MAHESH G

TECHNICAL LEAD BANGALORE, INDIA 9972199818

DETAILS

Bangalore
India
9972199818
mahesh.lakshmi.in@gmail.com

SKILLS

- Python
- Pandas
- Tensorflow
- Python Docx
- Scikit-Learn
- Git
- SQL
- AWS
- Langcahain
- Promp Engineering
- Large Language Models
- Generative AI

LANGUAGES

English

Hindi

Telugu

Kannada

CERTIFICATIONS

- Tensorflow Developer Certificate by Google
- Neural Networks and Deep Learning Coursera

PROFILE

- **Experience:** Seasoned Machine Learning Engineer boasting 6+ years of deep expertise in both NLP and Computer Vision.
- **Achievements:** Demonstrated success in crafting text and image classification models, along with sophisticated data regression models.
- **Predictive Modeling:** Skilled in leveraging predictive data modeling techniques to unearth valuable insights and devise actionable solutions addressing intricate business challenges.
- **Deployment:** Specialized in developing and transitioning machine learning models to production environments, ensuring scalability and reliability.
- **Communication:** Possess robust communication and presentation acumen, facilitating collaboration and knowledge sharing.
- **Passion:** Driven by a fervor for harnessing technology to address pressing real-world dilemmas.
- **Tech Stack:** Proficient in building diverse applications leveraging Langchain and Azure Open AI GPT-4 capabilities.

EMPLOYMENT HISTORY

Technical Lead at Accenture, Bangalore
April 2021 — Present

1. Promotion

- Earned promotion from Senior Software Engineer due to outstanding contributions and consistent delivery of projects within stringent timelines.

2. Project Management

- Managed the project from start to finish, ensuring that it was completed on time and within budget.

3. Team Leadership

- Led a team of 5 professionals, consistently delivering projects on schedule.

4. Team Management

- Oversaw team workload, ensuring timely completion of projects that met the highest quality standards.

5. Mentoring

- Mentored team members, fostering skill and knowledge development.

6. Technical Achievement

- Designed and implemented an end-to-end solution using deep learning NER models to extract abbreviations and definitions from clinical documents.

7. Impact

- Reduced abbreviation table creation time for the medical writing team from 2-3 hours to mere minutes per document.

8. Deployment

- Successfully deployed the Abbreviation Solution as a web portal using AWS services such as Sagemaker, Lambda, and S3.
- Developed and deployed an Image Classification model using CNN for classifying pages into valid or non-valid pages in pdf documents and deployed in AWS Sagemaker.

9. Technological Utilization

- Employed YOLO for handwritten data recognition in PDFs, enhancing document quality checks.
- Leveraged Generative AI and Langchain for efficient redaction and synopsis creation.

Projects:

Automated Abbreviation Web Portal

- **Problem**: Medical writers were spending several hours manually compiling lists of abbreviations and definitions from clinical documents.
- **Solution**: Developed an automated web portal enabling users to upload clinical documents and download abbreviation lists as Word files. This portal harnesses a deep learning NER model, trained on a dataset of 100,000+ clinical documents, to extract abbreviations and definitions.
- **Benefits**: The portal significantly reduces the time medical writers spend compiling abbreviation lists and enhances accuracy through the deep learning model's precision.
- **Skills and Tools**: Hugging Face, AWS, Python, Pandas, Docx

QR Phase-3 – Application development for quality review of documents uploaded by the users in the client's portal

- **Problem**: The client aimed to enhance the quality of user-uploaded documents on their portal. This included ensuring that user-inputted data matched the content within the document and verifying the document's legibility.
- **Solution**: Developed an automated application to review documents upon upload. This application cross-referenced user-inputted data with the document content and identified valid and legible pages within the PDFs. Incorporated a series of 35 distinct checks and utilized multiple image classification models, alongside object detection models via YOLO and CNN architectures. Leveraged various OCR engines, including AWS Textract and Pytesseract, for text extraction.
- **Benefits**: The application markedly elevated the quality of documents on the portal, streamlining user experience. Moreover, it realized significant time and cost savings for the client by automating the previously manual quality review process.
- **Skills and Tools**: Python, Pandas, AWS, Sagemaker, YOLO, CNN, AWS Textract, Pytesseract

Synopsis Creation

- **Problem**: Medical writers faced inefficiencies due to extensive time spent on analyzing and drafting summaries for clinical documents.
- **Solution**:Developed a portal enabling writers to upload clinical documents and instantly receive summaries. Leveraged Python docx to formulate summaries for designated sections of the body document, with entities extracted from key content areas using AWS Comprehend and Azure OpenAI. Able to generate the summary in Layman terms.
- **Benefits**: Dramatically cut summary creation time for medical writers, reducing the process from hours to just 2-3 minutes per document.
- **Skills and Tools**: Python docx, AWS Comprehend, Azure OpenAI, Langchain, Prompt Engineering

Redaction Bot

- **Problem**: The publishing or sharing of numerous clinical and research documents across teams presented a challenge due to the presence of Personal Identification Information (PII) within them. The necessity to redact PII in compliance with regulations such as HIPPA, GDPR, and Health Canada from documents, which could span up to 3,000 pages, placed significant time demands on the writers' team.

- **Solution**: Developed an application empowering writers to forward documents to an RPA bot via email. Upon receipt, this bot initiated a Python-based procedure to identify and redact PII from PDFs. To spot PII data, AWS Comprehend and Hugging Face's NER pretrained BERT Models were employed. The actual masking of PII in PDFs was facilitated by AWS Textract and the Python Fitz library. Post-redaction, the Azure OpenAI Large Language Model Chat GPT-4 was leveraged to validate the Python bot's redaction performance.
- **Benefits**: By streamlining the redaction process and enhancing quality control checks, the application significantly expedited the time required by the writers' team to mask PII within PDF documents.
- **Skills and Tools**: PyMuPDF, Fitz, AWS Textract, AWS Sagemaker, HuggingFace, Langchain, Azure OpenAI, GPT-4.

Safety Drug Report Generation:

- **Problem**: The Pharmaco Vigilance team faced challenges in creating safety drug reports, specifically the PBRER and DSUR documents. These reports, particularly the body document, can span 80 to 90 pages. Consequently, the Clinical writer team had to invest significant time in manually producing summaries for the PBRER and DSUR reports.
- **Solution**:Developed a web application where the medical writers can upload the body document of either PBRER or DSUR. Once uploaded, the document is processed, and summaries are automatically generated. This is achieved by leveraging AWS Lambda to invoke Azure OpenAI, which identifies key data and populates predefined placeholders in the fixed PBRER or DSUR templates. Flexibility is maintained as template modifications can be accommodated with minor config changes to placeholders.
- **Benefits**:The application significantly expedited the summary creation process for PBRER and DSUR reports. Previously time-consuming, the writers can now produce these summaries in under 5 minutes, enhancing efficiency and accuracy.
- **Skills and Tools**: AWS S3, AWS Lambda, AWS Sagemaker, AWS S3 Webkit, Langchain, Prompt Engineering, Azure OpenAI, GPT-4 model.

Software Engineer at Wipro, Bangalore
August 2017 — April 2021

Built and deployed a text classification API for Wipro Ehelpline

- Built a text classification API to classify helpline tickets that come to Wipro Ehelpline, an internal product used by Wipro like Service now.
- Moved the previous API from Elastic Search to Machine Learning, which significantly improved the accuracy of the API.
- Improved the text classification model performance using pre-trained word embeddings and RNN up to 90%.
- Deployed the model as a Flask API for the Ehelpline product.

Projects:

Classifier Bot API for Ehelpline Tool

Problem: Wipro Ehelpline is an internal product used by Wipro to manage helpline tickets. The previous API for classifying helpline tickets was based on Elastic Search, but it was not very accurate.

Solution: I built a new text classification API using Machine Learning. This API is significantly more accurate than the previous one, and it can be used to classify helpline tickets with up to 90% accuracy.

Benefits: The new API has several benefits, including:

- **Increased accuracy**: The new API is significantly more accurate than the previous one, which means that helpline tickets can be classified more accurately.
- **Reduced manual effort**: The new API can automatically classify helpline tickets, which reduces the amount of manual effort required to manage the helpline.

- **Improved customer experience**: The new API can automatically route helpline tickets to the correct department, which improves the customer experience.

Skills and Tools: Neural Networks, Keras, Tensorflow, Scikit-Learn, Pandas, Flask

Prediction of loan Closing Date (Gallagher Solutions)

Problem: Gallagher Solutions is a financial services company that provides loans to businesses and individuals. The company wants to be able to predict the closing date of a loan application so that they can better manage their resources.

Solution: I developed a machine learning model to predict the closing date of a loan application. The model was trained on a dataset of historical loan applications. The model was able to reduce the mean absolute error (MAE) from 24 to 10

Benefits: The benefits of model include:

- Improved efficiency: The company can now better manage their resources by knowing when to expect loan applications to close.
- Increased accuracy: The company can now make more accurate decisions about loan applications.
- Reduced risk: The company can now reduce the risk of loan defaults by identifying loans that are more likely to close on time.

Skills and Tools: Machine Learning, Data Science, Python, Scikit-Learn, Flask

EDUCATION

Post Graduate Diploma, Great Lakes Executive Learning, Bangalore
June 2019 — December 2021

Btech, PES University, Bangalore
May 2013 — May 2017

Completed with a CGPA of 7.98

12th Standard, Varadhadri PU College, Bangalore
April 2021 — April 2013

Completed with an Distinction of 88%

10th Standard(SSLC), Young India School, Bagepalli Taluk,Chikkaballapur District, Karnataka
April 2010 — April 2011

Completed with an distinction of 91%