as well. The costs for Spectrum Protect increase as more data is stored incorrectly which is why handling the storage tier on ingestion is crucial. Our project is important because it can save time, money, and effort that can be handled by this pipeline.

In this document, we will go into detail as to how we will be designing and implementing our project. Our pipeline starts with Apache Spark, our big data processing tool that will allow us to pre-process data before sending it over to Tensorflow where we can create a model using supervised learning to then be able to classify incoming data correctly. This pipeline must be scalable, secure, reliable, and be able to create a model with at least 80% accuracy. This model will be used to classify data into the correct tier.

This project will be utilizing free or open source technology only and use of machine learning to devise a solution is necessary. Throughout our entire process, we will also need to document every step of the way extensively.

2. Implementation Overview:

Now that we know the issue, understand why it's a problem, and have a generalized plan of attack, let's go into slightly more detail as to how our system will be implemented.

Our pipeline will consist of two main technologies and one main machine learning method. The first is Apache Spark. Apache Spark is a big data processing tool capable of handling petabytes of data and pre-processing this data so that Tensorflow can create more accurate models. Tensorflow is the second big technology that will take in this pre-processed data and create and continuously train a model so to better the accuracy of classifying data into correct storage tiers.

Before we can pre-process data in spark, we must combine the test CSVs so that they are readable to Apache Spark as parquet files. To combine these CSVs, we will be using dask and pyarrow (or fastparquet) in a jupyter notebook. Once these CSVs are combined into parquet files, we are then able to pass those files over to Spark.

We will be using Apache Spark via PySpark, a Python library that allows easy interfacing to all of Apache Spark's features. In the initial stages of development we will be using a Jupyter Notebook that can access Spark, but in the middle to later stages, we will have everything in a docker. One reason why Python was a very popular language in that Apache Spark and Tensorflow both had Python libraries so our project would stay consistent.

Our machine learning framework, Tensorflow, is a powerful user-friendly tool easily interfaceable with Python. It is compatible with multiple operating systems and can be easily installed using Python's Pip. Tensorflow can also be used in a Jupyter Notebook which is convenient for early development stages.

Our machine learning method of choice for this project is supervised learning. This is because we will have been given a training set and test set of data which we believe will provide more accurate results than other machine learning algorithms such as unsupervised or semi-supervised learning.

3. Architectural Overview:

The architecture of our application will consist of three critical parts, Data preprocessing, Model training, and Model Demonstration.

4. Module and Interface Descriptions:

Module ….

5. Implementation Plan:

Our plan ...

6. Conclusion:

To conclude, data uploaded to IBM's Spectrum Protect product is often miscategorized into the wrong storage tier. This requires manual intervention by the system administrators to correct miscategorizations after the files have been in the wrong tier for too long. The miscategorization of data can lead to an increase in costs to not only our sponsor and clients but can also be a considerable inconvenience. With our pipeline of technologies hitting the desired accuracy mark is achievable.

In this document, we went into detailed specifics on how we will be creating our product. This will not only help us have a better understanding of where to go from here but also to identify risks and possible changes we might need to make in our architecture before it's too late. However, our technologies and methods should prove to work as intended, and we don't expect any significant changes in our requirements or architecture. Through this document, we are also able to make sure that we are on the right path to success in terms of how we will go about our tasks inside each technology. Having a secure, written, and easily understandable design document can help the outcome of the project overall.

So far we have verified that the technologies we have chosen can work together and create the desired pipeline. We are almost completely done with the preprocessing portion and we have successfully created a model using sample data sent to us by our sponsor. We have also verified that our code so far is scalable to the extent that is required by our sponsor. At this point we will be able to optimize our code to produce a model with better accuracy and add other desirable features such as a GUI that displays necessary graphs and data. Moving forward we believe this project will be able to attain that 80% accuracy and provide a well constructed product for our sponsor.