**Sentiment Analysis on IMDB Reviews - Project Report**

**Introduction**

The objective of this project was to build a sentiment analysis model capable of classifying movie reviews as positive or negative. The dataset used for this task was the IMDB Reviews dataset, which contains a balanced set of 50,000 reviews labeled with binary sentiment (positive/negative).

**Objective**

* Preprocess raw text data for model input
* Train and evaluate a sentiment analysis model
* Understand challenges associated with natural language processing (NLP)
* Compare performance across different methods

**Dataset Overview**

* Dataset: IMDB Large Movie Review Dataset
* Total reviews: 50,000 (25,000 train, 25,000 test)
* Label: Sentiment (0 = Negative, 1 = Positive)

**Preprocessing Steps**

* Lowercased all text
* Removed punctuation, numbers, and special characters
* Tokenized reviews into words
* Removed stopwords
* Applied stemming/lemmatization
* Converted text into numerical format using techniques such as Bag-of-Words (BoW) and TF-IDF

**Model Implementation**

* Used logistic regression as a baseline classifier
* Experimented with Naive Bayes and simple neural network models
* Evaluated model performance using accuracy, precision, recall, and F1-score

**Personal Findings**

* Preprocessing played a crucial role in improving model performance
* Simple models like logistic regression performed surprisingly well on clean and balanced text data
* TF-IDF representation outperformed raw count vectors
* Neural network models captured more complex patterns but required more careful tuning and computational resources

**Problems Faced**

1. **Handling Imbalanced Words**:
   * Challenge: Frequent use of generic words reduced model’s ability to distinguish sentiment
   * Solution: Used TF-IDF to weigh words based on their uniqueness across reviews
2. **Overfitting on Neural Networks**:
   * Challenge: The model started memorizing training data
   * Solution: Added dropout layers and used early stopping during training
3. **Long Review Truncation**:
   * Challenge: Many reviews exceeded typical model input length
   * Solution: Applied fixed-length padding and truncation to ensure uniform input sizes
4. **High Dimensionality**:
   * Challenge: Large vocabulary size led to sparse and high-dimensional feature vectors
   * Solution: Limited vocabulary to top frequent words and tried dimensionality reduction techniques

**Counter Actions and Solutions**

* Conducted exploratory data analysis (EDA) to understand review length, sentiment distribution, and common words
* Applied cross-validation to ensure generalization across data splits
* Tuned hyperparameters using grid search and held-out validation sets

**Performance Summary**

| **Model** | **Accuracy** | **F1 Score** |
| --- | --- | --- |
| Logistic Regression | ~88% | ~88% |
| Naive Bayes | ~86% | ~86% |
| Simple Neural Net | ~90% | ~90% |

Actual values might vary based on specific vectorization and split settings.

**Conclusion**

The sentiment analysis project provided valuable hands-on experience with text data processing and classification. It highlighted the importance of preprocessing and representation in NLP workflows. Future steps include experimenting with word embeddings (e.g., Word2Vec, GloVe) and using deep learning models like LSTM or BERT for further improvement.