Leveraging Parallel Processing for Advanced Graph Computations

Kaustubh Shivdikar, *PhD Student, Northeastern University*
Shamil Imakaev, *REU Participant Student, MassBay Community College*
Professor Kaeli, *College of Engineering, Northeastern University*

Northeastern University
College of Engineering

Northeastern University
Center for STEM Education

Abstract:

Goal:

- Dive into the core mechanisms of GNNs
- Approach various projects that reflect the idea of GNNs such efficient organization of data and scalability.
- This research is not just an exploration of GNNs, but an important step towards future innovations in the era of big data and AI.

Background:

Parallel Computing:

- Computation, many calculations or processes are carried out simultaneously
- Leverages power of multiple computing resources(in our case, the CPU) to solve large problems more efficiently.

Python:

- Learned basics of Python during our first few weeks to carry out our research.

Jupyter Notebook:

- create and share documents containing live code, equations, visualizations
- used for data cleaning, statistical modeling, machine learning

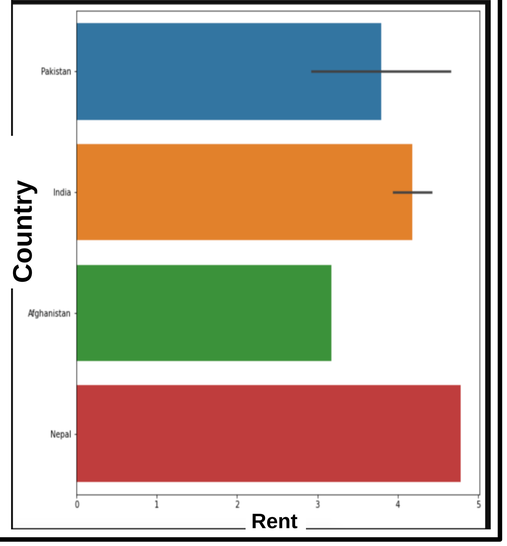
Projects

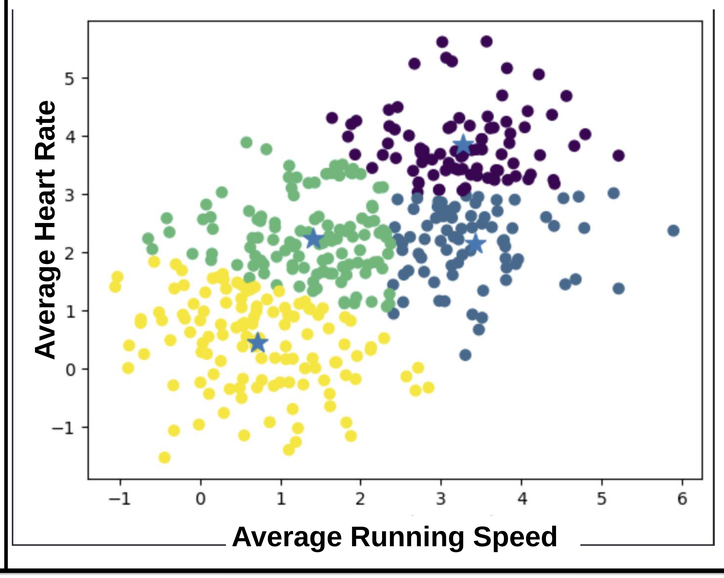
Cost of Living Index

Goal: An introduction to Jupyter notebook and Data visualization.
Results: Visualized data by using matplotlib, such as bar graphs and scatter plots.
Sample:

```
# lets try and do least expensive countries here
country_cheap = df.sort_values('Rent', ascending =True)

from matplotlib import pyplot
fig, ax = pyplot.subplots(figsize=(10,10))
ax = sns.barplot(x='Rent', y='Country',data = country_cheap.head(30))
```





Multi threaded parallel computing

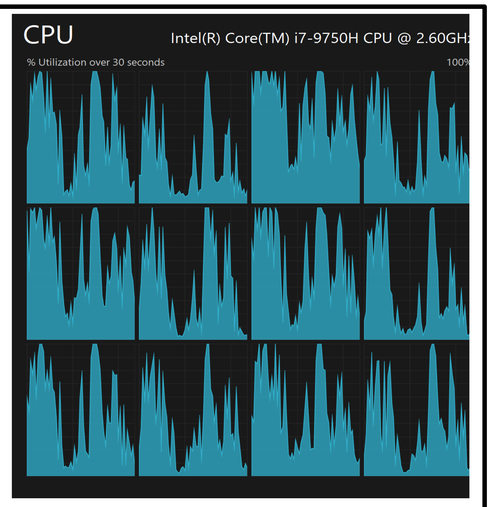
Goal: Used all cores of a CPU to run simultaneously when solving a matrix multiplication problem.
Results: The algorithm solved the problem efficiently
Reasoning: To gain a better understanding of using multi threading instead of single threading for more efficient results

```
matrix_A = np.array([[1,2,3], [4,5,6], [7,8,9]])
matrix_B = np.array([[1,2,3], [4,5,6], [7,8,9]])
```

- Given an array of two matrices
- The algorithm quickly solves the multiplication of the two Matrices to get the (3,3) matrix below

```
[[ 30.  36.  42.]
 [ 66.  81.  96.]
 [102. 126. 150.]]
```

- When running the code, we can see the CPU using all of its cores to run the multiplication problem simultaneously instead of doing each operation individually



- Program divides up the workload throughout each core of the CPU

K-Means Clustering

Goal: Group similar data points together and discover underlying patterns. K-means looks for a fixed number (*k*) of clusters in a dataset.

•Process:

•The K-means algorithm groups data points together based on how similar they are.
•It does this by first randomly selecting k points, called centroids, to represent each cluster.
•Then, it assigns each data point to the cluster with the nearest centroid.
•Finally, it updates the centroids to be the average of all the data points in their respective clusters.
•This process repeats until the centroids no longer change.

Summary

Parallel Computing:

•Important operation in GNNs as they often involve multiple nodes and edges. The application of parallel computing in matrix multiplication shows how GNNs perform operations on different nodes simultaneously

Data Visualization:

•GNNs deal with complex, high-dimensional data. Understanding this data and the results of GNNs can be done through visualization. The skills learned in creating bar graphs and scatter plots can be applied to visualize GNN results.

Multi-threading:

- Crucial for efficient GNN operations. The Matrix Problem gives a better understanding of efficient GNN computation

K-Means Clustering:

- Method that clusters data based on similarity, shows how GNNs group nodes in the network. Understanding K-means clustering helps in comprehending how GNNs can classify nodes or communities in a network.

NSF REU-Site: Research Experience for Undergraduates site to create "Pathways Opening World Energy Resources" (REU - POWER)

This work was supported by the National Science Foundation Grant ##2150417