Detailed Step-by-Step Explanation of Projects

1. Offensive Language Detection in Social Media Comments

Technologies: Python, TensorFlow, NLTK, LSTM, BERT

Duration: Jan-Mar 2025

1. Problem Understanding:

- Goal: Detect offensive language in tweets and social media posts in real-time.
- Challenges: Social media text is noisy, informal, contains slang, emojis, and misspellings.

2. Data Collection:

- Gathered datasets containing labeled social media comments (offensive or not).
- Used publicly available datasets like the Twitter Hate Speech dataset or created a custom dataset.

3. Data Preprocessing:

- Cleaned text by removing URLs, mentions (@user), hashtags, special characters, and emojis.
- Tokenized sentences into words using NLTK.
- Removed stopwords and applied lemmatization/stemming.
- Converted text into sequences (integer encoding or embeddings).

4. Feature Extraction:

- Used word embeddings such as Word2Vec, GloVe, or BERT embeddings for semantic representation.

5. Model Selection and Training:

- Built LSTM (Long Short-Term Memory) networks to capture context and sequence in text.
- Fine-tuned pre-trained BERT model for transfer learning to boost accuracy.
- Split data into training, validation, and test sets.

6. Model Evaluation:

- Used metrics like accuracy, precision, recall, and F1-score to evaluate performance.
- Achieved over 90% accuracy.

7. Deployment:

- Designed the model to run in real-time on streaming social media data.
- Created APIs or integrated the model into a monitoring system.

2. Predictive Analysis for Loan Defaults

Technologies: NumPy, Pandas, Matplotlib, Seaborn

Duration: Mar-Apr 2024

1. Problem Understanding:

- Objective: Predict if a loan applicant is likely to default.

- Helps financial institutions reduce risk and personalize loan offers.

2. Data Collection:

- Obtained historical loan data with features like applicant income, credit score, loan amount, payment history, etc.

3. Data Cleaning and Preprocessing:

- Handled missing values by imputation or removal.

- Converted categorical variables to numerical using encoding techniques.

- Scaled numerical features using normalization or standardization.

4. Exploratory Data Analysis (EDA):

- Used Matplotlib and Seaborn to visualize data distributions, correlations, and trends.

- Identified key features that influence loan defaults.

5. Feature Engineering:

- Created new features such as debt-to-income ratio, number of previous defaults, etc.

6. Model Building:

- Used machine learning algorithms like Logistic Regression, Decision Trees, or Random Forests.

- Split data into training and test sets.

7. Model Evaluation:

- Evaluated model using accuracy, precision, recall, ROC-AUC.

- Tuned hyperparameters for better performance.

8. Reporting and Compliance:

- Generated detailed reports for stakeholders.

- Ensured model compliance with financial regulations.

3. Face and Hand Landmark Detection using CNN

Technologies: CNN, Deep Learning

1. Problem Understanding:

- Objective: Detect key points (landmarks) on face and hands in real-time video or images.

- Used in applications like gesture recognition and facial expression analysis.

2. Data Collection:

- Gathered labeled datasets with annotated facial and hand landmarks (e.g., MediaPipe, MPII datasets).

3. Data Preprocessing:

- Applied data augmentation (rotation, scaling, flipping) to increase dataset size and robustness.
- Normalized images and landmark coordinates.

4. Model Design:

- Built a Convolutional Neural Network to detect landmarks.
- Used architectures suited for localization tasks (e.g., heatmap regression).

5. Training:

- Trained CNN on augmented data.
- Used loss functions appropriate for regression (e.g., mean squared error).

6. Model Evaluation:

- Evaluated accuracy of landmark positions using distance-based metrics.
- Fine-tuned model parameters to improve performance.

7. Deployment:

- Integrated the model into real-time applications.
- Optimized for speed and accuracy.