Meghana Denduluri

Redmond, WA, USA

Email:dmeghana18@gmail.com
LinkedIn:linkedin.com/in/denduluri-meghana
Phone Number:(832) 6 0 5 -7829

Skills

Programming Languages: Python, R, SQL, HTML, CSS, C/C++

Frameworks & Tools: Pytorch, AWS, Keras, Transformers, ScikitLearn, XGBoost, Pandas, Numpy, NLTK, Matplotlib, Plotly, Seaborn, Jupyter, Git, Tableau

Professional Experience

Amazon Web Services, *Machine Learning Engineer* 07/2022 - Present

- Designed and implemented a scalable machine learning pipeline using AWS services such as **Lambdas, State machines, Step Functions, SQS and S3 buckets** for a global client in the aerospace and defense industry, allowing them to perform fuel consumption analysis on their flight data and integrate it into an application that provides live recommendations on completed flights.
- Performed **exploratory data analysis** on banks, insurance and credit union data for a large investment bank based out of Chicago, to understand and clean the data and provide recommendations/assumptions for model development.
- Trained and fine-tuned an **XGBoost and Amazon Forecast model** in python to forecast average advance balances. **Improved XGBoost model RMSE by 5%** using **blocking time series data split cross validation strategy**.
- Developed and implemented an **MLOps pipeline using SageMaker pipelines** to bootstrap ML project and establish a CI/CD pattern for model building and training.
- Leveraged **AWS Cloud Development Kit** to create a continuous integration / continuous deployment (CI/CD) pipeline for the supply chain prototype, enabling iterative model deployments and updates in a streamlined fashion.
- Built an Amazon Neptune graph database instance infrastructure as code employing **AWS CloudFormation templates** with comprehensive VPC & security protocols for a **lambda-based** centrality calculation service **utilizing gremlin queries**.

Amazon Kids, *Software Developer Engineer* 07/2021 - 07/2022

- Built a data ingestion tool **to automate the process of validating and transforming ingested client data**, **reducing lag time** for downstream jobs **by 50%**.
- Constructed **infrastructural as code pipelines** using Kinesis & Firehose streams to enable rapid yet reliable infrastructure creation, resulting in 70% reduced maintenance costs.
- Collaborated closely with senior software engineers on code reviews, integrating feedback resulting in scalable solutions.

Education

University of Pennsylvania, *M.S Computer Science, GPA: 3.6/4.0* 08/2019 - 05/2021

- Machine Learning, Data Mining, Statistics for Data Science, Big Data, Database and Information Systems, Computational Linguistics, Machine Perception, Data Structures and Algorithms

Gandhi Institute of Technology and Management, *B.E EEE, GPA: 9.3/10.0* 07/2015 - 05/2019

- Operating Systems, Software Engineering, Computer System Architectures, Digital Signal Processing

Projects

Vector Space Models to learn document and word similarity

- Implemented term-document and term-context matrix in **python** to compute similarity of Shakespeare's plays using weighting methods as well as similarity metrics such as Cosine, Jaccard and Dice similarity.
- Since term-context matrix contains the raw frequency of the co-occurrence of two words in each cell, raw frequency is not the best way of measuring the association between words. Word similarity was achieved by computing **positive point wise mutual information (PPMI) and term frequency inverse document frequency (TF – IDF).**
- Performed thorough analysis of the performance of each similarity metric to compare against respective weighting methods.

Recommendation System

- Designed and implemented a recommendation system as a multi-armed bandit problem to recommend movies to users in an online fashion.
- Developed, implemented, analyzed, and compared performances for **Epsilon greedy, Thompson sampling and upper confidence bound algorithms**.

LSH-based Text Retrieval

- Developed a large-scale fast text retrieval system in Python using **fuzzy string matching based on locality-sensitive hashing**.
- Implemented **k-shingles algorithm and Jaccard distance to compute similarity**. Applied the system to the **Amazon Review data set** containing **2M** review texts.

Text Classification

- Developed **7 text classifiers** in python using **scikit learn** to distinguish between simple and complex words that were evaluated and compared using precision, recall and f-score.
- Plotted Precision-Recall curve for various settings of the classifier to gauge its performance and compare it to other classifiers.
- Achieved an **f-score of 0.759** on Gradient Boosting classifier using features like word length, word frequency, number of syllables, ratio of vowel and word frequency from SubtlexUS database.

Clustering on IMDB movie data set

- Applied distance-based clustering analysis to cluster movies based on their attributes on **IMDB data set containing 1.5M movies** in **Python** using **NumPy and Scikit-Learn.**
- Centroid adjustment for **online mini batch k-means using Llyod's heuristic approach**.
- Significantly improved initialization of centroids by K means ++ approach for online mini batch environment.

Certifications

AWS Certified Solutions Architect – Associate [Link]