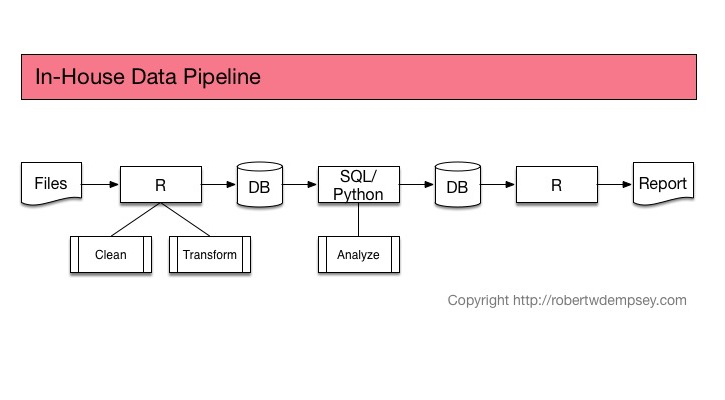
Data is the oil of the 21st Century. Data analytics drives decision making, which creates value for companies and its stakeholders. However, data analytics is often times restricted to an active person using a personal device and analyzing stationary data. In today’s age, there are platforms that allow organizations to harness and utilize data in a sophisticated fashion. Hence in this new age of data analytics, there are new technologies and platforms burgeoning that allow us to conduct real-time data analysis and big data analysis. This is where an important concept of data pipelining comes into picture.

Data pipeline is a framework to illustrate workflows of data mediation. In other words, data is generated which then passes through processes like analysis, E.T.L Etc using techniques such as machine learning and others. This data is processed based on the users’ requirements for their useful application.

An illustration of a data pipeline is as below:



Referencing this to a real world scenario, we can discuss an implementation of data pipeline being implemented by security systems giant – Tyco.

Tyco is supplier of security systems such as CCTVs, Circuit boards, Electronic Article Surveillance, Fire Alarm Systems and so on. We are going to address the problem Tyco is facing in regards to CCTV systems.

**THE PROBLEM:**

* A very fundamental problem for any security systems based company is when their systems predict a false positive.

An example

Tyco builds motion detector cameras. A motion detector camera is a security system that tracks movement of an object in its surveillance radar. Once the object is identified on the detector lens, the camera follows the objects position due to its motion sensing ability. This is utilized to alert the company and further alert the police for immediate action. Here lies the big elephant we will discuss about.

As long as it is a crime detection by the detector, then the alert sent to the police is good. But what if it’s any other object which has nothing to associate with crime whatsoever. Eg. Racoon, Hare etc. However, what if the motion detector detected is as a threat. Hence, the problem faced by Camera Motion Detector is of False positives. False positives are instances when a suspect of threat is identified when it is not.

There are interesting scenarios that may further add to the misery. Imagine a scenario where a person walks into an apartment with a Racoon costume. Even though this is a prank situation, a false positive may occur. Therefore the kind of activity of the object is important, than the mere object itself in most cases.

* Thus, a data pipeline that can help solve such a problem is very essential for companies facing false positive results.

**IMPLICATIONS OF THE PROBLEM:**

As a business person, tradeoffs are the most important aspect of decision making. The product’s developer might argue that the likelihood of such a prank instance is very less. Therefore, I would not invest in further improving the model’s accuracy. Hence, several questions intrigue a business person such as what are the costs and how much are the costs associated with building an better detection additional plug, what are the costs incurred for false positives, will I lose my customers, how much of my profit is going to get affected, how will the dynamics related to the police department come into play, and so on and so forth.

Also depending on the company’s total customer base, the development and operations costs will be different. The code build for better detection has to be modified for different servers. For some companies, it might be very computationally expensive. In the future, when consultants make a recommendation for additional plugins, they will need a strong justification for the associated costs.

**GOAL:**

Our objective is to build a data pipeline that will help solve the issue related to detection of false positive suspects.

**MAIN ACTIVITIES:**

Our goal is pertinent to resolving issues such as appropriately detecting objects in an image such that highest possible accuracy can be achieved. This will be extremely beneficial for security based companies that rely on machine learning and AI to precisely identify what information lies in a particular graphic file such as an image or video.

Hence, our main activities are:

Gather image data for processing

Process images to detect potential threats

Organize and aggregate the data for consumption

**TOOLS and TECH:**

We can imagine a person writing codes to gather the data from a database, then organizing it on his personal computer, then loading to another device to convert the format of data. Then redownloading it, making changes and then pushing it to his client.

For our goal, we can imagine using technology to build an automated process which includes extraction of files (images or videos captured by the camera), management of these files across the interlink, analysis of these files and so on. For achieving this, we look into cloud platform utilization for automation deployment. Following are the software and tools we will utilize to help achieve our goals:

1. Kubernetes
2. Docker
3. Pachyderm
4. Python
5. Python - library TensorFlow

**IMPLEMENTATION:**

Docker allows to create an image of process flows (processing stages) in the form of a docker containers. For instance, we can create an image such as the following :

Kubernetes deploys these containers to various servers. If we were working without Kubernetes, we would have to manually distribute and deploy our docker containers to various servers on the cloud. Because of Kubernetes, this process gets automated. This also enables the space utilization for our Docker containers to be optimized. Below is an example of Kubernetes workflow:

TensorFlow library provides us with an extremely powerful analysis and processing package for object detection tasks. This is particularly important for goal.

Pachyderm is extremely useful for our goal in terms of data versioning, defining our data pipelines and managing access to data.

We will utilize Python for data manipulation and processing.

For a simpler understanding, we can say that Kubernetes handles the server allotment for Docker’s containers (and the images), and Docker is utilized to create Docker containers (and the images) of processing stages. Pachyderm will allow us to create various versions of our dataset, defining our workflow and management and dataset related administration. TensorFlow will allow us to analyze our datasets.

Considering the heavy usage of various cloud, we might work with at least one of the following data storages:

1) Block Storage:

This storage is for organized data files usually. Organized data files has reference to files containing data in the form of tables or a structured format. Almost all the companies have such data pertaining to their customers, internal operations, financial data and so on. Some of the storage applications include Virtual Machines (VMs), Databases, Email Servers etc.

2) File Storage:

This storage can be observed something similar to a drive on our computer, eg. C: drive, D: drive. Speaking in relevance to big data and cloud platform, a File storage platform can be Apple Cloud, Google cloud. Such type of storage is very familiar in case of email storage, file sharing etc.

3) Object Storage:

This storage is extremely handy when it comes to dumping large amounts of data that is unstructured and unclean, such data is utilized in the future but temporarily requires storage that is cost-effective and flexible. Data is stored in a flat memory model and saved with a unique ID. Hence it becomes very easy to retrieve the data based on the unique id. Some of the storage applications include Big Data, Web Apps, Backup Archives etc.