Ben J Settle IV

05/12/2023

Professor Cates

Intro to Data Science

FINAL PROJECT DOCUMENTATION

This document will outline the project summary, the methods used throughout the project, a section on results and a short discussion of what these findings indicate, a section on what has been learned and finally a conclusion will be drawn on the project's outcome.

From Kaggle.com, I downloaded a dataset that contained data on Intel, AMD, and NVIDIA CPUs and GPUs. I wanted to use the dataset to see what kinds of differences and trends I could notice in the data between different vendors, and to try and determine through the numbers if there was a definitive winner for best CPU vendor and best GPU vendor based on performance specifications.

The 'Deliverable' for this project can be found in the Visualizations folder of this repository, in a file named ChipsetAnalysis.pdf, which contains eight visualizations of the data.

For instructions on which order to execute the Jupyter Notebook files in, please refer to ABOUTME.txt

See below for continued documentation

PROJECT SUMMARY & METHODS

This section of the report will discuss the overall summary of the project, as well as the methods used for data preparation and cleaning before the data was visualized. The summary and methods will be split into two separate sections.

Project Summary

The goal of this project was to determine which CPU / GPU vendor makes the best products, and to do this I searched for a dataset that contained performance metrics on a range of hardware from different vendors. I found a dataset containing this information on Kaggle.com, which I downloaded and began to analyze the data in order to determine the best approach for cleaning and processing it.

In order to better manage the many changes, I knew I would likely make to the code on this project, I decided it would be best to use VCS. I uploaded the project to GitHub and used the VCS support on Visual Studio Code to manage the repository and the project.

After the code was complete and the data was cleaned, processed and some of it binned, I was ready to visualize it using third party software. I decided Tableau was the best option for the data I was working with, and I added a Tableau workbook to the main branch of my repository.

In total, I created eight visuals to represent different aspects of the data. Four visuals were on CPU data (Intel vs AMD) and four visuals were on GPU data (AMD vs NVIDIA). The visuals seemed to indicate that Intel and AMD are in a close race, but AMD has been gaining on Intel's dominance over the market recently. However, the GPU market is where AMD falls short, and NVIDIA takes control. Both companies offer quality products, but NVIDIA has better perfected the technology of GPU chip manufacturing.

Project Methods

I immediately noticed that because the original dataset contained both CPUs and GPUs, there were some columns for which the CPUs had no entry, because the information was not relevant to CPU metrics. This would pose a problem if I wanted to drop null values from the dataset, as the pandas.Dataframe.dropna() method would assume that all entries for CPUs contained missing information. To remedy this issue, I decided to split the dataset into two separate pandas data frames, one for CPUs and one for GPUs. Using the entries for the 'Type' column I split the data frame by the two hardware types and reset the index entries for both data frames. The new data was saved to two separate csv files.

I now had two data frames to work with and could clean and prepare them without having to worry about interfering with attributes and relevant information for the other part of the dataset. For the CPU data I dropped all three columns referring to GFLOPS and the columns for

product name, release date, and foundry. The same was done for the GPU data except the GFLOPS columns were left in the data. Rows containing null values were dropped from the data frames and once again index entries were reset. The new data was saved to two new separate csv files.

With two separate clean data frames I could now determine if there were any outliers in my data. To determine this, I used the .describe() method to print the descriptive statistics for both data frames. Using the standard deviations, the 50^{th} and 75^{th} percentiles as well as the max values for each dataset attribute, I searched for columns that contained obvious outliers in the data and found a few columns that had clear outliers that should be removed. Using the Inter Quartile Range of these columns I was able to generate a lower and upper limit to the datapoints that I applied to the entire column, removing any data outside of these limits from the tables. The new data was saved to two new separate csv files.

Finally, I decided to bin some of the data so that I could have more options for visualizing the properties of the data frames. I chose a few attributes at random from each data frame and used the pandas.Dataframe.qcut() method to separate the data into quartiles for four distinct bins. When this was complete, the data was saved to two new csv files. I now had data that was ready for visualization and imported the two csv files that had all the previous operations performed on them into a Tableau workbook to create my visuals. The CPUs and GPUs were evaluated on the following metrics:

CPUS:

- Process Size (Nanometers)
- Thermal Design Power, aka TDP (Watts)
- Die Size (mm^2)
- Transistors (Millions)
- Frequency (MHz)

GPUS:

- Process Size (Nanometers)
- Thermal Design Power, aka TDP (Watts)
- Die Size (mm^2)
- Transistors (Millions)
- Frequency (MHz)
- FP16 GFLOPS (Billions)
- FP32 GFLOPS (Billions)
- FP64 GFLOPS (Billions)

PROJECT RESULTS & DISCUSSION

This section of the report will discuss the findings from the data visualization section of the project as well as discuss what these findings mean. The Results and Discussion will be split into two different sections.

Project Results

Although it can not be said for certain whether these findings are representative of real-world performance, we can infer a fair amount about each vendor's overall methods for creating and designing hardware. When we visualized the CPU data, there did not appear to be any obvious winner at first. Both Intel and AMD are competent and respected chip manufacturers and there did not seem to be much of a gap between them. However, upon further inspection we can see that Intel seems to be more consistent with their technology than AMD, with the former following what could be assumed to be an iterative design approach to CPUs and the latter taking more risks and maybe switching their designs up more often. We can see this in the fact that intel usually produces the same amount of chips for each generation of transistor ranges. AMD on the other hand seems to produce whatever is necessary to meet demands at that moment in time.

Additionally, Intel's newest chips are more thermally adaptive than AMD's newest chips are. Despite this AMD has the advantage in volume, where they produce significantly more, albeit slightly less advanced, CPUs than intel does. Despite the clear advantage that Intel currently holds over the CPU market, the data does indicate that AMD has been slowly catching up to intel in areas of higher performance.

Discussion

As for our GPUs there was a clearer and more definite winner in this category, which should come as no surprise since NVIDIA has been dominating the market for decades now. In almost every aspect NVIDIA GPUs beat AMD's and often by a sizeable margin. There were some places where AMD kept up an a few where they even managed to exceed NVIDIA, such as Thermal Design Power, where up until recently AMD had been ahead of NVIDIA for quite some time.

As far as winners go, I would have to say Intel and NVIDIA are the clear champions of their respective categories. Now this isn't to say that AMD makes bad products, the opposite is true, and in fact the results of this analysis are not intended to sway anyone to purchase from one vendor or the other, but rather to inform the end user what kind of hardware they're paying for.

As unlikely as it is to not run into any issues, the programming and visualization went incredibly smoothly for this project. I can only assume this is because I used GitHub for version control and issue resolution, making the process much smoother than trying to manage every little problem at once.

Conclusion and Takeaways

Given that the project went smoothly, I was able to make a decent number of visuals with the data and the fact that the data was representative of the general attitude and atmosphere around each of the three companies, I would call this project a success. I really enjoyed using Tableau to visualize my data, it is a fantastic tool and in my opinion beats PowerBI any day of the week. It was a learning curve trying to figure out where to drag the Measure Names and Values, but once I got it down it was incredibly rewarding and satisfying to see my data come to life after all the work I put into it.