SALIL NADKARNI

Ann Arbor, MI | +1 (248) 949 9898 | snad@umich.edu | github.com/salnad | linkedin.com/in/salnad

EDUCATION

University of Michigan   **Ann Arbor, MI**
Bachelors of Science Engineering in Computer Science with a Math Minor   **Expected Graduation May 2023**
- *GPA*—3.91 / 4.00, *Honors*—Briggs Scholar ($12,000 scholarship), Dare2Dream Mayleben Venture Fund Recipient
- *Relevant Coursework*—Operating Systems (EECS 482), Bayesian Data Analytics (STATS 451), Foundations of CS (EECS 376), Data Structures and Algorithms (EECS 281), Theoretical Probability (MATH 425)

EXPERIENCE

Google   **Mountain View, CA (Remote) | New York City, NY (Remote)**
Software Engineering (SWE) Intern   **May 2021 - Aug 2021**
- Developed Ads Simulation tool for 10+ member Geotargeting team to reduce testing of ad models from >1 week to < 7 hours
- Created a performant, multithreaded C++ binary with capabilities of processing 100,000+ logged requests that transforms upstream data, calculates key metrics, randomly samples candidates, and highlights differences between two ad models
- Produced detailed design document, gave multiple presentations and published documentation to team website

Student Training in Engineering Program (STEP) Intern   **May 2020 - Aug 2020**
- Utilized WebSockets and YouTube Data API to create a chat centric I/O game around guessing song titles from their audio
- Implemented a synchronous chat between clients using Pusher API, utilized regex and string comparison algorithms for complex answer checking and designed the datastore data model for games used throughout the codebase. code on github
- Added quality of life features including automatic testing, formatting, and a library of helper functions for team of three

ProQuest   **Ann Arbor, MI**
MDP Student Research Engineer   **Jan 2021 - May 2021**
- Implemented deep-learning OCR correction system in python that edits input XML files from ProQuest's database of New York Times 10 million articles from last ~150 years; current model reduces word error rate in validation set by 21.4%
- Applied transfer learning through hugging face library on BART sequence-2-sequence model with ~50k training samples
- Presented weekly to ProQuest stakeholders, generated in depth executive summaries and related documentation

Michigan Data Science Team (MDST), University of Michigan   **Ann Arbor, MI**
Project Lead, Code Autocomplete Project   **Sept 2020 - Dec 2020**
- Directed a project team of 5+ members to using NLP models (eg. Markov, Transformer) to create predictive "code autocomplete" for Python; trained on 30k+ training samples with pytorch and achieved 76% accuracy on validation set
- Produced and presented educational materials on NLP and Machine Learning concepts; hosted weekly office hours for team

Data Analyst, r/rateme Project   **Sept 2019 - Dec 2019**
- Investigated whether post title affected scores distribution for r/rateme, where users submit pictures for others to 'score'; extracted age, gender, and score from posts and comments through developed regex expressions
- Conducted exploratory data analysis and synthesized visualizations used in final writeup, comparing several variables (age, gender, upvotes) using python, pandas and matplotlib. writeup at @ mdst.club/projects and code on github

PROJECTS AND SKILLS

WikiRacer, code @ https://github.com/salnad/wikirunner   *using HTML5,CSS, Bootstrap, SNAP, Flask*
Created a chrome extension that automatically plays the "Wikipedia Game" (find the shortest path between two wikipedia articles by clicking links). Utilizes an API created with data from WikiDump, Stanfords SNAP library, and a custom implementation of A*. Extension placed first for 30+ games against 300+ users after running one day.

Financial Subreddit Analysis, code @ github.com/aveekd/financeSubredditAnalysis   *using Python, pandas, matplotlib, pushshift API*
Wrote and generated an analysis of trust in the market within multiple financial subreddits (eg. r/investing, r/wallstreetbets). Used sentiment analysis libraries and custom implementation of TF-IDF to calculate sentiment on large amounts of reddit data. Found strong negative correlations at the beginning of COVID.

Languages/Skills: Python, C++, Javascript, MATLAB, AGILE Development
Backend Frameworks: Flask, Node.js, Java Servlets
Machine Learning Frameworks: Pytorch, Hugging Face, Natural Language Toolkit (NLTK)
Data Libraries: Numpy, Pandas, Matplotlib, Seaborn, BeautifulSoup, Selenium