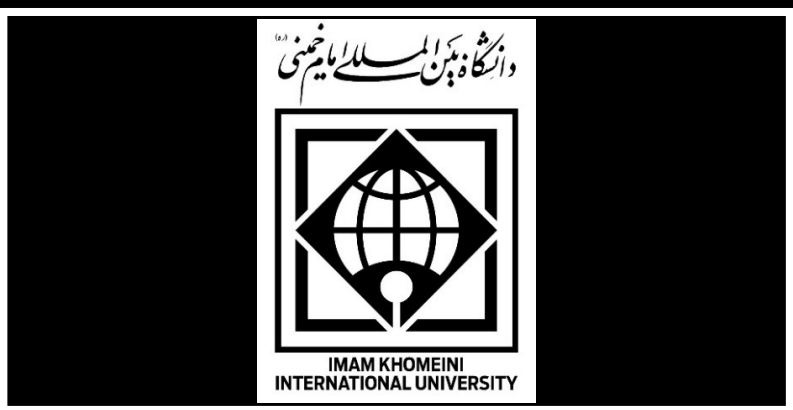

Final Term Project: Binary Classification Using Deep Learning Models

1 PROJECT OVERVIEW:

This project focuses on solving a binary classification problem using deep learning techniques. It includes working with CNNs, LSTMs, hybrid CNN-LSTM models, GRUs, vanilla LSTMs, and autoencoders as feature extractors. The project aims to evaluate and optimize each model's performance using comprehensive metrics and draw conclusions based on comparative analysis.

2 DATASET SELECTION:

For this project, the ideal dataset is the "Diabetes 130-US hospitals for years 1999-2008" dataset available from the UCI Machine Learning Repository. This dataset includes information about diabetes patients and is suitable for a binary classification task predicting readmission within 30 days.

URL: https://archive.ics.uci.edu/ml/datasets/Diabetes+130-US+hospitals+for+years+1999-2008

3 PROJECT STEPS:

3-1 STEP 1: DATA PREPROCESSING

- Objective: Prepare the dataset for model training.
- Tasks:
  - Load the Diabetes dataset and perform initial data exploration.
  - Handle missing values, categorical features, and outliers appropriately.
  - Normalize or standardize numerical features as required.
  - Encode categorical variables if necessary.
  - Split the dataset into training, validation, and test sets.

3-2 STEP 2: CNN MODEL

- Objective: Implement a Convolutional Neural Network tailored for tabular data.

- Tasks:
  - Design a CNN architecture suitable for the Diabetes dataset.
  - Train the CNN model on the training set and validate on the validation set.
  - Evaluate the model's performance on the test set using accuracy, precision, recall, F1-score, and confusion matrix.

3-3 STEP 3: LSTM MODEL

- Objective: Utilize Long Short-Term Memory networks for sequential data analysis.
- Tasks:
  - Design an LSTM model optimized for sequential features in the dataset.
  - Train the LSTM model on the training set and validate on the validation set.
  - Evaluate the model's performance on the test set and compare metrics with the CNN model.

3-4 STEP 4: HYBRID CNN & LSTM MODEL

- Objective: Combine CNNs and LSTMs to leverage both spatial and sequential information.
- Tasks:
  - Develop a hybrid model architecture incorporating CNN layers for feature extraction followed by LSTM layers.
  - Train the hybrid model on the training set and validate on the validation set.
  - Evaluate the model's performance on the test set and compare results with standalone CNN and LSTM models.

3-5 STEP 5: GRU AND VANILLA LSTM COMPARISON

- Objective: Compare the performance of Gated Recurrent Units (GRUs) with vanilla LSTMs.
- Tasks:
  - Implement both GRU and vanilla LSTM models.
  - Train and validate both models on the training and validation sets.
  - Evaluate and compare their performance on the test set using appropriate metrics.

3-6 STEP 6: AUTOENCODER AS FEATURE EXTRACTOR

- Objective: Use an autoencoder for unsupervised feature extraction.

- Tasks:
  - Design and train an autoencoder using the Diabetes dataset.
  - Extract features using the encoder part of the autoencoder.
  - Train a classifier (e.g., logistic regression, dense network) on the extracted features.
  - Evaluate the classifier's performance on the test set and report relevant metrics.

3-7 STEP 7: EVALUATION OF EACH MODEL

- Objective: Assess and compare the performance of all implemented models.
- Tasks:
  - Calculate accuracy, precision, recall, F1-score, and other relevant metrics for each model.
  - Visualize results using plots and tables to facilitate comparison and analysis.

3-8 STEP 8: MODEL COMPARISON AND OPTIMIZATION

- Objective: Optimize models to improve performance.
- Tasks:
  - Discuss strengths and weaknesses of each model architecture.
  - Apply optimization techniques such as hyperparameter tuning, regularization, or learning rate adjustment.
  - Report optimized results and analyze improvements achieved.

3-9 STEP 9: CONCLUSION

- Objective: Summarize findings and draw conclusions from the project.
- Tasks:
  - Summarize the performance of CNNs, LSTMs, hybrid models, GRUs, vanilla LSTMs, and autoencoder-based models.
  - Discuss which model(s) performed best for the binary classification task and why.
  - Reflect on lessons learned, limitations encountered, and suggestions for future research.

Dataset and URL:

The dataset used in this project is the Diabetes 130-US hospitals for years 1999-2008 dataset from the UCI Machine Learning Repository, available at:

https://archive.ics.uci.edu/ml/datasets/Diabetes+130-US+hospitals+for+years+1999-2008

Notes:

- Provide detailed explanations and reasoning for each project step in the final report.
- Include visualizations and tables to support analysis and conclusions effectively.
- Address challenges encountered during model development and evaluation.
- Highlight the significance of model optimization in improving performance outcomes.