The Data Engineering Cookbook

How to master the plumbing of data science

Andreas Kretz

January 6, 2019

v0.4

Contents

1 Introduction

What do you actually need to learn to become an awesome data engineer? Look no further, you find it here.

How to use this document: This is not a training! It's a collection of skills, that I value highly in my daily work as a data engineer. It's intended to be a starting point for you to find the topics to look into.

This project is a work in progress! Over the next weeks I am going to share with you my thoughts on why each topic is important. I also try to include links to useful resources.

How to find out what is new? You will always find the newest version on my Patreon **https://www.patreon.com/plumbersofds**

Help make this collection awesome! Join the discussion on Patreon or write me an email to andreaskayy@gmail.com. Tell me your thoughts, what you value, you think should be included, or where I am wrong.

– Andreas

2 What is Data Science?

2.1 Data Scientist

Data scientists aren't like every other scientist.

Data scientists do not wear white coats or work in high tech labs full of science fiction movie equipment. They work in offices just like you and me.

What differs them from most of us is that they are the math experts. They use linear algebra and multivariable calculus to create new insight from existing data.

How exactly does this insight look?

Here's an example:

An industrial company produces a lot of products that need to be tested before shipping.

Usually such tests take a lot of time because there are hundreds of things to be tested. All to make sure that your product is not broken.

Wouldn't it be great to know early if a test fails ten steps down the line? If you knew that you could skip the other tests and just trash the product or repair it.

That's exactly where a data scientist can help you, big-time. This field is called predictive analytics and the technique of choice is machine learning.

Machine what? Learning?

Yes, machine learning, it works like this:

You feed an algorithm with measurement data. It generates a model and optimises it based on the data you fed it with. That model basically represents a pattern of how your data is looking You show that model new data and the model will tell you if the data still represents the data you have trained it with. This technique can also be used for predicting machine failure in advance with machine learning. Of course the whole process is not that simple.

The actual process of training and applying a model is not that hard. A lot of work for the data scientist is to figure out how to pre-process the data that gets fed to the algorithms.

Because to train a algorithm you need useful data. If you use any data for the training the produced model will be very unreliable.

A unreliable model for predicting machine failure would tell you that your machine is damaged even if it is not. Or even worse: It would tell you the machine is ok even when there is an malfunction.

Model outputs are very abstract. You also need to post-process the model outputs to receive health values from 0 to 100.

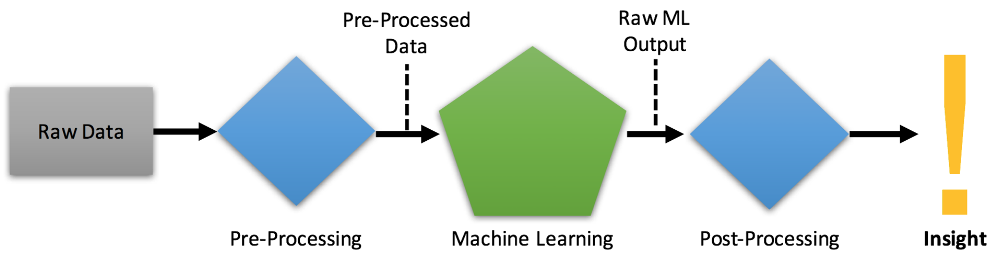

Figure 1: The Machine Learning Pipeline

2.2 Data Engineer

Data Engineers are the link between the management's big data strategy and the data scientists that need to work with data.

What they do is building the platforms that enable data scientists to do their magic.

These platforms are usually used in four different ways:

– Data ingestion and storage of large amounts of data – Algorithm creation by data scientists – Automation of the data scientist's machine learning models and algorithms for production use –Data visualisation for employees and customers – Most of the time these guys start as traditional solution architects for systems that involve SQL databases, web servers, SAP installations and other "standard" systems.

But to create big data platforms the engineer needs to be an expert in specifying, setting up and maintaining big data technologies like: Hadoop, Spark, HBase, Cassandra, MongoDB, Kafka, Redis and more.

What they also need is experience on how to deploy systems on cloud infrastructure like at Amazon or Google or on premise hardware.

2.3 Data Analyst

2.4 Who Companies Need

For a good company it is absolutely important to get well trained data engineers and data scientists.

Think of the data scientist as the professional race car driver. A fit athlete with talent and driving skills like you have never seen.

What he needs to win races is someone who will provide him the perfect race car to drive. That's what the solution architect is for.

Like the driver and his team the data scientist and the data engineer need to work closely together. They need to know the different big data tools Inside and out.

Thats why companies are looking for people with Spark experience. It is a common ground between both that drives innovation.

Spark gives data scientists the tools to do analytics and helps engineers to bring the data scientist's algorithms into production.

After all, those two decide how good the data platform is, how good the analytics insight is and how fast the whole system gets into a production ready state.

3 The Basic Skills

4 Learn to Write Code

Why this is important: Without coding you cannot do much in data engineering. I cannot count the number of times I needed a quick Java hack.

The possibilities are endless:

- Writing or quickly getting some data out of a SQL DB
- Testing to produce messages to a Kafka topic
- Understanding Source code of a Java Webservice
- Reading counter statistics out of a HBase key value store

6 My Big Data Platform Blueprint – available

Some time ago I have created a simple and modular big data platform blueprint for myself. It is based on what I have seen in the field and read in tech blogs all over the internet.

Today I am going to share it with you.

Why do I believe it will be super useful to you?

Because, unlike other blueprints it is not focused on technology. It is based on four common big data platform design patterns.

Following my blueprint will allow you to create the big data platform that fits exactly your needs. Building the perfect platform will allow data scientists to discover new insights.

It will enable you to perfectly handle big data and allow you to make data driven decisions.

THE BLUEPRINT – available The blueprint is focused on the four key areas: Ingest, store, analyse and display.

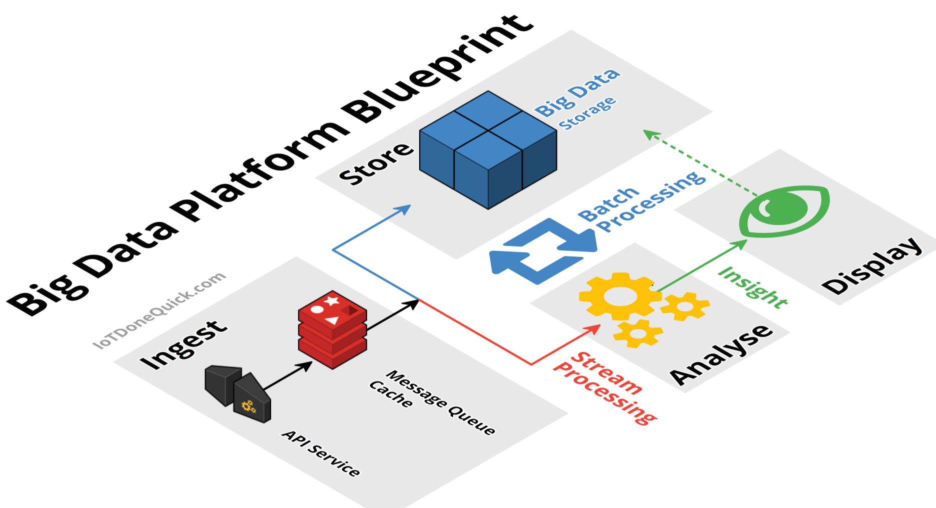

Figure 2: Platfrom Blueprint

Having the platform split like this turns it it a modular platform with loosely coupled interfaces.

Why is it so important to have a modular platform?

If you have a platform that is not modular you end up with something that is fixed or hard to modify. This means you can not adjust the platform to changing requirements of the company.

Because of modularity it is possible to switch out every component, if you need it.

Now, lets talk more about each key area.

6.1 Ingest – available

Ingestion is all about getting the data in from the source and making it available to later stages. Sources can be everything form tweets, server logs to IoT sensor data like from cars.

Sources send data to your API Services. The API is going to push the data into a temporary storage.

The temporary storage allows other stages simple and fast access to incoming data.

A great solution is to use messaging queue systems like Apache Kafka, RabbitMQ or AWS Kinesis. Sometimes people also use caches for specialised applications like Redis.

A good practice is that the temporary storage follows the publish, subscribe pattern. This way APIs can publish messages and Analytics can quickly consume them.

6.2 Store – available

This is the typical big data storage where you just store everything. It enables you to analyse the big picture.

Most of the data might seem useless for now, but it is of upmost importance to keep it. Throwing data away is a big no no.

Why not throw something away when it is useless?

Although it seems useless for now, data scientists can work with the data. They might find new ways to analyse the data and generate valuable insight from it.

What kind of systems can be used to store big data?

Systems like Hadoop HDFS, Hbase, Amazon S3 or DynamoDB are a perfect fit to store big data.

6.3 Analyse / Process – available

The analyse stage is where the actual analytics is done. Analytics, in the form of stream and batch processing.

Streaming data is taken from ingest and fed into analytics. Streaming analyses the "live" data thus, so generates fast results.

As the central and most important stage, analytics also has access to the big data storage. Because of that connection, analytics can take a big chunk of data and analyse it.

This type of analysis is called batch processing. It will deliver you answers for the big questions.

To learn more about stream and batch processing read my blog post: How to Create New and Exciting Big Data Aided Products

The analytics process, batch or streaming, is not a one way process. Analytics also can write data back to the big data storage.

Often times writing data back to the storage makes sense. It allows you to combine previous analytics outputs with the raw data.

Analytics insight can give meaning to the raw data when you combine them. This combination will often times allow you to create even more useful insight.

A wide variety of analytics tools are available. Ranging from MapReduce or AWS Elastic MapReduce to Apache Spark and AWS lambda.

6.4 Display – available

Displaying data is as important as ingesting, storing and analysing it. People need to be able to make data driven decisions.

This is why it is important to have a good visual presentation of the data. Sometimes you have a lot of different use cases or projects using the platform.

It might not be possible for you to build the perfect UI that fits everyone. What you should do in this case is enable others to build the perfect UI themselves.

How to do that? By creating APIs to access the data and making them available to developers.

Either way, UI or API the trick is to give the display stage direct access to the data in the big data cluster. This kind of access will allow the developers to use analytics results as well as raw data to build the the perfect application.

7 Data Science Platform

7.1 Security Zone Design

7.1.1 How to secure a multi layered application

(UI in different zone then SQL DB)

7.1.2 Cluster security with Kerberos

I talked about security zone design and lambda architecture in this podcast: https://anchor.fm/andreask to-Design-Security-Zones-and-Lambda-Architecture–PoDS-032-e248q2

7.2 Lambda Architecture

7.2.1 Stream and Batch processing – available

Batch Processing: Ask the big questions. Remember your last yearly tax statement?

You break out the folders. You run around the house searching for the receipts.

All that fun stuff.

When you finally found everything you fill out the form and send it on its way.

Doing the tax statement is a prime example of a batch process.

Data comes in and gets stored, analytics loads the data from storage and creates an output (insight):

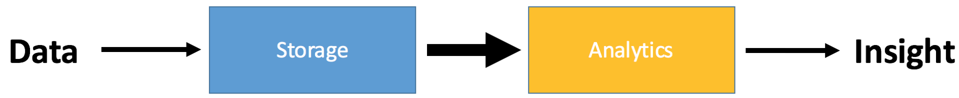

Figure 3: Batch Processing Pipeline

References

[1] J. Ely and I. Stavrov, *Analyzing chalk dust and writing speeds: computational and geometric approaches*, BoDine Journal of Mathematics **3** (2001), 14-159.

List of Figures

List of Tables