RISHABH DWIVEDI
Mobile: 9999597047 | **Email:** dwivedi.rishabh1995@gmail.com | linkedin.com/in/rishabh-dwivedi-a7623094

SUMMARY

Rishabh Dwivedi is a Strategic **AI, Machine Learning, and NLP leader** with ~7 years of experience architecting and deploying **enterprise-scale multi-agent system, Gen AI, predictive modeling, and deep learning solutions** across consulting, analytics, and technology sectors. Proven expertise in **LLM-driven system design**, **neural network–based text analytics**, and **intelligent automation** using Python, LangChain, LangGraph, FastAPI, and cloud-native architectures. Experienced in leading cross-functional teams to build **end-to-end ML pipelines**, optimize large-scale data processing, and implement **robust MLOps frameworks** for production AI systems. Specialized in **Natural Language Processing**, **transformer-based architectures**, and **deep learning models** for automated content generation, document understanding, and unstructured text classification. Recognized for blending **technical depth with strategic vision**, enabling organizations to scale AI adoption through innovation, reliability, and data-driven impact.

WORK EXPERIENCE

AI and Data Consultant (Fast tracked for Senior Consultant)
Deloitte, Gurugram, India | AI & Data **[January 2024-Present]**

Multi-Agent System for Job Monitoring

- **Led** a team of 4 AI engineers to Design and implement a scheduler-driven **Multi-Agent System** using **LangChain** and **LangGraph** to automatically fetch failed SAP batch jobs, perform root-cause analysis via a **RAG pipeline** (SAP tickets + knowledge articles), apply fallback LLM inference, and trigger automated notifications through **SMTP** and **Microsoft Teams**.
- **Architected** an enterprise-scale **Knowledge Base** combining a **Neo4j knowledge graph** with a **Postgres + pg_vector** hybrid vector store, enabling **dense + sparse semantic retrieval** across 2M+ documents. Integrated embeddings, metadata, and entity linking for context-aware reasoning and improved retrieval precision and recall.
- Built the system with **reusable, modular tools and connectors**—including **RFC and OData interfaces**, SMTP email service, Teams messaging adapter, and a central Scheduler—allowing these capabilities to be **plugged into future Multi-Agent workflows** and extended to new enterprise automation use cases.
- Achieved **~70% reduction in manual triaging** of SAP job failures across diverse error classes by automating failure identification, classification, and RCA workflows.
- Integrated **Langfuse and LangSmith** for observability, evaluation, and governance across agent actions, prompts, and LLM outputs.

MIM Communication

- Built a robust **Generative AI** solution leveraging **GPT-4o** on Langchain to provide automated periodical updates on P1 Major incident tickets
- **Led** a team of 3 for solutioning, Sprint planning, effort estimation and Feature enhancements
- **Developed +20** APIs in **Django Rest Framework** and **Docker** containerized to deploy on a **SaaS** platform as a Microservice
- Made it compatible for integration with SNOW and JIRA with a **WebHook** and independent scheduler for each incident using **Airflow**
- **Reduced** manual requirement of filling template with summaries of Live call conversations and ticket updates using **AWS chime** by **~90%**
- **Increased** the overall efficiency of the MIM process by **~50%.**

Clustering service

- **Led** a team of **5** and developed Python FastAPI clustering microservice processing **500K+** records with asynchronous workflows, **OpenSearch** integration, and real-time data preprocessing for **enterprise-scale** analytics.
- **Implemented 8+** clustering algorithms (KMeans, BERTopic, GMM, BIRCH, Hierarchical Agglomerative, OPTICS, DBSCAN, Fuzzy) with UMAP dimensionality reduction and hyperparameter tuning, achieving 0.08+ silhouette scores on production datasets.
- **Implemented SonarQube** compliance, and optimized performance from **4 seconds** (10K) to **2.8 hours** (500K records) with comprehensive testing across multiple data domains.
- **Built** RESTful APIs with **SQL** Server integration, **AWS** services, priority-based **job queuing**, and robust error handling for enterprise reliability with comprehensive transaction management.
- **Delivered** advanced clustering metrics with **memory optimization** (**+367MB** efficiency gains) and cost-effective processing through intelligent sampling and batch operations.

LLM Service

- **Led** a team of 3 developers to build Python **FastAPI** microservice with **Redis**-backed priority queue managing 4-level priority LLM processing jobs with rate-limited throughput control **(10 req/sec, 70K tokens/6sec)** for **large-scale text summarization** and AI analysis operations.
- **Implemented** intelligent text chunking with multi-threaded background processing, automatic token calculation using **tiktoken**, and consolidation workflows ensuring **99%+** processing reliability across **63K+** token document operations with seamless chunk-to-summary aggregation.
- **Designed** enterprise-grade failsafe mechanisms with **Redis** status tracking, structured logging, automatic task recovery, and graceful error handling managing background worker threads with exponential backoff retry logic for uninterrupted LLM service operations.
- **Developed** secure RESTful endpoints with header-based authentication, AWS Services integration, and multi-format file processing (PDF, DOCX, TXT) from S3 enabling seamless AI service integration with real-time status monitoring.

- **Architected** multi-environment deployment supporting **APIM** configurations with **Docker** containerization, automatic scaling based on queue depth metrics, and stateless worker architecture using atomic Redis operations for high-availability LLM processing.

Machine Learning Engineer
Hewlett-Packard Enterprise | Global Marketing Analytics, Bengaluru, India **[October 2021-December 2023]**

1. Propensity-to-Buy Models

Objective:
- Build robust predictive models to target **high propensity to transact** customers in a defined prediction window for various HPE Offerings.
- Validate model on Out-of-Time data to analyse the lift in buyers capture for **Top 20%** recommended customers.
- Deploy model to recommend top accounts from more than **100K customers in HPE account base** on weekly basis.

Approach:
- Understand business requirement (customer base, product/service offered etc.) from stakeholders and ideate strategy and methodology for solution.
- Implement end-to-end Model life cycle:

1. **Problem Statement**: Define problem statement, target customers and model cohort by utilizing customer profile (Firmographics, HPE trans., etc.)
2. **Data Retrieval** : Account level ~ **4K** transformed features from First-party (Transaction, sales pipeline, workloads, contract, etc) and Third-party (IT spend potential, Digital activity, etc) for various time periods.
3. **Data Wrangling** : Treatment for missing values, outliers, encoding, Normalization etc for continuous and categorical features followed by treatment of imbalanced data (Undersampling – CNN rule, ENN rule, Neighbourhood cleaning rule etc. and Oversampling – SMOTE, etc.)
4. **Feature engineering and selection** : Check for Collinearity and multicollinearity (VIF etc.) followed by Feature extraction using dimensionality reduction techniques (like embedded methods, PCA, RFE, factor analysis etc.).
5. **Model selection** : Train and Hyperparameter tune Supervised ML models (Random Forest, XGBoost, Neural Networks, etc.) and validate on Out-of-Time data with **Lift and Gains analysis**, aim to capture on an average **>90% accounts in Top 2 deciles**.
6. **Model Deployment in Production** : Python and SQL automation codes for weekly scoring and recommendation of top accounts using **Git**.
7. **Model Supervision** : Monitor model performance on Quarterly conversions **(within ± 20% range)**

- Build weekly reports with various account level data dimensions (Spend potential, Potential Workloads presence, reasons for selection etc.)

2. Explainable AI Objective:
- For non-technical PTB consumers, generate account level **reasons for recommendations** for **Top 20%** recommended customer.

Approach:
- Trained a base-level ML algorithm using the data and a binary target variable and generated list of top **250** variables using feature importance.
- Calculated **lift/factor** across each variable i.e. Ratio of account level value of the variable to the average value of the variable for a non-buyer.
- On the basis of factor and variable data density, shortlisted at most 5 top variables from each source as a final list of variables.
- For each account, calculate factor for all variables and select variables which surpass a given **threshold**.
- Formulated a **selection order** and description dictionary, which generates an **English language based explanation** in the sequence of top variables from each source. This provides a holistic business understanding for the PTB's recommendation.
- Created an **automation script** for periodical appending of reasons for recommendation with the **100K** active customer base.

Literature:
- Published an internal white paper titled – "**Explainable AI: A statistical approach to interpret AI/ML based Account recommendations**"

Analytics Practitioner
Brillio | Data science, Analytics Department, India **[July 2019–September 2021]**

1. Automated Complaint-log Classification using RNN

Objectives:
- ML-Based **Automated Complaint-log Classification** of complaint-logs into various categories for the OS product group.
- Generate periodical reports based on insights from the complaint-log classification.

Approach:
- Complaints-logs Classification using **multi-class classification** for the following three levels:
  - Classify into Operating System (OS) and Non-Operating System (Non-OS) calls.
  - Classify into Type of Operating system—Windows, ESXi, and Linux.
  - Classify into Complaint Symptoms—OS issue, OS crash, etc.
- **Insights Generation** and Automation of Business Report Creation using Python:
  - Frequency Analysis: Visualization across levels
  - Trend Analysis: Over various time windows
- Formulated an architecture for the project consisting of steps for ingestion of data from Teradata, processing of data, deep neural network framework pipeline (**TensorFlow and Keras**), and generation of insights.
- Built a **Natural Language Processing** and business logic-enabled automatic classification of unstructured call logs into OS and Non-OS calls.
- Formulated a "**labelling process for unstructured text data**" to use in Supervised Machine Learning.
- Converted call logs into Feature vectors by transfer learning using **Multilingual BERT embeddings.**
- Built a Deep Neural Net model using **Sequence Model (Bi-directional GRU)** and multiple feature engineering layers to build a multi-class classification model for Type of OS and Symptoms.

2. Root cause analysis (RCA) Objective:
- Develop a **B2B SaaS** to **automate network analysis and resolution process** by assessing network occurrences such as faults and congestion and provide **prescriptive/reactive maintenance.**
- Conceptualize and implement a hierarchical root cause analysis (RCA) using **Closed-Loop Automation (CLA)** – Automate ticket resolution process utilizing Machine learning models for RCA.

Approach:

- Developed a hierarchical RCA tree utilizing **multi-class classification** to classify into Devices (Cisco, Aruba, etc), network types (Wi-Fi/LAN), and root causes (Wi-Fi authentication, Policy issue, Port error, etc).
- Generated, collected, and labelled data from Radius server/WLC with the inputs of Network SMEs.
- Built pipeline in **AWS SageMaker** notebooks to extract important features, vectorize using various vectorization techniques (Textual data—**TFIDF/CountVectorizer/Word2Vec**, Categorical data—**OneHot encoder**); trained on Supervised machine learning algorithms (**Decision tree/Random Forest/XGBoost)**; and conducted model evaluation, selection, and registry using **MLflow** to manage Machine learning lifecycle.
- Responsible as the SPOC and Data science lead for a team of 4, understanding the problem statement, and planning of 3 phases.
- Periodically planned discussions with the Client, SMEs, feature leads, and Project leads for multiple use cases across multiple phases for successful and quality-assured delivery.

ACADEMIC QUALIFICATIONS

Course Name	College/School/University	Year of Passing	Marks Obtained
M.A. Economics	Delhi School of Economics	2019	**55**
B.A.(H) Economics	University of Delhi	2016	**78**
XII (C.B.S.E.)	Little Scholars Sr. Secondary	2013	**90.60**
X (C.B.S.E.)	Little Scholars Sr. Secondary	2011	**9.4/10**

ACADEMIC ACHIEVEMENTS AND ACCOMPLISHMENTS

- Secured All-India Rank of **40** in DSE M.A. Entrance Test and **Top 36** at All-India level in ISI M.Sc. Quantitative Economics Entrance test [2017]
- Awarded with **Academic Excellence** for 2 Consecutive years; obtained **9+/10** in Statistics and Operational Research. [2013-2016]
- Distinguished as **School Topper** in C.B.S.E Board Exam as well as awarded with Certificate of Merit for securing Rank **1/90** in school [2013]

ACADEMIC PROJECTS

Title : 'Decomposition of Total Factor Productivity and Energy Efficiency in India's Paper and Pulp Industry at aggregate and state level'

- Decompose Total Factor Productivity growth, using Time-varying Stochastic Production Frontier (Parametric approach) into change in Technical Efficiency (TEC), change in technological progress (TP), Scale efficiency change (SEC) in India's paper and pulp industry.
- Study the trend of its component across pre-reform (before 1980), reform (1980-90) and post- reform period (1991-98). In addition, analyse the Individual growth trends as well as a comparative analysis for four major states- Maharashtra, West Bengal, Andhra Pradesh and Gujarat

Title : 'An Analysis of Energy Consumption pattern at Household level in India'

- Analyse disaggregated and heterogeneous energy consumption pattern across economically developed and developing states of India during 2009-12
- Utilised Linear Approximation of Almost Ideal Demand System (LAAIDS) to estimate price and income elasticities for all the energy consumption items at the household level.
- The predicted estimates of elasticities produce the impact of various national and state level policies, designed to discourage inefficient fuel consumption and address climate change concerns.

PUBLICATION AND CERTIFICATIONS

- **Authored** a comprehensive **Prompt Engineering Guide** covering foundational concepts through advanced techniques, ethical considerations, and AI platform comparisons to empower teams with best practices for accurate, efficient, and responsible AI solution development.
- **Co-authored** a technical paper on **AWS Chime-based live transcription solution**, designing a Python framework that joins and transcribes meetings across Teams, Zoom, and Webex platforms with real-time S3 storage for enhanced meeting analysis and system integration capabilities.
- Completed **NVIDIA-Certified Associate: Generative AI LLMs certification**
- **Published** an article on Medium titled "Labelling unstructured text data in Python"
- Completed **AWS Machine learning- Specialty** certification
- **Deep learning specialization** with 5 courses by **Andrew Ng** (Coursera)