Parminder Singh

Data Scientist and Full Stack Engineer

parmsingh129@gmail.com
(+91) 941 776 3589

Website: trion.me
Github: Trion129
Blog: medium.com/@trion

EXPERIENCE

Chingu Cohorts, Remote — *Data Scientist*

1 YEAR

- Implemented tools for classification and detection of applicants for maintaining best quality of community.
- Used Sentiment and Clustering analysis for detecting churn and productive behaviours via application surveys and team discussions.
- Used Scikit learn and NLTK for text analysis and Pandas for performing data transformations.
- This led to an increase in productivity of community as the inactive team members decreased by 80% and the members that disappeared suddenly, were retrieved back because of early detection.

Creatella, Remote — *Front End Engineer (Core Team)*

1 YEAR

- Implemented service worker for data caching and web push notifications for increasing user engagement by notifying users without keeping website open.
- Designed and implemented complete messaging system of chat and notifications using Laravel and Web Push notifications for apps, Bilingua and Phoneswap.
- Joined as an intern and got hired as core member within a week for my dedication in assigned tasks.
- Created input validation library for automatic validation of forms using Regex. It ensured data integrity throughout app and was plugged in 2 projects.

Freelance, Remote — *Data Scientist*

6 MONTHS

- Implemented a Recommendation Engine based on reinforcement learning in Python. It allowed the emails to learn from the user's choices, thus increasing CTR.
- Implemented Python script for telecom industry client to perform churn analysis on user data.
- Implemented estimator for student performance using Bayesian Knowledge Tracing and performed detection of similar courses which have similar trends in GPA by doing Clustering analysis on data.

PROJECTS

Chingu Sort — *ML tool to sort applicants according to skills*

- Implemented Python script using Scikit Learn to group people based on the survey inputs.
- This process previously took a week to do by hands i.e. matching skills of various members and then adding them to cohorts.
- This tool solved the problem by using Clustering analysis to find similar people so that they can be organised easily.
- It sorted 500+ members within seconds when cohorts were rebooted. Accelerating the applicant joining process by 100 folds.

Skills

Algorithm and System designing

Data analysis and feature extraction

Machine Learning algorithms

MongoDB, React, NodeJS, Vue

Progressive Web Apps and Experimental Web APIs

Achievements

Led the data science cohort at Chingu Cohorts consisting of 34 members. I helped them grow in data science skills by providing a learning path, assistance and managing the team for image classification project for detecting the darth lord present in star wars images.

Ranked 20 / 4656 in Hacker Earth's Machine Learning Challenge 4 by performing feature engineering on TCP trace data and predicting the network attack associated with it. Used LightGBM model stacked with Naive Bayes.

Ranked 217 / 1386 in Personalized Medicine: Redefining Cancer Treatment Contest on Kaggle by implementing an algorithm that used Gradient Boosting and feature extraction to identify the class of Cancer using medical report and Gene variation as input.

Chingu Lasso — *ML tool for detection of unreliable team members*

- Implemented Python script with completely automated design with ability to scrape Slack API to retrieve messages, read backup message histories.
- It uses the scraped data to analyse the members of the team on basis of Recency, Frequency and Sentiment.
- A detailed Excel report is output that allows team managers to take action and use Lasso to re organise team easily.
- Implemented with Scikit Learn for Clustering algorithms, NLTK for sentiment analysis.

Evolution Strategy AI — *AI that used evolution to play space invaders*

- Implemented in C++ using MLpack library, this gamebot took just the game screen as input (frame by frame) and was able to make decision of moves.
- It was based on OpenAI's Evolutionary Strategies research paper from March 2017.
- Trained using 5 parallel threads for 8 Generations, it got very optimal and was able to gain 500+ score within the 30 second lifecycle.
- Implemented using OpenMP for parallel processing, C++ for learning algorithm and Elixir for bridge between C++ and OpenAI gym.

Slack N Study — *Slack bot for managing private groups*

- Created slash command bot using Node.js and Slack API for searching and management of private channels in Slack.
- This makes finding people with same learning goals, a lot easier and very intuitive.
- It solves the problem of people not being able to find existence of private study groups in Chingu slack team.
- It has been remarked as an amazing and convenient addition by the users as it helps the new members to get started with the study groups within a few minutes.
- Implemented using MongoDB and NodeJS.

EDUCATION

Chandigarh Engineering College — *B. Tech* (2015 - 2019)

Major: *Computer Science*

- Learned about various computer science concepts that include Design and analysis of Algorithms, Computer networks, Data structures and System programming.
- Implemented various algorithms practically, to understand the working more deeply for them.

Ranked 196 / 1245 in Spooky Author Identification on Kaggle by applying NLP on text samples of novels by 3 authors and used stack of Naive Bayes and XGBoost for finding which sample belonged to which author.

Ranked 708 / 99473 in Round 1 and 207 in Semi-finals of TCS Codevita conducted in 3rd year at college campus. The competition constituted of problem solving using algorithms and data structures.

Courses

Machine Learning A-Z - Udemy
Hands on experience of working with R and Python for Machine Learning tasks.

Computational Thinking and Data Science - MITx
Implemented algorithms in Python to solve problem sets based on data interpretation

Artificial Intelligence for Robotics - Udacity
Implemented various localization filters and path finding algorithms in Python from scratch, that are used in self driving cars to understand its location and figure out the route.