

CS 331 - MACHINE LEARNING

ASSIGNMENT-1



REPORT

TITLE : Performance difference between Python and C++ implementations

Team Members :

1. ANIKET CHAUDHRI (2003104)
2. ADARSH ANAND (2003101)
3. SOMESH AGRAWAL (20033--)

AIM

The main aim of this lab assignment was to explore the relative performance difference between Python and C++ implementations of matrix multiplication.

OVERVIEW

This report presents the findings of a lab assignment that aimed to explore the relative performance difference between Python and C++ implementations of matrix multiplication. The team, AdAnSo, was tasked with creating a C++ library to perform matrix multiplication, a Python function for matrix multiplication without using numpy, and using numpy to multiply matrices. Additionally, a Boost C++ library was created and invoked from numpy to perform matrix multiplication

TEAM

The team, AdAnSo, was composed of 3 members:

- **Adarsh Anand (2003101)**
- **Aniket Chaudhri (2003104)**
- **Somesh Agrawal (2003326)**

Adarsh Anand was responsible for implementing the C++ matrix multiplication. He also assisted in the creation of this document.

Aniket Chaudhri was responsible for reading and writing the matrices from file and also assisted in the implementation of the C++ library and the Boost C++ library. He also assisted in the implementation of the C++ library.

Somesh Agrawal was responsible for reading and writing the matrices from file and also assisted in the implementation of the C++ library and the Boost C++ library.

TOOLS USED

- C++ programming language
- Boost C++ libraries (boost::python and boost::python::numpy)
- Python programming language
- Numpy library
- Text editor or IDE (such as Visual Studio or Sublime Text)

IMPLEMENTATION

To implement the C++ library, we used the Boost library which allows the creation of C++ libraries that can be called from Python and vice versa. We used the numpy module of Boost to convert the input matrices from numpy ndarray to C++ vectors. This allowed us to perform matrix multiplication on the vectors using a nested for loop and then return the result.

For the Python function, we implemented matrix multiplication without using numpy by creating a nested for loop, similar to the C++ implementation. We also used the numpy dot method to multiply the matrices and compared the results to our previous implementations.

SCREENSHOTS OF OUTPUTS

```
aniket@aniket:~/Documents/lectures/cs331/Lab1$ g++ matrixmain.cpp
aniket@aniket:~/Documents/lectures/cs331/Lab1$ ./a.out
Time taken by function: 2.53544 seconds
```

Fig1. Matrix multiplication in C++

```
aniket@aniket:~/Documents/lectures/cs331/Lab1$ conda activate cs331
(cs331) aniket@aniket:~/Documents/lectures/cs331/Lab1$ python matrixboost.py
hello
Time taken: 2.914827346801758 seconds
```

Fig2. Matrix multiplication by invoking C++ using Python Boost

```
starttime = time.time()
matrixmul(A, B)
endtime = time.time() - starttime
print('Time taken: ', endtime, 'seconds')
✓ 1m 52.1s
Time taken: 112.16420698165894 seconds
```

Fig3. Matrix multiplication using for loops in Python

```
starttime = time.time()
numpymatrixmul(A, B)
endtime = time.time() - starttime
print('Time taken: ', endtime, 'seconds')
✓ 0.3s
Time taken: 0.00982975959777832 seconds
```

Fig3. Matrix multiplication using Numpy

RESULTS

Our findings showed that the C++ implementation was faster than the Python implementation without numpy, but slower than the numpy implementation. The boost C++ library and numpy implementation was the fastest of all. We also summarized our findings in terms of the performance in a report that was submitted.

REFERENCES

- Boost library documentation: https://www.boost.org/doc/libs/1_74_0/libs/python/doc/html/index.html
- Numpy documentation: <https://numpy.org/doc/stable/>
- C++ for Python developers: <https://www.learncpp.com/cpp-tutorial/a001-python-and-c-a-side-by-side-comparison/>
- Matrix multiplication in Python: <https://www.geeksforgeeks.org/matrix-multiplication-python/>
- Boost.Python tutorial: https://www.boost.org/doc/libs/1_74_0/libs/python/doc/html/tutorial/tutorial/hello.html
- C++ matrix multiplication: <https://www.geeksforgeeks.org/matrix-multiplication-c/>

END OF REPORT