

BHAVIN DAYALAL DHEDHI

+1 (917) 963 6146 | bhavin.dhedhi@columbia.edu | <https://github.com/thisisbhavin> | www.linkedin.com/in/bhavindhedhi

EDUCATION

Columbia University

New York, NY

MS in Electrical Engineering, Specialization in Machine Learning | GPA: 3.84/4.00

Expected Dec 2022

Courses: Computer Vision, Neural Networks and Deep Learning, Algorithms for Data Science, Big Data Analytics, Machine Learning

K. J. Somaiya College of Engineering

Mumbai, India

B. Tech. in Electronics and Telecommunication Engineering | GPA: 8.64/10.00

May 2018

Courses: Neural Networks and Fuzzy Logic, Image Processing, Digital Signal Processing, Cloud Computing, Operating Systems, DBMS

SKILLS

- Languages: Python, SQL, R, HTML, CSS, JavaScript, C/C++
- Frameworks & Libraries: TensorFlow, Keras, OpenCV, Pandas, NumPy, Scikit-Learn, Flask, R-Shiny, spaCy
- Platform, Tools and Databases: AWS, GCP, Docker, PySpark, Airflow, Neo4j, Git, Jira, Tableau, MySQL, PostgreSQL

PROFESSIONAL EXPERIENCE

Think360.ai

Mumbai, India

Software Engineer - Machine Learning

Aug 2018 - Jun 2021

- **Face liveness detection:** (1) Replaced task-based liveness detection with DNN based detection. (2) Trained **DenseNet** – initialized with ImageNet weights for the first 8 layers – on a curated dataset (3) Implemented **custom loss** function that calculates Pixel-wise binary cross-entropy at second last layer and binary cross-entropy on output label. (4) Improved the TAT from 30 seconds to 2 seconds on CPU and 0.1 seconds on HWA. ([kwik.id](#))
- Developed DNN to segment government ID cards from the images uploaded by users. Fine-tuned **U-Net** on 100,000 ID cards images and achieved an IoU of 0.97. The segmentation helped reduce the time taken during information retrieval.
- **Information Retrieval:** Revamped API suite in **Flask** for information retrieval from government-issued ID cards leveraging Google Vision API, improved recall rate from 78% to 98.3%. Reduced TAT from 6 seconds to 1.3 seconds. ([kwik.id](#))
- Implemented high-performance web scrapers in Python to scrape 10M reviews and business listings from Yelp and deployed the same to AWS Fargate. Trained a NER Model to extract menu items from the reviews and suggested top dishes for a given location.
- Constructed a Big Data pipeline utilizing PySpark on AWS EMR to extract research papers' metadata (200 GB) from Semantic Scholar and store it in PostgreSQL database.
- **No ball detection:** (1) Led a team of 3 in designing and developing a Deep Learning based POC system to detect No balls in Cricket, the solution was also presented to the **chairman of BCCI** (2) Trained **YOLOv3** on a curated dataset of 5000 images. (3) Calculated landing point of the foot using bounding box regression. The model could detect 80% of the No-balls missed by umpires. ([News Coverage](#))
- Mentored two interns during the entire lifecycle of feature development. I helped them manage bi-weekly sprints and assisted them technically. I also provided technical feedback in code reviews before pushing the code to production.
- Designed a framework by utilizing **PCA** and Hotelling-T² statistics to spot anomalies in time-series data. Improved the recall rate from 70% to 92%, allowing the company to save ~\$1000 for each correctly spotted anomaly.
- Developed credit score model using **Random Forest** with 85% accuracy and 10% improvement over the previous model.

PUBLICATIONS

- "Automatic license plate recognition using Deep Learning": ICIIT ([Link](#)) Dec 2018
- "Detection of birds in the wild using Deep Learning methods": IEEE ([Link](#)) Oct 2018

PROJECTS

Advanced lane finding for self-driving cars ([Link](#))

Jan 2019

- Removed camera distortions using Calibration matrix. Transformed road image to "Bird's Eye" view to locate curved lanes.
- Detected lane line pixels by employing sliding window technique and polynomial regression.
- Calculated curvature of the lane lines by transforming polynomial curvature to real-world using Highway Design Handbook.

Behavioral Cloning to Steer A Car ([Link](#))

Feb 2019

- Implemented a modified version of the architecture presented in Nvidia's self-driving car [paper](#) in TensorFlow.
- Data was augmented on the fly while training the model. It included illumination changes, noise incorporation and horizontal flips.
- Trained the model for 20 epochs and got 99.8% accuracy.

Web App For Exploratory Data Analysis - Think360.ai

May 2020 - Jul 2020

- Developed a web app that plots frequency charts and shows the descriptive statistics for each column in the user-uploaded data.
- Added capabilities for bivariate analysis by allowing users to create a Box plot, Line plot, Scatter plot, Heatmap, Bar chart.
- Reduced the preliminary data analysis time on an average from 2 hours to 15 minutes

AWARDS

- Awarded the most innovative solution at AIDL Hackathon 2020 for creating a framework for information extraction from invoices by leveraging Graph Theory and Random Forest. ([Link](#))
- Secured 1st Prize in Project Competition, Prakalpa 2017 - a statewide competition - for building a handwritten digit recognition system based on convolutional neural networks.