Amiya Mandal

Greater Noida West, Delhi(NCR) • +91-9958188149 • amiya19mandal@gmail.com • linkedin.com/in/amiya-mandal-445587b2/

I have seven years of experience in backend engineering, software architecture, and deep learning as a seasoned software developer. I optimize code performance using Python, Django, fast-API, NumPy, pandas, and Rust. With my knowledge in natural language processing, microservice architecture, and software deployment, I have successfully completed projects in a variety of industries.

PROFESSIONAL EXPERIENCE

Nagarro, Gurugram — June 2022 – Present

Associate staff engineer

- **Suggestive Tagging**
  - Created a system that utilizes file content analysis to suggest 10-15 relevant tags out of a pool of 3000, enhancing tagging precision and user experience
  - Demonstrated expertise in subject matter by optimizing tag selection and accuracy, resulting in improved quality of content organization.
  - Conducted experiments to evaluate the performance of Transformer-based language models trained on a dataset of 30k examples, achieving notable improvements in accuracy.
  - Designed the system architecture and layout, taking into account scalability, maintainability, and user requirements.
  - Built a set of REST APIs using Django, enabling seamless communication with the aforementioned system from the ground up.
- **Proposal Builder**
  - Experienced in building proposal builder project from scratch and managing its development lifecycle
  - Proficient in implementing vector databases using qdrant for cosine similarity, ensuring fast and reliable performance.
  - Skilled in integrating GPT-3 (text-davinci) to generate results based on user queries.
  - Developed a backend system for session management that utilizes Okta tokens for user authentication.
  - Expertise in creating batch jobs and using sbert for vector embedding to efficiently process and analyze large amounts of content.

UHG(OPTUM), Gurugram — August 2019 - June 2022

Associate Data Scientist

- **Transcription correction engine**
  - Achieved superior transcription quality by leveraging advanced techniques such as fuzzy matching, regular expressions, NLP, and a custom punctuation model to ensure perfect spelling and sentence structure.
  - Streamlined operations and boosted efficiency by utilizing multithreading to expedite processing times and deliver results faster.
  - Conducted a comprehensive analysis of a large corpus of transcriptions to identify and address the most common errors made by the transcription model.
  - Developed and deployed a GECTOR model as a RESTful API, enabling easy and efficient integration with other applications.
  - Built a Rust-based regular expression engine using PyO3 and packaged it as a Python function for seamless adoption in various contexts.
  - Exposed all core functionality as a RESTful API, enabling easy consumption and integration by other applications.
- **ASR engine**
  - Developed a robust pipeline using AI and machine learning to transcribe up to 10,000 calls per day in indigenous languages, providing accurate and efficient speech-to-text conversion.
  - Architected a pub-sub system featuring distributed workers and a centralized master head, with automatic failover mechanism for improved fault-tolerance.
  - Leveraged Python to interface with Kaldi C++ binaries, as well as Azure Blob and Azure Batch transcription services, catering to diverse business domains.
  - Coordinated all processes in a distributed environment through docker-compose, ensuring scalability and minimal points of failure.
- **Call Summarization**
  - Devised a mechanism to summarize lengthy phone calls for Optum's healthcare data using NLP, zero-shot classification, and GECTOR for grammar correction

- Trained and tested multiple transformer-based models, including T5, BART, LONGFORMER, LED, and GPT-2, for summary purposes
- Implemented database operations for ETL and Tableau processes to ingest and display generated data
- Provided a fair summary result, as transformer models are constrained and cannot provide human-like summaries.

- **Search Engine for COVID Questions**
  - Developed a system in response to COVID-19 that allows front-line employees to quickly locate answers to commonly asked questions (FAQs) without needing to read extensive materials
  - Utilized sbert to turn FAQ questions into dense vectors, which were then indexed and stored on disk
  - Built a REST-API for the frontend to retrieve results from an Elasticsearch index of all documents
  - Implemented natural language query comprehension for the system up to a certain point.

KiwiTech, Noida August 2018 - August 2019

Software Engineer

- **PAX**
  - Developed an Angular-based frontend and a Django-based backend for the PAX project's website
  - Built a private charter plane management system where customers can book a private charter plan based on location and filter
  - Built a full backend with authentication in REST-API, which the frontend utilizes
  - Indexed all the geolocations in Elasticsearch
  - Implemented distance calculation as per business requirements.
- **See-Food**
  - This project involves the development of an iOS application that uses augmented reality to portray food. It is backed by Django and AWS's CloudFront for content distribution, with elastic search for auto-suggestions.
  - Developing a full backend with authentication in the REST API that the iOS app uses.
  - Indexing all food and their details in Elasticsearch for easy searching.
  - Indexing all AR files in CloudFront for scalable download requests.
- **Vinified-ML**
  - The data cleansing system utilizes natural language processing (NLP) and fuzzy matching to separate wine names based on their properties (winery, location, vintage, etc.) and determine if two wines are identical.
  - The system was trained and tested using various NLP techniques to split the data into categories such as area, location, and age of the wine.
  - A REST-API was built to enable the frontend to utilize the system.

Untrodden Labs, Delhi December 2017 - August 2018

Python Developer

- **Muffin**
  - User management for the Muffin application was developed using Django Rest.
  - The app was designed to anticipate the user's desired service using speech-to-text recognition and NLP prediction.
  - Rivescript was implemented to make the app more interactive and push notifications were added for reminders.
  - Reminder value prediction was included using a Stanford NLP model.
  - The system utilized REST-APIs for communication between the frontend and backend.

Secninjaz, Delhi December 2016 - December 2017

Python Developer and Researcher

- **WEB-Mine**
  - The career began with developing Scraping services to capture data from social media networks such as Facebook, Twitter, LinkedIn, and Instagram
  - Custom scrapers were built and maintained for each social media site.
  - The system utilized REST-API for communication between the frontend and backend.
  - The captured data was saved in MongoDB for easier searching and access.

EDUCATION

Noida Institute of Engineering & Technology**,** Greater Noida, UP 2012-2016

Bachelor of Technology(Computer Science and Engineering)

- Awards: earned third place in coding hackathon in the third year.

SKILLS & OTHER

Vector-Database, GPT-3, Cosine similarity algorithm, Python, Transformer, pytorch, Django, flask, FastAPI, msSQL, redis, celery, Pyo3, Scrapy, PostgresSQL, Docker, Jenkins, Sagemaker, rabbit-mq, azure-batch-transcription, Software Development Life Cycle, Agile Methodology, Rust.