Portfolio

Srikar Sarma PV
656 E Duane Ave, Sunnyvale, CA.
Ph.no : 2142281231
Email : srikarsarma@gmail.com

About me.

My name is Srikar, and I currently work as a Machine Learning Engineer at Iron Mountain. My passion lies in designing, implementing, and deploying Deep Learning models for various tasks like Intelligent Document Processing, integrating Natural Language Processing, Computer Vision, and multimodal LLMs like LiLT and GenAI. I have hands-on experience with technologies such as Kubeflow, MLflow, ONNX, Triton, RAG, Open source LLM, Langchain, and PyTorch.

Experience :

Software Engineer- Machine Learning, Iron Mountain

In my role at Iron Mountain, I am responsible for the complete machine learning lifecycle, from conducting research and development to implementing production-ready solutions. I have built reusable AutoML pipelines and worked extensively with Kubeflow for orchestration, MLflow for tracking experiments, and ONNX, TF and Pytorch for model interchangeability. Additionally, I have integrated Triton for high-performance inference serving, RAG for retrieval-augmented generation, and leveraged open-source LLMs and Langchain for natural language processing tasks.
Previously, I was part of a team at Iron Mountain that developed the Lending AI vertical, delivering scalable machine learning solutions for package splitting, document identification, and entity validation. Our AI system significantly reduced loan processing costs, from $3 per package to just 1 cent, becoming a significant income stream for Iron Mountain and serving three banking clients. This human-in-the-loop AI system brought

efficiency and cost-effectiveness to loan processing, optimizing operations and improving customer experience.
I am currently working on deploying Gen AI and OpenSource LLM (Mistral) to build automated pipelines for various tasks.

Tools Used: TensorFlow, Pytorch, Kubeflow , MLflow, Triton, RAG, Open source LLM, langchain, Hugging Face, Docker, Grafana
Languages : Python, Bash

Software Development Engineer-Data Lab, PricewaterhouseCoopers(PwC)

I have extensive experience in designing and implementing data-driven solutions to address complex challenges. In my previous role, I collaborated with the Digital lab team to develop an Anomaly Detection tool aimed at identifying fraudulent transactions within SAP financial data. This involved utilizing DBScan and K-means clustering techniques to detect anomalies effectively. The resulting web application, hosted on Azure, provided real-time insights and actionable intelligence for our Fortune100 client.

Additionally, I contributed to the development of a recommendation system designed to streamline operations for the Operations Support team. This system leveraged NLP algorithms such as Bag of Words, TF-IDF, and doc2Vec to recommend relevant knowledge base articles when handling support tickets, enhancing team productivity and customer satisfaction.

I played a key role in creating a comprehensive Dashboard using the MEA(R)N stack, enabling efficient monitoring of team activities and automating various tasks to enhance operational efficiency.

Tools Used: TensorFlow, Pytorch, NLTK; DBScan, Kmeans, clustering algorithms

Languages : Python, JavaScript, C#, Apple Script

Machine Learning Intern (AI/NLP), Verizon Wireless, New Jersey, USA.

I interned at Verizon Wireless, Piscataway, NJ in the summer of 2018 as a Machine Learning Intern. I was involved in an agile environment to prototype and productionize a patent for

test case auto matching using Machine Learning and Natural Language Processing techniques. I worked on matching developer code changes(commits) from Accurev and Git to 50k+ test case flows. The idea was to speed up regression testing. I used TF-IDF and PCA to analyze the entire code base (5 million+ lines of code) of Java and Javascript code from the POS (Point of Sale) team to get insight on what functionality is being changed eg: By looking into classes and function definitions, mapped these keywords to appropriate test case flows with word embeddings, trained the models using Stochastic Gradient Descent classifiers, word2vec, SVMs. The speedup was 10x times that of regression testing which took about 3 days for a complete cycle of testing. I was also involved in prototyping a chatbot for the POS (Point of Sale) team to help internal manual testing team with frequently raised issues and bugs using the seq2seq tensorflow model.
This experience provided me with hands-on experience in ML, NLP, and software development methodologies within a corporate setting, honing my skills in data analysis, model training, and project management.

Tools Used: TensorFlow, Pandas, Scikit, numpy, Plotly, AWS, boto3, AccuRev, Git, SQL, MongoDB, Jenkins

Languages : Python 3.5

Jr. Data Scientist, MetaIoT Technologies (Vehico) , Bangalore.

MetaIoT is an analytics company focused on connected cars and IoT, their flagship device is called Vehico. I was involved with developing a model to detect driver performance. My first task was to develop a speed bump and pothole detection algorithm using thresholds of the ax,ay,az (acceleration vectors) and gx,gy,gz (g-force vectors) . The speed with which a driver swerved left/right and went over a speed bump/pothole was compared with a said threshold to determine a rashness factor. This factor was combined with a customer rating for that trip and an overall driver performance based on 5 stars was given, which was used by the clients to assign better drivers for trips. Our clients included JetFleet (fleet wing of JetAirways), OLA cabs , etc.

My second task was to predict the distance, time, cost and estimated time of arrival. I used regression and classification algorithms such as linear, Ridge, Lasso regression and SVM for classification on the data to generate predictions. I automated the predictions for any given time frame eg; 3 hours, 5 hours, 1 day etc. and plotted real time predictions on the company's web and mobile app using chart js and nodejs.

My third task was to create summary sheets which was an overall summary of trips, cost, fuel consumption and duration of a trip. I developed this using Tableau Dashboards.

Tools Used: AWS, Pandas, SQL, Scikit, numpy, pandas, tensorflow, Plotly, boto, bitbucket, Tableau.
Languages : Python 3.5, nodejs

Research Assistant, Institute for simulation and training (IST), University of Central Florida :

I worked on user behavior analysis in a social network for a DARPA funded project at IST under the Guidance of Dr.Wingyan Chung. My task was to collect data from the github api in json format, convert this data into a dataframe and store it in csv format. I worked on modeling and simulation of user- repository interactions such as fork, push, pull, commit, watch and delete operations. Used Repast Simphony tool in Java to build an Agent-Based Modelling system for simulation of interactions between agents.

Tools Used : Repast Simphony, Bash shell, SQL
Languages: Java, Python 3.5

Graduate Teaching Assistant, University of Central Florida:

Worked with Professor Mike McAlpine and Assisted in building coursework and grading for COP-4600 Fall’17 (Operating Systems) . I designed programming homeworks for a class of 175 students in Java and C sharp. Assisted students with assignments and understanding Operating System concepts.

Tools Used : Webcourses@UCF grading tool, excel.
Languages : JAVA , C#

Inplant Trainee| Intern, ISRO Telemetry Tracking and Command Network, ISRO, INDIA

Monitored various operations during satellite passes across ground station, Analyzed satellite scheduling data using Microsoft Excel. Used openCV2 and PIL python packages to analyze satellite images for detecting edges and corners.

Tools Used : MS Excel, PIL, openCV2.
Language : C++, Python 2.7

Current Projects and Research.

Publications :

Deep feature based on convolutional auto-encoder for compact semantic hashing. (Submitted for the ACM SIGGRAPH / Eurographics Symposium on Computer Animation (SCA))

For content-based image retrieval, a good presentation is crucial. Nowadays, as deep learning models can be used to generate an excellent presentation, it has been extensively investigated and widely used in research systems and commercial production systems. However, the deep representation (deep feature) is still too large. Compared with directly using deep representation, binary code can reduce significant storage overhead. Meanwhile, bitwise operations for binary code can dramatically fasten the computation. There exist some schemes used to convert the deep feature to binary code, but all of them directly applied the last layer of the connection layers, which exhibit global feature and discriminating features. To achieve deep generative feature and avoid destroying the image locality, we aim to construct the binary hash code based on convolutional auto-encoders.

The paper can be viewed here.

I was involved in designing the convolutional autoencoder train the given datasets (MNIST, CIFAR-10 and NUS-WIDE), implementing a hashing algorithm to hash image features to binary hash codes, unsupervised learning of these codes using encoder-decoder network and again supervised learning of close codes using a restricted boltzmann machine(RBM) from the convolutional layers. For Image retrieval our model shows promising performance boosts over state of the art techniques such as LSH, ITQ, etc. Detailed comparison results are shown in the form of Graphs and tables (MaP scores).

Tools Used : Tensorflow, CAFFE, MATLAB.
Languages : Python 3.5 , C++

Computer Vision Based Approaches for Driver Fatigue Detection - A Review*

This Paper is based on my Final year project for my Bachelors degree titled " Real Time Intelligent system for Driver Fatigue Detection. " The project involved a lot of background research and related work. We skimmed through lot of Journals and research papers before we arrived at an intelligent, smart and fast system for detecting driver fatigue using Computer Vision Techniques. We used Viola Jones Algorithm with adaboost technique for facial detection in realtime separated facial features such as eyes, mouth contour, and cheeks using segmentation available in Matlab's Image processing toolbox. Based on eye closing (duration) and mouth opening (Yawning) we detect driver fatigue. We initially used pretrained models such as FaceNet and Vgg19. Later we developed a technique to build a dataset of a Driver's Images through a Webcam mounted on the car. Trained the model for a specific driver image avoiding the hassle of worrying about color, ethnicity and noise which was the case with pre-trained models. Our project received "Project of the year" award at the annual college symposium.

The paper summarizes the state of the art computer vision techniques being used to detect Driver Fatigue, It provides a detailed review and comparison of different computer vision techniques and also discusses possible improvements and future works.

Tools Used : MATLAB, Image processing toolbox, Matplotlib, Webcam Api, sklearn
Languages : Python 3.5, Java(Gui).

Projects :

Lab Project (Ongoing) :
Predicting Block I/O from system traces using seq2seq tensorflow model for prefetching i/o addresses.

This is the current project I am working on. The idea is to try to learn and predict I/O block addresses from the SPC TRACE FILE dataset which has over 10 million file traces from various systems.
My task is to set up a high performance computing cluster, analyze the huge trace file and handle the big data using hadoop and spark frameworks. I analyzed the data using Pandas and seaborn libraries to get insights about correlation, mean, std deviation and develop dependency plots. I am currently using tensorflow seq2seq Neural Machine Translation model to predict a sequences of the Block I/O address. It is a model that uses Recurrent Neural Networks to translate and predict sequence of codes. The idea is to predict sequences of Block I/O addresses to accelerate processing of Block I/O requests with the prefetched data.

Fake News Detection :

This project was done as a coursework fulfillment for Natural language processing course I took this semester. In this project, I worked on one of the biggest problem, detecting fake news with 52,000 article and 97% Accuracy.

I started with scraping news from NYT API and The Guardian API to have data set labeled as real news, and downloaded fake news dataset from kaggle.com. At the end I scraped more than 200,000 articles. I wanted to be able to represent the real world in terms of the proportion of the real news and the fake news in my training. I had 12,000 fake news articles from kaggle.com so I decided to have more real news, assuming there are more real news than fake in real world. Eventually I had 43,000 real news and 12,000 fake news.

I used Beautifulsoup tool in python for scraping articles through the NYT, Guardian news API's and stored the data in a Mongo client using pymongo, Used NLTK to preprocess and tokenize the text. After text processing I used TF-IDF of bi-grams and tri-grams to know the relative importance of words in both our fake news and real news datasets. For classification I used Support Vector Machines (SVM), Stochastic gradient descent (SGD), Gradient Boosting(XGBoost), Bounded Decision Trees(DT), Logistic Regression and random Forests(RF). The purpose of using so many classifiers was to get detailed comparison results of each classifiers performance.

Tools Used : Tensorflow, NLTK, pymongo, beautifulsoup, scikit.

Language : Python 3.5

Neural style transfer using Convolutional Neural Networks :

This project was done as a part of course fulfillment for the computer vision course at UCF in fall'17. This project is a Keras implementation using Tensorflow as backend that performs an alternative style transfer method. My implementation differs from the original (Gatys et al., 2015), in that the semantics of the content image is not iteratively being transferred onto the output by an optimization function. Instead, local information from the style image is being directly transferred over to the content image. This creates a different output image where the unwanted local features that comes with the content image transformation is being iteratively removed, making style image features more salient. We used a pretrained VGG16 model and extracted features from intermediate layers of the CNN.

Tools Used : Keras, Tensorflow, PIL,OpenCV2
Language: Python 3.5

Chat Bot using Recurrent Neural Networks :

This project was done for Knight Hacks, a Hackathon conducted by UCF. We designed a chat bot using the popular “Cornell Movie Dialogs Corpus”and 300 dimensional GloVe embeddings. The idea was to reproduce the results of this paper. We used Tensorflow’s seq2seq model and keras for the implementation. The sequence to sequence model uses two LSTM networks, one each for encoding and decoding respectively. I used three LSTM layers with 512 as layer sizes respectively. The model performed well for few common sentences but it did not perform very well on uncommon sentences although it tried to generate some answers.

Tools Used : Tensorflow, Keras.
Language : Python 3.6

Sentiment analysis on Movie Reviews :

For this project I used the bag of words (BoW), word2vec and doc2vec model for sentiment analysis on movie reviews (SAMR). This was a Kaggle challenge, The dataset was obtained from kaggle, I removed stop words if unigram is used, and did not remove them if bigram or bigram/unigram is used. Dictionary size for non-word2vec vectorizers was 5k for unigram , 10k for bigram and 10k for bigram/unigram. For word2vec, to obtain the feature vector for each review, first I learn the vector representation of words, and then average all vectors of the words in each review. I used 10-fold cross validation for different machine learning algorithms ie. Random Forests, Gaussian naive bayes, multinomial naive bayes and linear SVM to get insight into what algorithm performs best, Empirical results show that Linear SVM with tf-idf and bigram/unigram vectorizing yields the best result.

Tools Used : NLTK, Tensorflow, pandas,seaborn

Language : Python3.5

Contact Information :

Email : srikar@knights.ucf.edu
srikarsarma@gmail.com

Github : https://github.com/srikarpv
LinkedIn: https://www.linkedin.com/in/srikarpv
Phone no. +1 (214) 228 1231