Deep Learning Model to Predict Hardware Performance

Shrreenithi Srinivasan[1], Avik Satrughana Nayak[2]

[1]*BE Computer Science, Anna University, Chennai, India*
[2]*BE Computer Science, Biju Patnaik University, Bhubaneswar, India*

Abstract – *The goal of the project is to harvest the power of machine learning algorithms and come up with a novel approach towards predicting the best possible combination of hardware features for a computer system, (e.g. microprocessor/memory specs, Software specs etc.) such that the baseline and peak scores measured on a certain benchmark program (e.g. CINT 2006 of SPEC benchmark program) is maximized. The problem involves predictive analysis and optimization of hyper parameter for high accuracy output. The baseline score is the target value. The hardware features are the independent variables and the baseline is the dependent variable.*

Key Words: **Multi Linear Regression, Pandas, Sklearn package, Backward Elimination, Cross Validation, Logistic Regression, One hot Encoding, Forward Selection.**

1. INTRODUCTION

An application that is able to predict hardware performance and analysis with Deep Learning (DL) algorithms, could drastically improve the performance/ cost curve of a system of machines. An analysis is needed to verify the result with these expectations. Doing it manually takes time and this void is filled by a model which predicts part of the analysis which is the performance that can be expected with a set of input configurations. Performing this type of analysis by hand can require a day or two to complete per benchmark. Thus a Machine Learning (ML) model is used to do robust analysis and predict results in a fraction of time. The Deep Learning model takes input data such as CPU type, frequency, number of cores, memory size and speed, flash or disk architecture, network configuration that correlates against the corresponding performance and system response. For this project, SPEC CPU 2006 and SPEC CPU 2017 (Reference: 1) will be selected from the PerfKitBenchmarker suite of industry standard cloud platform benchmarks. These two Spec benchmarks are the most popular performance tools in repository which will be downloaded to PostGresQl for Deep Learning analysis. It contains SPEC's next-generation, industry-standardized CPU intensive suites for measuring and comparing intensive performance, stressing a system's processor, memory subsystem and compiler. Once a Deep Learning model is chosen and fully implemented, it will have the ability to infer a score from a given hardware configuration or infer a hardware configuration from a given score. This Deep Learning model is used to infer a benchmark score given a hardware configuration.

2. RELATED WORK

The Deep Learning project is a green field project given by Flex Cloud Labs to students to explore and implement machine learning models. Thus there is no previous work done in the domain as per our investigation.

3. METHODOLOGY

The approach involves training a Deep Learning model that takes input as the machine parameters and gives the baseline as output and also maximizes accuracy. To arrive at an accurate model, we followed a scientific approach towards optimizing a multi-variable problem involving following steps.

3.1 Exploratory Data Analysis (EDA)

This is the process of developing an intuition of the input dataset, analyzing the distribution of mean, median and standard deviation of scores (in our case the baseline scores of machine performance), and coming up with an initial heuristics based model that provides some result (which need not be accurate), but works within the framework of existing models.

3.2 Data cleanup

Perhaps the most challenging and major portion of developing a machine learning model is cleaning up the input data set. Some studies show that more than 80% of the work involving the development of Machine Learning (Reference: 2), involves data cleanup. This is a critical and often overlooked step, and is of utmost importance since noise and invalid data will create model parameters that have little to no correlation to the expected model. We spent a significant amount of effort in data cleanup using approaches we explain in the next section.

3.3 Model Fitting

Once we are confident that the dataset has been sufficiently organized, it is now time to fit a model into our data set. This requires that we split our dataset into test set and training set and use the training set to generate a model, and use the test set to measure the accuracy of our model and fine tune its parameters if necessary. Models used in predicting hardware performances are one hot encoding using logistic regression, multi-linear regression using label encoding, backward elimination, forward selection and cross-validation.

4. IMPLEMENTATION OF THE DL MODEL

Jupyter notebook, Numpy, Pandas, Sklearn Package and Python programming language are used to implement this work. Implementation of deep learning project is done as follows:

- Data pre-processing (Clean-up, Encoding, Feature analysis)
- Changing test and training data size
- Algorithm selection & model experimentation (Linear Regression, Lasso, Logistic etc.)
- Model tuning
- Experiment with Cross validation, Backward elimination, Forward selection.
- Experimenting with different permutation of above mentioned factors.

In the area of data analysis, we explored the relationship between various parameters like hardware, vendor, system, num_cores, num_chips, processes, num_of_cores per_chip, auto parallelization, num_of_threads_per_core, cpu-orderable, processor characteristics, base pointer size, peak pointer size, first level cache, second level cache, other cache, memory, operating system, file system on baseline score. As part of the data analysis we did lot of research on the datasets and gained insight about the extent of dependencies with respect to vendor (Figure 4.1), num_cores (Figure 4.2), system (Figure 4.3), processor (Figure 4.4) and speed (Figure 4.4).

```
test_df.groupby(['vendor']).mean()['baseline'].nlargest(20).plot(kind='bar',figsize=(20,6))
<matplotlib.axes._subplots.AxesSubplot at 0x1a11d9b240>
```

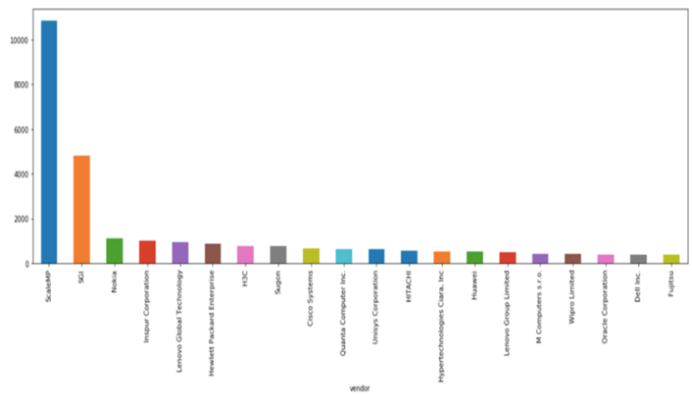

Fig - 4.1: Study on the distribution of different vendors with respect to baseline score

```
test_df['num_cores'].value_counts().plot(kind='bar',figsize=(20,6))
<matplotlib.axes._subplots.AxesSubplot at 0x1a1157a160>
```

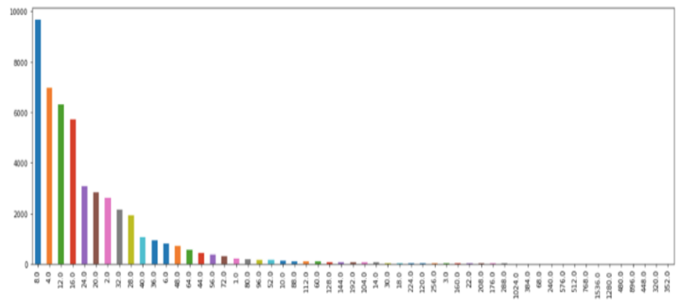

Fig - 4.2: Study on the distribution of num_cores with respect to baseline score

```
In [25]: test_df.groupby(['system']).mean()['baseline'].nlargest(20).plot(kind='bar',figsize=(20,6))
Out[25]: <matplotlib.axes._subplots.AxesSubplot at 0x1a12c11cc0>
```

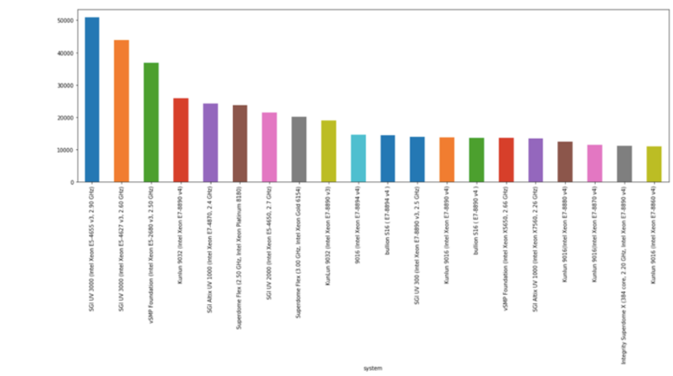

Fig - 4.3: Study on the distribution of system with respect to baseline score

```
test_df.groupby(['processor']).mean()['baseline'].nlargest(20).plot(kind='bar',figsize=(20,6))
<matplotlib.axes._subplots.AxesSubplot at 0x1a11dc0ac8>
```

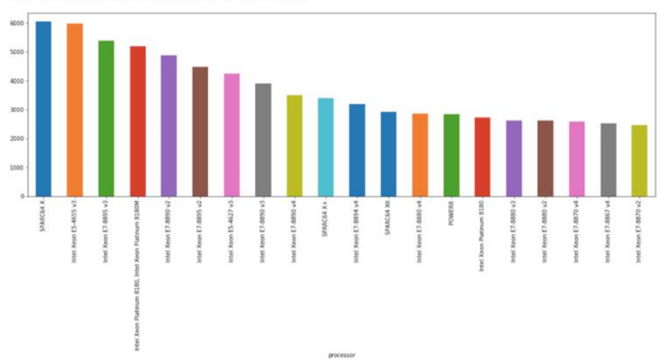

Fig - 4.4: Study on the distribution of processor with respect to baseline score

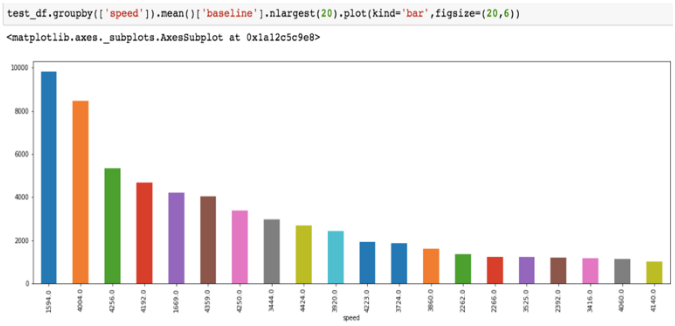

Fig - 4.5: Study on the distribution of speed with respect to baseline score

After implementation, we came up with the clarification that, cross validation score is getting nearer to training score with respect to number of datasets (Figure 4.6)

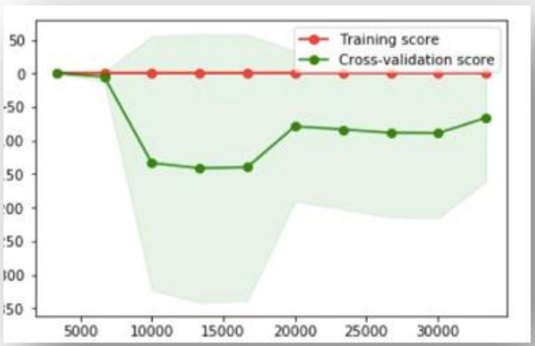

Chart-4.6: Result of cross validation score with respect to training score

5. RESULTS

We explored different machine learning models such as, multinomial regression, linear regression, logistic regression, multi linear regression, multi linear lasso regression and implemented them to preprocessed dataset (preprocessing involves data cleaning, encoding dataset using one hot encoding and label encoding) and also used various machine learning algorithms like backward elimination, forward selection (Figure 5.1) and found out the different results which are mentioned below:

- Using Backward Elimination (all features- model linear regression) – 80.82%
- Using Forward Selection (all features- model linear regression) - 75.3%
- Using Cross Validation (only continuous feature- multi linear regression) - 80.77%
- Using Cross Validation (only continuous feature- multi linear lasso regression) -78.1%
- Multi Linear Regression (only continuous feature)- 77.64%.
- One Hot Encoding using Logistic Regression (categorical & all continuous feature) – 50.8%

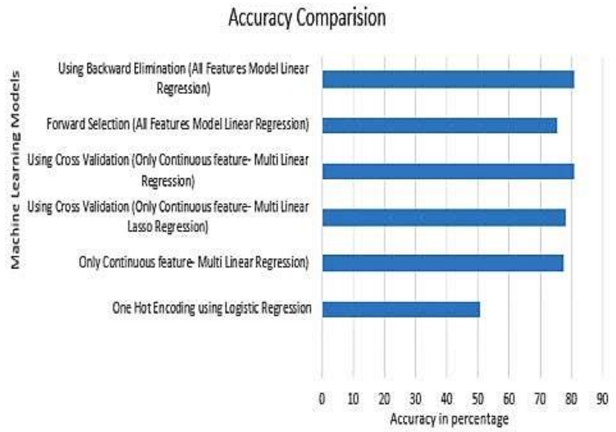

Fig-5.1: Results of models in predicting score

6. CONCLUSION

Key takeaways in this analysis of deep learning model for predicting hardware performance are listed as follows:

- Data preprocessing is the main and very important task when handling with more number of data in the dataset.
- For Datasets that consists of numeric, alpha numeric and strings, it is better to use label encoding and one hot encoding to convert all the alpha numeric and strings to numeric before passing on to the model. In this project, both label and one hot encoding is used to get the better results.
- It is always better to test with different regression models to check the accuracy and compare the performance. In this project, we have used linear regression, logistic regression, multi linear lasso regression.
- Backward elimination and forward selection helped in getting good accuracy.
- Best accuracy can be achieved by using backward elimination and linear regression model with encoding the features.

ACKNOWLEDGEMENT

We would like to thank our Professor Ms. Vijayalakshmi for her continuous guidance and encouragement extended to me. We also like to express our sincere gratitude to the department for the support and providing the necessary materials.

REFERENCES

[1] Ould-Ahmed-Vall, E., Doshi, K. A., Yount, C., & Woodlee, J. (2008). Characterization of SPEC CPU2006 and SPEC OMP2001

[2] https://www.ibm.com/blogs/bluemix/2017/08/ibm-data-catalog-data-scientists-productivity/

[3] H. Shakouri |G.R. Nadimi | F.Ghaderi Year 2007: Fuzzy linear regression models with absolute errors and optimum uncertainity

[4] Li Zhang | Ruizhen Wu | Yintang Yang Year 2013: A high-speed and low-power synchronous and asynchronous packaging circuit based on standard gates under four-phase one-hot encoding

[5] Fort Lee, N.J (Business Wire): ScaleMP February 22, 2016 https://www.businesswire.com/news/home/20160222005378/en/ScaleMP-Ranks-No.-1-Standard-Performance-Evaluation