Sandesh Patnurkar

+1-(716)-381-3599 | sandeshv@buffalo.edu | sandeshvp.github.io | www.linkedin.com/in/p-sandesh

SUMMARY

A driven and dedicated individual with a will to make a difference and persistence in problem-solving and finding optimal solutions, I'm an emerging Full Stack Developer with Master's in Computer Science from University at Buffalo. I have an expertise in Java, Python and SQL with good communication and interpersonal skills to complement them. I'm looking for a challenging full time opportunity where my skills are tested and put to good use.

EDUCATION

University at Buffalo, State University of New York, NY   **GPA 3.47/4.0**
Master of Science, Computer Science and Engineering   **Feb 2019**
Courses: Analysis of Algorithms, Data Mining and Bioinformatics, Data Intensive Computing, Computer Security, Computer Vision and Image Processing, Database Systems

Government College of Engineering Aurangabad, India   **GPA 6.875/10**
Bachelor of Engineering, Information Technology   **June 2017**
Courses: Data Structures, Object-Oriented Programming, Programming in Java

COMPUTER SKILLS

Programming: Java, Python, HTML, CSS, SQL, JavaScript, MATLAB, R, VHDL, PHP, MongoDB
Frameworks: Bootstrap, Hadoop, Angular, jQuery, WordPress, RESTful Web Services
Operating Systems: Linux, Windows, OSX
Tools and Services: LaTex, Git, Docker, Tableau

ACADEMIC PROJECTS

- **'Relational Query Engine'** – *Java, SQL*   **Feb - May 2018**
  - Built a Relational Query Engine in Java using JSQLParser to evaluate SQL queries
  - Performed optimizations such as selection pushdown, projection pushdown and join optimizations to improve the throughput of the evaluator and provide output faster.
  - Evaluated cross operations with 50% better efficiency in terms of memory usage.

- **'Implementation of PCA and Apriori Algorithms'** - *Python, NumPy, Pandas*   **Sep - Dec 2018**
  - Reduced higher dimensional data to 2 dimensions by implementing PCA algorithm from scratch in Python
  - Compared PCA algorithm with SVD and t-SNE algorithms by plotting the 2 dimensional data achieved by execution of respective algorithms
  - Performed association rule mining using the Apriori Algorithm to generate rules for a gene dataset

- **'Classification algorithms'** – *Python, NumPy, Pandas*   **Sep - Dec 2018**
  - Classified data (both discrete and continuous) using nearest neighbor and naïve bayes algorithms that were implemented in Python from scratch (without scipy implementation) with the help of NumPy and Pandas library
  - Performed 10-fold cross validation to evaluate performance metrics of both algorithms
  - Prepared a comprehensive technical report on the summary of the project outlining the detailed analysis of working of the code and provided step-by-step instructions as to how to execute the program with different data input files.

- **'Data Aggregation, Big Data Analysis and Visualization'** – *R, Python, Hadoop MapReduce, D3.js*   **Feb - May 2018**
  - Used the twitter API to extract tweets on a topic of interest using R
  - Used Python to extract NY times articles using NY times API and BeautifulSoup library
  - Processed the tweets and articles using Python then ran them through Hadoop's MapReduce framework
  - Inspected most frequently occurring words data and visualized them in a word-cloud that was developed using D3.js

- **'Data Collection and Exploratory Data Analysis'** – *R*   **Feb - May 2018**
  - Used R to extract tweets on a topic of interest with the help of Twitter API
  - Extracted the geolocation from the metadata associated with the tweets. In cases where the tweets were not tagged with a geolocation, the users were looked up and the location mentioned in their profile was used
  - Extracted location data and plotted a heat map using the state-specific twitter data using Google Maps API

- **'To-Do App – MEAN'** – *MongoDB, ExpressJS, AngularJS, NodeJS*   **May – June 2018**
  - Built a To-Do app using NodeJS as backend and AngularJS as frontend which was connected using ExpressJS. Used MongoDB as the database.
  - Improved the front end design using Google Material Design Lite. Designed it as a Student- Teacher application wherein Students could only read the To-Dos and Teachers could add, delete and update the to dos

LANGUAGE SKILLS

English, Hindi, Marathi, Japanese (N5 level), German (A1 level)