prabhunathan.g@gmail.com
+1 (980) 875 - 1874

Prabhunathan Gnanasekaran

https://github.com/nathanprabhu

Microsoft CERTIFIED Professional

EMPLOYMENT

Programmer Analyst — **AlertOps, Bloomingdale, IL** — **Summer 2016 – Present**

- Built a complete and responsive ionic mobile application using ***AngularJS*** that interacts with custom APIs built to configure call-backs, automatically open and close alerts, priority rule mapping etc
- Implemented AlertOps integration with several 3rd party tools for monitoring, chat and help desk systems
- Built open API using ***C++*** for custom integration, built and updated RESTful API that service the front end
- Optimized and increased performance in existing modules by implementing synchronous constructs
- Built AlertOps graphics library using ***C++ OpenGL*** which was used in multiple applications

Research Assistant — **Department of Psychology, UIC** — **Fall 2015 – Apr 2016**

- Built distributed software application using ***C++ and Python*** to simulate Re-categorization experiment (Cognitive science research)
- Statistically analyzed responses using ***Machine Learning Algorithms [includes C4.5, K-Means, Apriori, XGBoost]***
- Built dashboard to present research findings and anomalies
- Designed and implemented dynamic and on-the-fly image generation algorithm that generates combinations of images using minimal image set (Reduced the space complexity by 95%)
- Integrated with '***R'*** language for analyzing the response and generated graphs for the results using JS

Software Engineer — **Infosys Limited, Chennai** — **Fall 2013 – July 2015**

- Worked with multiple stakeholders, interpreted business requirement and led a team in TDD Environment
- Implemented legacy healthcare software solutions in an enterprise environment using ***C++***
- Built automation framework to deploy, test and monitor applications - reviews automation runs, reports/logs and managed source code
- Designed and implemented an algorithm to track company's internal tools usage
- Published four internal knowledge documents on developing data analytics tools through Java and VBA

EDUCATION

M.S (Computer Science) — **University of Illinois, Chicago** — **Fall 2015 – May 2017**

Coursework: Algorithms, Human Computer Interaction, Semantic Web, Data Science, Software Engineering, Mobile Apps Development, Distributed Network Security, Social Network Analytics, Machine Learning

B.E (Computer Science) — **Anna University, Chennai** — **Fall 2009 – May 2013**

Coursework: Operating systems, Databases, Computer Architecture, Algorithms and Data Structures, Compiler Design, Mathematics and Electronics

SKILLS & INTERESTS

- Data Science and Analytics
  - Data Mining, Anomaly Detection, Predictive Analytics, Neural Networks, Data modeling, Big Data and Data Visualization
  - SQL, Python, R, Dashboard generation – JS, Spotfire
  - Pandas, Scikit-learn, NLTK, PySpark, HDFS, MapReduce, Sqoop
- Software Engineering, Full Stack Development, Web Services and API Development
  - C++, OpenGL, C#(MCP), Java, Mobile Application Development (Android, Ionic)
  - HTML, CSS, PHP, JavaScript (Framework - AngularJS), MySQL, JSON/XML, REST API

TECHNICAL EXPERIENCE (PROJECTS)

1. Experimental Platform for Cognitive Learning

(as a Research Assistant/Application Developer in UIC) This application helps cognitive science researchers to analyze the results ***"Deep Learning"*** experiment implemented using ***Convolutional neural networks (tensorflow and hyperparameter tuning*** which makes the application distributed using ***Spark).*** The model's input add operations that read and preprocesses images for evaluation and training. The model then performs inference (classification on supplied images) and computes loss, gradients, variable updates and visualization summaries. Each layer in model computes the logits of the prediction such as convolution and rectified linear activation, max pooling, local response normalization. The output reports the total loss every 10 steps and the batch speed.

The test administrator could define the type and nature of the experiment by controlling various parameters (default/custom). The subject's response in the experiment is the data, which was munged and then analyzed. The results of the experiment are then visualized in various formats with UI following the Researcher's requirement.

Technologies used: C++, OpenGL, Python, SQL, JavaScript, Machine Learning
Demo Links: https://www.youtube.com/watch?v=K7MVdmndEvs https://www.youtube.com/watch?v=741iafNrSWQ

2. Sentimental Analysis on 2016 Presidential Candidate

This application aims to predict the overall opinion of news agencies on President Donald Trump. The dataset contained 50000 tweets about President in the last 2 years and analyze whether the news agencies have positive or negative opinion on him. The tweets were obtained using ***Twitter Streaming API*** and stored in MongoDB. The data is cleaned using ***NLTK*** library and the prediction model is trained using ***k-nearest neighbors*** algorithm

Technologies/Concepts used: Python, PySpark, R, SQL, Link Prediction, Community detection, sklearn

3. AlpenStock - Preventive Healthcare Application

This application imports raw datasets from various sources like CDC, AHRQ, CMS, Data.gov, Google Maps API, City of Chicago Data. These data were then cleaned, munged and converted to standard JSON formats. The structured data was then used to predict and analyze the likelihood of a disease outbreak in a neighborhood city. The application implements ***C4.5 Decision Tree*** algorithm in ***Python*** to classify the likelihood of outbreak and visualizes results using ***JavaScript D3*** library.

Technologies Used: C++, R, JavaScript, Machine Learning, JSON, numpy, D3
Project Link - https://devpost.com/software/alpenstock

4. MySentry - Secure Neighborhood and Travel Planning

This application detects any violence using computer vision. Data (raw frames) obtained from cctv cameras installed. The violence is reported to law enforcement officers with the description of victim and offender (using computer vision), the violence is marked on the Map with description generated automatically. The dashboard imports raw datasets from various sources like Data.gov, Google Maps API, City of Chicago Data, US Crime data, World bank. These data were then cleaned, munged and converted to standard JSON formats. The structured data was then used to analyze whether the neighborhood is safe on a time (prediction is done using previous crime data), also the application suggests the user an alternate route avoiding recent violence that took place with 3-mile radius. The model was trained using ***Clarifai API*** and ***Deep learning algorithms (convolution and pooling)*** for image classification implemented using MLlib and Mahout libraries in Apache Spark.

Technologies Used: Python, R, JavaScript, Machine Learning, JSON, Clarifai API, MLlib, Mahout
Project Link - https://devpost.com/software/safetynet-dab9l2

5. Comparison of ML algorithms (Naïve Bayes, Logistic Regression and Decision Tree) using Sentiment Analysis

A Machine Learning application was designed using ***Apache Spark (MLlib library)*** that learns from 80000 tweets and predicts the sentiment of any input tweet. Real time tweets fetched through Twitter's ***Streaming API*** were imported to ***MongoDB*** using Python scripts. The data is then cleaned using ***NLTK*** library and Regular expressions. The application implements three most commonly used Machine Learning algorithms ***(Naïve Bayes, Logistic Regression and Decision Tree)*** on the cleaned data, performs sentimental analysis and compares speed and accuracy. The validation is done using ***k-fold*** cross validation estimator which has a lower variance than a single hold-out estimator and the amount of data available is limited.

Technologies/Concepts used: Python, PySpark, Machine Learning, Apache Spark, MLlib, Streaming API, NoSQL, NLTK Library

6. Anomaly Detection in Predictive Maintenance

This application continuously monitors the condition of the machine by looking at the pre-defined parameters of each IoT equipment and tracks the pattern that might indicate equipment failure. The Time to Failure was estimated by building a ***Regression*** model, ***Binary classifier*** was built to predict if an asset will fail with a time frame, ***Multi-Class classifiers*** were built to predict if an asset will fail in different time windows. The data which comes through IoT devices (1200 records per second) were cleaned using ***Pandas*** and transferred to a Relational Database using ***Sqoop*** from ***Hadoop Distributed File System*** *and* ***MapReduce***. The models were implemented in ***R*** using ***tree*** package (classification and regression tree). ***R*** is integrated to ***Spotfire*** for live visualization and monitoring the IoT device.

Technologies/Concepts used: C++, Machine Learning, MS SQL

ADDITIONAL EXPERIENCE & AWARDS

- Institute of Electrical and Electronics Engineers - Webmaster, Madras Section (2012 – 2013) – Built and maintained a web application for student networking and to provide visitors quick access to all major IEEE activities by organizing into information portals
- HackIllinois 2016 Winner (36 hour coding competition at University of Illinois - Urbana Champaign)
- Best Project and Team – 2015 (By Infosys Limited)
- Outstanding Student Member – 2012 (By IEEE)