Projects

As part of my extracurricular activities beyond my university coursework, I participated in hackathons on Kaggle and at my university to enhance my understanding and practical skills. I've worked on various projects in areas including Natural Language Processing, Computer Vision, Video Processing, Time Series Analysis, chatbots, large language models (LLMs), vision-language models (VLMs), Transformers, and other deep learning applications. To deepen my understanding, I also built machine learning (ML) and deep learning (DL) models from scratch, without using tools like Scikit-learn, PyTorch, or TensorFlow. These activities have helped me strengthen both my theoretical knowledge and practical skills. A few of my projects are listed below.

I have experience working with a wide range of tools and technologies across ML, DL, CV, NLP, and DevOps. My toolkit includes libraries and frameworks such as Scikit-learn, TensorFlow, PyTorch, Keras, MLflow, HuggingFace Transformers, NLTK, SpaCy, Gensim, OpenCV and Detectron2.

I have hands-on experience with a wide range of deep learning model architectures across various domains. For image and sequence modeling, I've worked with CNN, RNN, LSTM, GRU, and advanced networks such as ResNet, VGG, AlexNet, MobileNet, EfficientNet, NasNet, ShuffleNet, and U-Net. In the domain of object detection and segmentation, my experience includes models like YOLO (v8+), Mask R-CNN, Faster R-CNN, R-CNN, Fast R-CNN, Swin Transformer, and SAM. For natural language processing and understanding, I've worked with models like BERT, RoBERTa, DistilBERT, LLAMA, MISTRAL, and ViT. Additionally, I have experience with vision-language models (VLMs) and very large language models (VLLMs), including CLIP and multimodal transformers.

In the domain of video and real time processing, I have worked with tools like WebRTC, and RTSP for handling video streams and building live streaming applications.

Additionally, I am proficient in deployment and DevOps tools such as Linux, AWS, Docker, Kubernetes, FastAPI, and Flask, which I have used for model deployment, containerization, and API development in various projects.

For Further details, check out my Website. https://dawoodsarfraz.github.io/

Projects Overview

- **Smart Surveillance System**
  I have developed a real time Traffic Analysis System that processes live RTSP video streams by converting them to WebRTC for efficient web based analytics. I implemented advanced object tracking algorithms like SORT, DeepSORT, and ByteTrack to monitor movement and estimate direction. The system comprises modules for object detection, direction estimation, and vehicle counting, enabling the tracking of congestion. Additionally, I built algorithms for wait time estimation, queue detection, speed calculation, and traffic light violation detection to enhance traffic flow and ensure rule enforcement. The project also features line and zone intrusion detection to improve security and manage restricted areas.
  GitHub: https://github.com/Dawoodsarfraz/Smart-Surveillance-System

- **Deep Learning Approaches for Multi-Class Cancer Classification: A Comparative Study of CNN, ShuffleNet, and NasNet Models (Research Project)**
  Skin cancer, a prevalent form of cancer, impacts millions of individuals annually due to abnormal cell growth. This can lead to a dangerous spread to other parts of the body, which can be life threatening. Early detection and staging are crucial because skin cancer has high mortality rates. We have done extensive studies to identify the best classifier suitable for dermoscopic image classification using different deep neural networks. Various deep neural network architectures, including CNN, NasNet, and ShuffleNet models, were evaluated using the HAM10000 dataset. The dataset was obtained from the Harvard Dataverse and comprises seven distinct types of lesions. To improve model performance, we used RandomOverSampler to address data imbalance. Results showed promising outcomes: CNN achieved 92% accuracy, NasNet improved further to 93%, and ShuffleNet reached 87%. These findings highlight the potential of advanced neural networks to enhance skin cancer diagnosis.
  GitHub: `https://github.com/Dawoodsarfraz/Deep-Learning-Approaches-for-Multi-Class-Cancer-Classification`

- **MetaMed (FYP)**
  The rapid evolution of technology continues to redefine traditional educational paradigms. This report delves into the pioneering exploration of leveraging Immersive Virtual Reality (VR) to transform medical education. Focused on the project ”Enhancing Medical Education through Immersive Virtual Reality (MetaMed)” our study aims to revolutionize the pedagogical landscape of medical training. In the initial phase of our project, we embarked on a journey to harness cutting edge VR technology. The objective was to create immersive learning environments that simulate real world medical scenarios, allowing students to engage in interactive, hands-on learning experiences. Through the integration of advanced VR techniques, our project aspires to bridge the gap between theoretical knowledge and practical application, fostering a deeper understanding of medical concepts. ***GitHub:*** `https://github.com/Dawoodsarfraz/MetaMed`

- **Gastrointestinal Disease Classification using Endoscopic Images**
  I worked on automatic disease detection using endoscopic images from the Kvasir dataset, which contains labeled gastrointestinal (GI) tract images, annotated by experienced medical professionals. For this task, I implemented and experimented with hybrid deep learning architectures, combining the strengths of EfficientNet, RegNet, Inception, ResNet152, and VGG19 with transformer-based models like ViT and Swin Transformer, to capture both local and global features effectively for enhanced performance in medical image classification. These

hybrid models were designed to leverage the spatial precision of CNNs and the contextual reasoning power of transformers, providing a robust solution for automated GI disease detection.
GitHub: `https://github.com/Dawoodsarfraz/Gastrointestinal-Disease-Classification-using-Endoscopic-Images`

- **RAG System with Llama 3.2**
  This project implements a Retrieval-Augmented Generation (RAG) system that integrates document retrieval and text generation to provide accurate, context-aware answers. It uses LangChain to orchestrate the retrieval and generation workflow, Ollama for efficient model deployment, and Fasiss CPU for energy efficient inference. PyPDF extracts relevant text from PDFs for the retrieval phase, while MTEB ensures high-quality embeddings for improved accuracy. The system is optimized for tasks like research, customer support, and knowledge management, offering dynamic and precise responses.
  GitHub: https://github.com/Dawoodsarfraz/RAG-Llama-3.2

- **InsightFlow**
  In this project, I am building InsightFlow, a system for real time news retrieval, research, and content creation. It utilizes four core agents: News Retriever, News Scraper, News Summarizer, and News File Writer to efficiently gather, analyze, summarize, and write articles on any given topic. The project integrates tools like Ollama for context-aware summaries, Llama 3.2 for processing complex news data, CrewAI for task orchestration, and Serper for retrieving up-to-date articles. Together, these components streamline the news gathering and content generation process, ensuring timely and relevant information.
  GitHub: https://github.com/Dawoodsarfraz/InsightFlow

- **LLamaAssist**
  LLamaAssist is an application built with Llama 3.2-3B and using Ollama. It simplifies content creation, offering customizable options for assignments, blogs, and AI-driven interactions. It leverages LangChain for advanced language model orchestration and Streamlit for an interactive user interface. Key features include content generation, file saving, and downloading, document interaction (QnA with PDFs), and image chatting with Llama 3.3 Vision. LLamaAssist offers a seamless and efficient tool for content creation and document management.
  GitHub: https://github.com/Dawoodsarfraz/LlamaAssist

- **PaperScope**
  PaperScope is a chatbot designed to assist researchers in navigating and analyzing academic papers. Utilizing advanced Retrieval Augmented Generation (RAG) with Mistral 7B and Langchain, PaperScope helps users extract insights, answer specific questions, and summarize complex research documents with ease.
  GitHub: https://github.com/Dawoodsarfraz/Paperscope

- **RoboText Classifier**
  This project leverages RoBERTa and NLTK for text classification on a large 500k-row dataset, aiming to build a model that efficiently classifies text using advanced techniques. The model utilizes dynamic masking to improve generalization by randomly masking tokens during training, enhancing its ability to handle varied contexts. Sentence packing is employed to combine multiple sentences into a single input sequence, maximizing the 512 tokens input limit and enabling the model to process longer documents. Larger batch sizes are used to optimize

training speed and stability, allowing for the efficient processing of multiple samples at once. Finally, the model uses a byte-level BPE vocabulary to tokenize text at the byte level, ensuring robust handling of diverse text, including special characters and misspellings.
GitHub: https://github.com/Dawoodsarfraz/RoboText-Classifier

- **Named Entity Recognition (NER) system using Hugging Face models**
  I built a Named Entity Recognition (NER) system using Hugging Face models. The goal is to identify and classify entities such as people, organizations, locations, dates, and other relevant information from text. I am utilizing DistilBERT, a smaller and faster variant of BERT, for NER tasks. The project leverages Hugging Face's transformers library, where the model is initialized using the AutoTokenizer for tokenizing the input text. The pre-trained model, distilbert-base-cased, is fine-tuned for NER tasks to extract named entities from various sources of text efficiently.
  GitHub: `https://github.com/Dawoodsarfraz/Name-Entity-Recognization-using-HuggingFace`

- **Intellicrew**
  I developed Intellicrew, a multi-agent system built using the CrewAI framework. Intellicrew is designed to streamline complex workflows by coordinating a team of intelligent agents, such as the Researcher and Reporting Analyst, each specialized in handling specific tasks. These agents collaborate autonomously to gather, process, and present information efficiently. The system is powered by Llama 3.2, hosted locally via Ollama, enabling fast, secure, and context-aware language generation. The entire backend is implemented in Python, with optional support for the OpenAI API for enhanced processing capabilities.
  GitHub: https://github.com/Dawoodsarfraz/Intellicrew

- **Electronic Products Recommendation System**
  I built a recommendation system using the Amazon Electronics dataset to enhance product suggestions based on user interactions and product metadata. The dataset contains over 7 million user-product interactions, including ratings and timestamps. I applied techniques like collaborative filtering, content based filtering, and matrix factorization to model user preferences and item similarities. Additionally, I implemented hybrid methods to combine these approaches, improving both the accuracy and diversity of recommendations. The system aims to deliver personalized product suggestions, ultimately boosting user satisfaction and engagement in e-commerce platforms.
  GitHub: https://github.com/Dawoodsarfraz/electronic-products-recommendation-system

- **Stock Market Prediction using LSTM**
  Trained a LSTM model using historical stock market data to predict future stock prices based on past trends. The dataset includes features like opening/closing prices and volume, which are preprocessed into training and testing sets. We used stock data from different companies, such as Microsoft, fetched using Yahoo Finance via the yfinance library. The model is built using TensorFlow, with optimized hyperparameters to improve performance and prevent overfitting.
  GitHub: https://github.com/Dawoodsarfraz/Stock-Market-Prediction-using-LSTM

- **Duplicate Question Pairs**
  Built a model that can identify and detect duplicate question pairs from a given dataset. I used the Quora Question Pairs dataset, which contains pairs of questions asked on the Quora platform. Duplicate question pairs are a common occurrence in various question answering platforms and forums, leading to redundant information and inefficiency in user interactions.
  GitHub: https://github.com/Dawoodsarfraz/duplicate-questions-pair

- **Text Generation using LSTM**
  Developed a word level text generation model using Long Short-Term Memory (LSTM) networks in an unsupervised learning setting. A raw text file was used as the dataset, which was preprocessed by cleaning, tokenizing, and splitting into sequences of words. Each sequence was structured so that a group of preceding words was used as input to predict the next word, allowing the model to learn contextual relationships and language patterns. The LSTM model was trained to capture these sequential dependencies and generate coherent and meaningful text. This project helped deepen my understanding of sequential modeling and natural language generation techniques.
  GitHub: https://github.com/Dawoodsarfraz/Text-Generation-using-LSTM

- **Pakistan Food Price Analysis**
  The primary goal of this project was to analyze and understand the trends, patterns, and factors influencing food prices in Pakistan. By leveraging data, the project aims to provide valuable insights for policymakers, researchers, and the general public to make informed decisions about food consumption, pricing strategies, and potential interventions. The dataset that used is from kaggle. Dataset contains food prices data for Pakistan, sourced from the World Food Programmed Price Database. Data Set is contains on 9723 rows x 14 columns.
  GitHub: https://github.com/Dawoodsarfraz/Pakistan-Food-Price-Analysis

- **Cyber Attacks Classification using Machine Learning**
  In the rapidly evolving landscape of cybersecurity, the ability to quickly and accurately classify cyber attacks is crucial. This project leverages machine learning algorithms to analyze and classify patterns in network traffic, helping to identify and mitigate potential threats. The project includes several machine learning models for cyber attack classification, like Decision Trees, Random Forests, Support Vector Machines (SVM), and Deep Neural Networks.
  GitHub: https://github.com/Dawoodsarfraz/Cyber-Attacks-Classification-using-Machine-Learning

- **Text, Image, Audio and Video Steganography**
  This is an encryption and decryption software developed using Python 3 on the Linux platform. This desktop application can encrypt and decrypt simple text files(.txt, .py), Image files(8, 16, 24 bit depth jpg, jpeg, bmp, png, etc), Audio files(8 and 16-bit Mono and Stereo PCM files), Video files(.avi, .mp4). This application uses the RSA algorithm to perform encryption and decryption of files, though for image and audio files, the algorithm is enhanced to make the encryption and decryption more efficient.
  GitHub: https://github.com/Dawoodsarfraz/Text-Image-Audio-and-Video-Steganography