

Hafiz Muhammad Noor

Machine Learning Engineer

Email: hafizmuhammadnoor01@gmail.com

LinkedIn: [linkedin.com/in/hafiz-muhammad-noor](https://www.linkedin.com/in/hafiz-muhammad-noor)

Website: <https://hafiz-muhammad-noor.site>

Phone: +92 316-4042403

GitHub: <https://github.com/cmd-HMN>

Address: Islamabad, Pakistan

PROFILE SUMMARY

Results-driven Machine Learning Engineer and Data Scientist with a strong foundation in predictive modeling and applied deep learning. Expertise in building models for computer vision and risk forecasting problems, with a focus on real-world, decision-ready outputs. Adept at designing data pipelines, fine-tuning model architectures, and validating performance through rigorous testing. Committed to writing clean, well-calibrated models that hold up under real-world conditions.

SKILLS SUMMARY

- **Languages:** Python, Rust, SQL, Bash
- **Frameworks/Libraries:** PyTorch, TensorFlow, Scikit-learn, Keras, XGBoost, LightGBM, CatBoost
- **Techniques:** Computer Vision, Survival Analysis, Probabilistic Forecasting, Feature Engineering, Hyperparameter Tuning
- **Testing:** Pytest, K-Fold Cross-Validation, Model Calibration, Optuna
- **Tools:** Git, Docker, Kaggle Notebooks, Google Colab

PROJECTS

BBQ-RAG – Visual Document Retrieval & Late-Interaction Search Engine

Key Technologies: Rust, PyO3, AVX2 SIMD, PyTorch, NumPy, Vision-Language Models, SQLite

- Engineered a **fused AVX2/FMA SIMD MaxSim** compute kernel in Rust with 4-way query-unrolled dot-product pipelines, achieving **90,737** document pages scored per second on multi-core CPUs.
- Designed and executed a **comprehensive benchmark suite** comparing five maxsim implementations validating strict numerical equivalence and demonstrating **4x–8x speedup** over standard PyTorch baselines at scale.
- Architected an **end-to-end visual document retrieval pipeline** combining vlm embeddings with an automated PDF directory watcher, persistent SQLite embedding metadata tracking, and **client-side multimodal RAG generation** with graceful offline fallback when no API key is available.

CSIRO – Image2Biomass Prediction

Key Technologies: PyTorch, ConvNext, EfficientNet, Computer Vision, Python

- Fine-tuned a **ConvNext-Tiny** architecture in a single unified training-and-inference notebook, achieving weighted **R² of 0.53** on the private leaderboard.
- Built a separate **EfficientNet-B0** pipeline across dedicated train and inference notebooks to benchmark architecture performance independently.
- Implemented **Stratified K-Fold cross-validation** with **early stopping** and **model checkpointing** to prevent overfitting and retain best-performing weights across folds.

Wildfire Risk Prediction – WiDS Datathon 2026

Key Technologies: Python, LightGBM, Scikit-learn, Survival Analysis, StratifiedKFold

- **Ranked 131 out of 1,754 teams (top 8%)** in the WiDS Global Datathon, predicting wildfire escalation risk using discrete-time survival modeling.
- Built a multi-horizon **LightGBM** framework to forecast calibrated hit-probabilities at 12h, 24h, 48h, and 72h windows, handling right-censored event data.
- Applied **isotonic regression** to enforce monotonicity across prediction horizons and calibrate raw model outputs into decision-ready probabilities.
- Validated performance using a 5-fold Stratified K-Fold pipeline, achieving a hybrid score of **0.96** (weighted C-index + Brier score) on out-of-fold predictions.

Peeler – Async Scraper

Key Technologies: Python, Asyncio, argparse, UTRNet (OCR), Rich (CLI), Custom Logging

- Designed a **config-driven dispatcher architecture** that automatically routes execution to the correct extraction pipeline (website, PDF, or XML) based on a JSON config file, without requiring code changes to add new source types.

- Integrated **UTRNet**, a deep learning-based Urdu OCR model (pretrained weights via Hugging Face), to extract Urdu text from scanned/image-based PDFs where standard parsing fails.
- Built a **CLI interface** with structured logging and rich console output, including graceful shutdown handling for interrupted or cancelled async processes.
- Implemented **modular, async extraction pipelines** per source type, isolating errors per pipeline so a failure in one format doesn't halt the others.

CERTIFICATIONS

IBM AI Engineering Professional Certificate - Coursera

- Covered deep learning, generative AI, NLP, and model deployment using PyTorch, TensorFlow, and Hugging Face.

IBM Machine Learning Specialization - Coursera

- Covered supervised and unsupervised learning, gradient boosting, MLOps, and model validation using Scikit-learn.

EDUCATION

Bachelor of Science in Computer
COMSATS University Islamabad

Expected Feb 2028
Islamabad, Pakistan

LANGUAGES

- **English:** Fluent, professional proficiency in verbal and written communication
- **Urdu:** Native
- **Punjabi:** Native