business-understanding-ammar-150454388

January 25, 2019

Table of Contents

0.1 Report Introduction

This report documents the Business Understanding stage of the CRISP-DM process followed by this project (Chapman et al., 2000). This stage documents decisions from a business/stakeholder perspective, highlighting the objectives and requirements of the data mining project required to form its problem definition (the specification).

0.2 Organisation and Organisational Objectives

Even though this project is not directly related to any business or an organisation, this project needs to have a beneficial impact to problems faced by stakeholders that were not tackled before in its domain.

0.2.1 Background

Terapixel and high resolution imaging reconstruction and tiling is becoming crucial in many fields, however due to their high resolution nature they become overwhelming in size and difficult to process. (Wang et al., 2015). Due to this, a lot of Terapixel processing applications might require powerful high performance computational resources which might be offered over cloud computing for scalability due to the inconsistency of scientific workflows and for better economics (Agarwal et al., 2011). Heavy workloads like terapixel processing make use of graphical processing units (GPUs), but GPUs generally bottleneck HPC application due to the difficulty of optimising their performance (Madougou et al., 2016).

This project objective is to analyse the bottlenecks and trends of the usage of graphical processing units (GPUs) deployed for a terapixel rendering jobs over a cloud computing platform

(Microsoft Azure). The jobs are involved with a terapixel visualisation of Newcastle upon Tyne by the Urban Observatory, which is used to visualise environmental data throughout the city ("krpano - test," n.d.). The compuational power for the visualisation is provided by public IaaS cloud GPU nodes.

0.2.2 Resources and Data

The datasets provided are based on the results from a run of three 3 jobs (levels 4, 8 and 12 of the terapixel image) using 1024 GPU nodes over the public cloud, recording timings of the rendering, GPU performance, and which image details are being rendering. This data is divided into three separated datasets stored in .csv files:

- gpu.csv: GPU status metrics (temperatures, memory usage, etc.).
- application-checkpoints.csv: checkpoint events throughout render job saved by the application.
- task-x-y.csv: Co-ordinates of image parts being rendered in tasks carried by the application for the job.

0.2.3 Success Critera and Objectives

To obtain what criterion should the models pass, a look into what stakeholders on the field expect is needed. To do this, a look into published work in the area is necessary.

When it comes to GPU performance, (Madougou et al., 2016) states that parallelism is crucial (being that GPUs are parallel devices by nature), and memory access and usage is also key to unlocking the chips performance. However, the rest of the paper (similarly to other on similar topics) concentrates on developing a modelling approach to recoginsing bottlenecks instead of concentrating on the specific underlying causes.

When it comes to GPU performance with Cloud and Terapixel processing in the picture however, it is difficult to find academic literature that covers all three topics. Nonetheless, (Yamaoka et al., 2011) suggests that a considerable bottleneck occured in the benchmarks carried for a high resolution interactive application due to disk to GPU memory load time which was spotted through frame rate analysis. Moreover, other caveats to watch for when cloud computing enters the picture for GPU analysis, is the importance of how well the application performs when scaled is very critical as scientific data is scalable in nature (Agarwal et al., 2011).

To sum up, the project will need to meet the following criteria:

- Analyse how well the cloud application parallelises the workload
- Measure and analyse memory usage
- Measure how well the GPUs respond to varying data scales and loads

0.3 Project Specification and Plan

0.3.1 Requirements and Rationale

- Clean the data in the provided datasets
  - Reasoning: Prepare the data for exploration by removing data that might distract the analysis
- Merge and Aggregate the data in the cleaned datasets

  - Reasoning: Prepare the final dataset for exploration by collecting and aggregating required fields

- Perform Exploratory Data Analysis
  - Reasoning: Gives clue about data structure and relationships

0.3.2 Tools

Analytics and modelling steps will be performed using python code and libraries (e.g. matplotlib), which are tested with the pytest library and reported on using python notebook (literate programming). Documentation for the code will be managed using reStructuredText based Sphinx and saved as HTML pages ("Overview — Sphinx 2.0.0+/4f37b33 documentation," n.d.). The project structure template is created using a Cookie Cutter Data Science hierarchy ("Home - Cookiecutter Data Science," n.d.) which provides a Cookie Cutter and Python based environment to work with. The entire project hierarchy is version controlled using git and GitHub.

0.3.3 Project Plan and Risks

As stated in the introduction to this report, the project will follow the CRISP-DM hierarchical cyclic process model for data mining. The process is typically done iteratively in order, but, flexibility is possible (Chapman et al., 2000). The flexibility will be used to carry Data Preparation before Understanding, and to skip the modelling, deployment and evaluation steps as well since no modelling will occur in this project (no final dataset to setup, only EDA).

Test Driven Development will be carried using PyTests and literate programing (Jupyter Notebook) will be used for the analysis documentation. These tools ensure that both tasks can be carried simultaneously with the code, which reduces the risk of testing and documentation are cut or rushed when time constraints occur.

GitHub is used for version control to ensure that any risk of changes causing a difficult to revert change is minimised through version control. Moreover, its use ensures that the project development route can branch and change easily without worry over the risk of not being able to revert back when the changes are scrapped.

0.4 References

1. Agarwal, D., Cheah, Y.-W., Fay, D., Fay, J., Guo, D., Hey, T., Humphrey, M., Jackson, K., Li, J., Poulain, C., Ryu, Y., van Ingen, C., 2011. Data-intensive science: The Terapixel and MODIS-Azure projects. The International Journal of High Performance Computing Applications 25, 304–316. https://doi.org/10.1177/1094342011414746
2. Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., Wirth, R., 2000. CRISP-DM 1.0 Step-by-step data mining guide.
3. He, G., Wang, H., Li, E., Huang, G., Li, G., 2015. A multiple-GPU based parallel independent coefficient reanalysis method and applications for vehicle design. Advances in Engineering Software 85, 108–124. https://doi.org/10.1016/j.advengsoft.2015.03.006
4. Home - Cookiecutter Data Science [WWW Document], n.d. URL https://drivendata.github.io/cookiecutter-data-science/ (accessed 1.4.19).
5. Madougou, S., Varbanescu, A.L., Laat, C.D., Nieuwpoort, R.V., 2016. A Tool for Bottleneck Analysis and Performance Prediction for GPU-Accelerated Applications, in: 2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW). Presented

at the 2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), pp. 641–652. https://doi.org/10.1109/IPDPSW.2016.198
6. Overview — Sphinx 2.0.0+/4f37b33 documentation [WWW Document], n.d. URL http://www.sphinx-doc.org/en/master/ (accessed 1.4.19).
7. Wang, C.-W., Huang, C.-T., Hung, C.-M., 2015. VirtualMicroscopy: ultra-fast interactive microscopy of gigapixel/terapixel images over internet. Scientific reports 5, 14069. http://dx.doi.org.libproxy.ncl.ac.uk/10.1038/srep14069
8. Yamaoka, S., Doerr, K.-U., Kuester, F., 2011. Visualization of high-resolution image collections on large tiled display walls. Future Generation Computer Systems 27, 498–505. https://doi.org/10.1016/j.future.2010.12.005
9. krpano - test [WWW Document], n.d. URL http://terascope.di-projects.net/cloudviz-tile-storage/vtour/index.html (accessed 1.12.19).