Big Data & Finance: Data Streaming and High-Frequency Trading

MSC Dissertation

Supervisor: Professor John Keane
Co-supervisor: Dr Firat Tekiner

2016

A dissertation submitted to The University of Manchester for the degree of Master of Science in the Faculty of Engineering and Physical Sciences

School of Computer Science

Yohan ANDRIANOMENJANAHARY

ABSTRACT

Four Vs traditionally describe data: Volume (the amount of data), Velocity (the speed of update), Variety (the different forms it can take) and Veracity (its accuracy, completeness and quality). Big data is the term used when at least some of those Vs pose a challenge when using usual methods of analysis. But now, clusters of nodes, and distributed programming, make the analysis of large and various datasets possible; Hadoop, was the first widely scalable framework used to develop more easily on clusters.

Machine Learning algorithms represent a new way to discover some and to automatically discover patterns and knowledge, potentially on a greater scale than before and to develop models with more predictive power. However, it turns out MapReduce has inherent flaws which make it unsuited to iterative programming, among which are many Machine Learning algorithms. To overcome these shortcomings, alternative frameworks have been developed. But very few peer-reviewed publications have investigated the performances of those systems for data stream online learning applications yet. Thus, it is currently unclear how scalable are those systems when used for Machine Learning applications, how they use resources and how to efficiently tune the parameters

This project aims to study the behaviour of libraries of Machine Learning tools implemented with distributed-systems frameworks (spark in particular) in the context of real-time streaming data processing. Through this, we seek to better understand the framework and its performances. In order to get a working knowledge of the framework, it is necessary to set up a use case close to a real-life application; we chose High Frequency trading.

Key findings of this project are:

- *Spark streaming limitations are mainly CPU related.*
- *Optimal amount of memory necessary for the system depends on the nature of the feed and batch duration*
- *Sub-seconds latency on a relatively fast stream is unlikely to be attained with spark streaming in our use case.*
- *Forcefully and manually parallelizing the entry data feed with a number higher than the default one is not necessarily a good solution.*
- *Hoeffding trees seem to be able to achieve lower latency than streaming logistic and linear regression.*