Analysis of Linux applications using perf

The main task was to analyse any three Linux application using perf based on factors such as number of instructions executed, number of cache references, number of cache misses and number of branch miss-predictions.

We took Firefox, gedit and chrome for doing the task and ran each of the applications 10 times. We took the analysis of the points for every 10 milliseconds. We considered each interval of 10ms as an individual phase and considered the above mentioned factors as a 4 dimensional point and found the cosine similarity between them.

We wrote Hadoop map function for finding cosine similarities between the phases and alongside we even calculated how many points were above threshold and how many were below and reduced it to get the total word count.

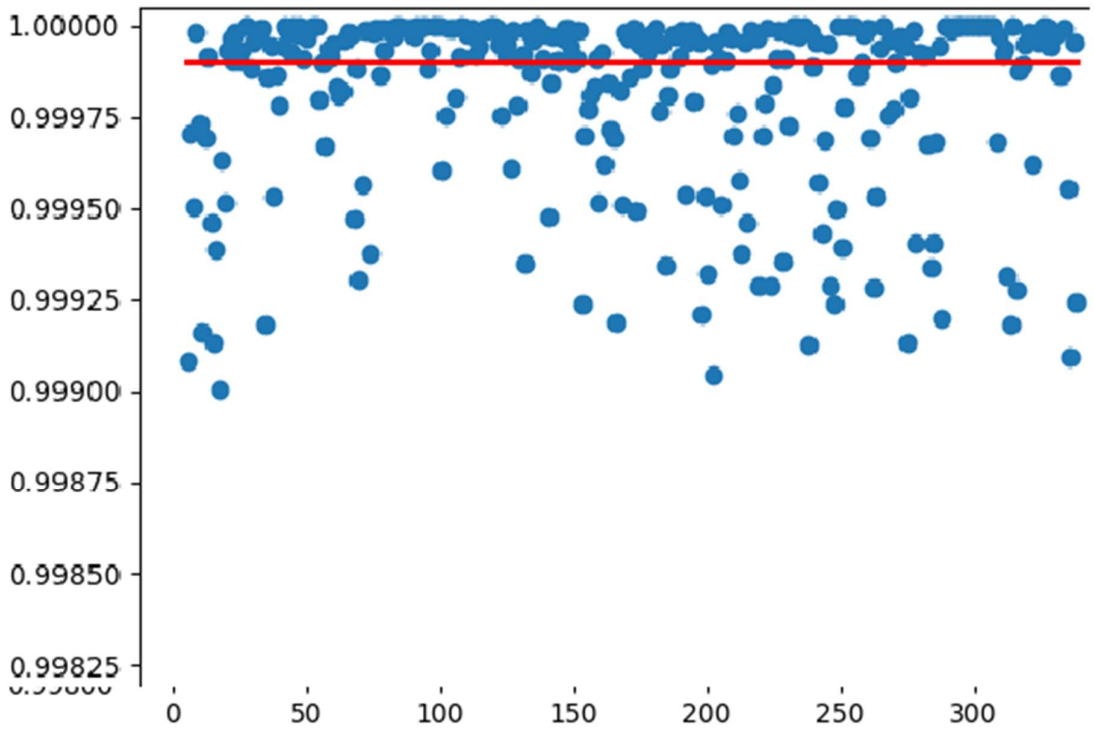
We even cleaned the data. That is put the acquired points to a data-frame and from there we passed it to the mapper function.

Using these cosine points we plotted a scatter plot with a predefined saturation level of 0.9999

Perf commands:

1) Command to get all performance details of a particular application(ex : Firefox)
   perf stat –d firefox
2) Command to run get details after regular interval of time(ex 10ms)
   perf stat –d –I 10 firefox
3) Command to get only selected points
   perf stat –I 10 –e instructions –e branch-misses –e L1-dcache-loads –e L1-dcache-load-misses
4) Storing all the information into a file
   perf stat –I 10 –e instructions –e branch-misses –e L1-dcache-loads –e L1-dcache-load-misses 1>error 2>report

Graph



Here x-axis depicts the interval number and y-axis depicts the magnitude which is got by applying cosine similarity and the red line depicts the saturation point which is 0.9999

Conclusion:

When we saw the cosine similarities between the phases in three application they were almost similar.

Team Members:
Saahitya Edamadaka(01FB16ECS322)
Sanjay K A(01FB16ECS341)
Shalini Sai Prasad (01FB16ECS350)
Shashank M Bongale(01FB16ECS355)