

# NIHAL MORSHED

nihalmorshed2000@gmail.com | +880-162-3240879 |  nihalmorshed2000

## Education

---

### Rajshahi University of Engineering & Technology Bachelor of Science in Computer Science & Engineering

Kazla, Rajshahi-6204, Bangladesh

Expected to graduate in May, 2025

Current CGPA: 3.64

• **Relevant Coursework:** OOP, Algorithms, Data Structures, Operating Systems, Neural Network & Fuzzy Systems, Compiler Design, Digital Image Processing, Microprocessors & Assembly Language.

## Skills

---

Languages: C, C++, Java, Python, HTML, CSS, JavaScript, Dart, SQL

Frameworks: Flutter, Bootstrap

## Research Experience

---

### Rajshahi University of Engineering & Technology (RUET) Department of Computer Science & Engineering

- Led research on development of intelligent system to detect and classify abusive comments in the Bengali language.
- Presented paper "Instance Hardness Threshold for Effective Sampling in Bangla Cyberbullying Detection Using Transformer Models" at the IEEE International Conference on Computer and Information Technology 2024, Bangladesh (Forthcoming in IEEE Xplore) [Available Here](#)
- Utilized transformer-based models for achieving improved classification metrics with BERT achieving a 93.11% F1-score and ELECTRA reaching 91.34%.
- Collaborated with team of two researchers to implement and test solutions.

## Professional Experience

---

### Niharon Technolgies

Junior Software Engineer (Flutter) - Mobile Application Development Team

Rajshahi

Aug 2022 – Jan 2023

- Built UI components using Flutter to materialize customers' requirements and software operations model
- Designed and implemented JSON data models to automate backend data handling processes
- Integrated and designed the server-end API models and functions to automate backend data handling processes
- Designed and implemented functions to regulate user authentication and verification processes for multiple applications

# Projects

---

## Sentiment Analysis Using DistilBERT Transformer Model

July 2024

- Implemented a sentiment analysis model using DistilBERT transformer architecture, achieving 94.86% ROC AUC Score and 92.86% F1 Score on the Twitter Airline Sentiment dataset.
- Used PyTorch and Hugging Face's Transformers library to fine-tune the pre-trained DistilBERT model, which operates 60% faster than BERT while maintaining 95% of its performance.
- Engineered a data processing pipeline using Pandas and NumPy to prepare thousands of airline-related tweets for sentiment classification.
- Optimized model training through GPU acceleration, significantly reducing computation time while maintaining high accuracy.
- Utilized Scikit-learn and Matplotlib for comprehensive model evaluation and performance visualization, demonstrating the model's effectiveness in sentiment classification.

**[GitHub Repository Link](#)**

## Bank Customer Churn Prediction Using Neural Networks

July 2024

- Developed an Artificial Neural Network (ANN) model using TensorFlow and Keras to predict bank customer churn with 86.4% accuracy.
- Implemented comprehensive data preprocessing including categorical encoding, feature scaling, and dataset splitting using NumPy and Pandas.
- Engineered a balanced neural network architecture with two dense layers, optimizing for both performance and computational efficiency.
- Performed model evaluation using confusion matrix and analyzed 11 key customer features including credit score, geography, and banking behavior.
- Created data visualizations using Matplotlib and Seaborn to communicate model performance and customer insights effectively.

**[GitHub Repository Link](#)**

## PawVision: CNN-Powered Image Recognition Model

Jul 2024 - Aug 2024

- Developed a robust Convolutional Neural Network (CNN) model using TensorFlow and Keras, achieving 90.81% accuracy in classifying cat and dog images.
- Implemented comprehensive data augmentation techniques including rescaling, shearing, zooming, and horizontal flipping to enhance model generalization and prevent overfitting.
- Designed and optimized a sequential CNN architecture incorporating multiple convolutional and pooling layers, followed by dense layers for effective feature extraction.
- Utilized the Adam optimizer and binary cross-entropy loss function for model training, ensuring optimal convergence and classification performance.
- Executed thorough model evaluation on unseen data validating the model's reliability and robustness in real-world applications.

**[GitHub Repository Link](#)**

## Predictive Modeling of Stock Market Trends with LSTM Architectures Aug 2024

- Engineered a stock price prediction model using stacked Long Short-Term Memory (LSTM) neural networks in Python, using Keras and TensorFlow frameworks.
- Implemented comprehensive data preprocessing pipeline including feature scaling and temporal data structuring to optimize model performance and accuracy.
- Designed an advanced architecture combining multiple LSTM layers with dropout regularization, effectively preventing overfitting and enhancing model generalization.
- Utilized Pandas and NumPy for efficient data manipulation, incorporating historical stock data to identify and predict market trends.
- Created detailed visualizations using Matplotlib to analyze model performance across various market conditions, demonstrating strong predictive capability in tracking market trends.

**[GitHub Repository Link](#)**

## Article Spinner Using Second Order Markov Model Aug 2024 – Sep 2024

- Developed a machine learning-based article spinner utilizing second-order Markov Model and natural language processing techniques for generating diverse text variations.
- Implemented text processing pipeline using NLTK library for tokenization and TreebankWordDetokenizer for sentence reconstruction while maintaining grammatical accuracy.
- Processed and analyzed the BBC Text Classification Dataset using Pandas to ensure balanced text generation across multiple categories.
- Built a probabilistic word prediction system based on preceding word pairs to generate coherent text sequences.
- Incorporated random selection techniques using NumPy to enhance text variability while preserving readability.

**[GitHub Repository Link](#)**

## TF-IDF Movie Recommender System Jun 2024 - Jul 2024

- Developed a movie recommendation system using TF-IDF vectorization on the TMDB 5000 Movie Dataset, enabling personalized content suggestions based on genres and keywords.
- Implemented comprehensive data preprocessing pipeline to extract and combine movie genres and keywords, enhancing the quality of recommendations.
- Utilized Scikit-learn to create TF-IDF vectors and calculate cosine similarity scores between movies, ensuring accurate similarity measurements.
- Applied NumPy and Pandas for efficient data manipulation and analysis, streamlining the recommendation generation process.
- Created data visualizations using Matplotlib to analyze and validate recommendation patterns and system performance.

**[GitHub Repository Link](#)**

## Handwritten Digit Recognition Oct 2023 – Dec 2023

- Utilized industry-standard Python libraries - Sci-Kit Learn, Pandas, Numpy, and Matplotlib.
- Implemented a Support Vector Classifier (SVC) to train the model.
- Successfully trained using the sample distribution produced from live-captured photos.
- Used Matplotlib to make visually appealing representations of the training data output.

**[GitHub Repository Link](#)**