

Mahesh Kumar

(202) 643-7703
maheshkumar51342@gmail.com
<https://github.com/gk19989>
<https://www.linkedin.com/in/maheshgudumala>
Washington DC



Education

The George Washington University (MS in Computer Science)	08/2021-05/2023
Design & Analysis of Algorithm, Machine Learning, Natural Language Processing, Software Engineering	
SpringBoard (Artificial Intelligence and Machine Learning Career Track)	03/2020-06/2021
Computer Vision, Natural Language Processing, Machine Learning, Deep Learning, Certified in AWS Machine Learning,	
Anurag Group of Institutions (BTech in Computer Science and Engineering) GPA: 8.8/10	07/2015-06/2019
Data Structures, Web Technologies, Database Management Systems, Cloud Computing, Machine Learning	

Technical Skills

Languages:	Python, Java, C#, JavaScript, SQL, HTML5, CSS3, XML
Tools:	Eclipse, Visual Studio Code, PyCharm, Postman, Docker
Technologies:	TensorFlow, Keras, ASP.NET MVC, Entity Framework, REST, JSON, AWS, Azure, Git
Database:	MongoDB, MS SQL server, MySQL, Redis, Elastic Search
Certifications:	AWS Machine Learning Specialty, AWS Solutions Architect- Associate

Professional Experience

OpenText (Hyderabad, India) Software Engineer	08/2019-03/2020
<ul style="list-style-type: none">Maintained quality of OpenText's Digital Asset Management (DAM) software called OpenText Media Management (OTMM).Working in agile environment on a day-to-day basis performed Regression Testing on various components of OTMM.Automated entire Application using Selenium, Java and TestNG on Page Object Model and published daily test reports.Performed Backend testing by validating source and target tables by developing SQL queries and verifying stored procedures.Ensured Unit, Integration test, Continuous Integration and Deployment using Tortoise SVN.Created the Test Plans, Test cases for Functional testing in coordination with Business Analyst and Developers.	

Projects

Question & Answer System – Natural Language Processing

- Developed a Question & Answering System using Stanford Question Answering Dataset & Bidirectional Encoder Representations from Transformers (BERT) model in TensorFlow and Keras.
- The model is trained to predict starting and ending of the answer to the question asked, represented in the form of vectors. Used sparse categorical cross entropy for loss function and Google's official optimizer to optimize the model.

E-Commerce Application – Software Development

- Developed an interactive E-commerce application named Sprint using MERN stack.
- Implemented front-end using React and Redux and built Web-API using Node and MongoDB.
- The application contains product-listing, product details, categories, product search screen, add-to-cart feature, check-out wizard, order summary screen, user's profile, order history and payment method.

Customer Segmentation using Clustering – Machine Learning

- Dataset contains information on Marketing Newsletters and Transaction-Level Data from customers, applied K-Means Clustering, chose clusters by the Elbow Sum-of-squares method and Silhouette method.
- Used PCA to find optimal number of dimensions and reduced the dimensions to 2-components and visualized 2d-plane clusters.
- Repeated the same process for other clustering algorithms viz, DBSCAN, Affinity Propagation, Spectral Clustering and Agglomerative Clustering. Compared the clusters formed by different algorithms.

Video Rental Application – Software Development

- Developed an ASP.NET MVC Application used to manage video rental store. The application has two user roles: admin (To manage inventory) and users (To create account, login and rent movies based on the availability).
- Implemented Entity framework using code-first workflow. Used SQL Server for Database Management.
- Built RESTful service with ASP.NET Web API. Performed Authentication and Authorization using ASP.NET Identity and OAuth.

Face Recognition - Computer Vision

- Project is based on FaceNet (Neural network that encodes a face image into a vector of 128 numbers). Compared two vectors to determine if two pictures match. Used pre-trained inception model based on the architecture specified in "Szegedy et al".
- Implemented Triplet Loss on the inception model. The loss is based on the distance between the images where the distance is computed using 128 number encoding.
- Using Keras, optimized the above model on Adam Optimizer and implemented Face Verification and Face Recognition with a small database of images.