Prakhar Dogra

Fairfax, VA | (571)888-4280 | https://github.com/prakhardogra921 | pdogra@gmu.edu

EDUCATION

Master of Science in Computer Science,   May 2018
George Mason University, Fairfax, VA   **GPA:** 3.8
(**Relevant Coursework**: Machine Learning, Computer Vision, Pattern Recognition,
Mining Massive Datasets using MapReduce, Data Mining)

Nanodegree: Deep Learning (Udacity)   June 2018
Nanodegree: Machine Learning Engineer (Udacity)   Feb 2018

Bachelor of Technology in Computer Engineering,   July 2016
Delhi Technological University, Delhi, India   **GPA:** 3.5

TECHNICAL SKILLS

Proficient: Python, Hadoop, Spark, Tensorflow, OpenCV
Familiar: Java, C/C++, HTML, CSS, JavaScript, R, SQL, Django, Amazon EC2, Google Cloud

RELEVANT EXPERIENCE

System Design Intern, CafeX Communications, New York   Jun - Aug 2017
- Developed a document parsing system to extract relevant information from meeting minutes.
- Deployed the system online using Django web framework.

Data Science Intern, Premier Logic, Noida, India   Feb - May 2016
- Scraped news websites in order to find recent news of a particular domain.
- Developed an API for Virtuagym to find clients interested in fitness and gym using Twitter.

PROJECTS

Vehicle Detection using Faster R-CNN   Feb – May 2018
- Implemented state-of-the-art *Faster R-CNN* architecture using *Keras* for detecting vehicles in Traffic Camera images.
- Used pretrained weights from *VGG16* and *ResNet50* to reduce training time.

Bank Marketing Classification   Dec 2017 – Feb 2018
- Used scikit-learn library to train and validate classification models like *Logistic Regression*, *Decision Tree*, *Random Forest, Naïve Bayes, Neural Network* and *Support Vector Machine*
- Conducted comparative study on the above models when applied on different feature sets obtained via feature selection (*Chi-Square Test*), feature transformation (*Principal Component Analysis*) and feature elimination.

Year Prediction using Regression   Sep – Dec 2017
- Predicting release year of a song from its timbre features using regression.
- Implemented *Linear*, *Lasso*, *Ridge*, *Elastic Net*, *Polynomial*, *Neural Network* and *Step-wise Regression* models from scratch using *numpy* library. Also implemented *batch* and *stochastic gradient descent* optimizers.

Meeting Minutes Parser   June – Aug 2017
- Developed a document parsing system and deployed it online, using *Django web framework*, that can extract required information from meeting minutes with a feedback system.
- Performed text analysis on meeting minutes to extract meaningful actions discussed in meetings

Finding similar images using Locality Sensitive Hashing   Jan – April 2017
- Implemented LSH in *Apache Spark* from scratch for image data points.
- Deployed the python script on multiple Amazon EC2 instances and configuring the master and slave instances.
- Implementation procedure includes techniques like shingling, min-hashing, random hyperplanes and sensitive hashing.

Credit Card Fraud Detection using Anomaly Detection   Sep – Dec 2016
- Implemented various anomaly detection techniques like *Log Likelihood, K Nearest Neighbour, Mahanalobis Distance, StrOUD (Strangeness-based Outlier Detection algorithm)* and *Local Outlier Factor*
- Compared the implementations with other scikit-learn models like *Random Forest*, *One Class SVM* and *Isolation Forest*
- Models were compared on *F1 score*, *Area under ROC* and *Recall*

Twitter Analysis to Find New Gym Clients   Mar – May 2016
- Used Twitter API to fetch local tweets (specific to a geographic location) with specific hashtags.
- Used *scikit-learn* library to train and validate classification models like SVM, Naïve Bayes Classifier and its variations.
- After classification, usernames and user mentions in the tweet for the classified users were extracted.

Social Media Complaint Workflow Automation Tool Using Sentiment Intelligence   Sep – Dec 2015
- Developed a complaint classification and forwarding mechanism for bank posts obtained from Facebook web pages.
- Used bag of words and NLTK for extracting features from text obtained from websites through crawler.