EXECUTIVE SUMMARY

Project Objective

The primary goal of this project is to create a binary classification model capable of determining the sentiment (positive or negative) of movie reviews from the IMDB dataset. With a total of 50,000 reviews, the dataset is preprocessed to focus on the top 10,000 most frequent words. This project examines how different training data sizes (100, 1,000, 3,000, 5,000, 10,000, and 20,000 samples) influence model performance. Furthermore, performance is evaluated on a constant validation set of 10,000 samples. After preprocessing the dataset, a pretrained word embedding model is used for further analysis and comparison.

Dataset Details

- **Total Reviews**: The IMDB dataset consists of 50,000 reviews.
- **Vocabulary Limitation**: To reduce complexity, only the top 10,000 most frequent words are considered for processing.
- **Data Splitting**: Training will be conducted on varying subsets of the data (100, 1,000, 3,000, 5,000, 10,000, and 20,000 reviews), with a fixed validation set containing 10,000 reviews.
- **Review Length**: Each review is truncated to a maximum of 150 words to ensure consistency in input size.

Methodology

1. Data Preprocessing

- **Subsetting Training Data**: To assess the impact of the size of training data, subsets with varying numbers of samples will be used.
- **Truncating Reviews**: Reviews are truncated at 150 words to maintain uniformity.
- **Top 10,000 Words**: A vocabulary limitation is applied, keeping only the 10,000 most frequent words for model input.

2. Word Embeddings

- **Word Representation**: In contrast to traditional one-hot encoding, word embeddings convert words into dense vectors that capture the semantic meaning of words.
- **Pretrained Embeddings**: By using pretrained word embeddings such as GloVe, which have been trained on large external corpora, the model can leverage semantic knowledge, reducing the computational cost of training embeddings from scratch.

3. Model Structure

- **Recurrent Neural Network (RNN)**: An RNN is utilized to process sequential data, as reviews are sequences of words. The model's ability to learn from past information makes it ideal for text classification.
- **Embedding Layer**: The embedding layer is responsible for converting each word in a review to its respective dense vector representation. The embeddings can either be learned from scratch during training or initialized with pretrained embeddings like GloVe.
- **Hyperparameters**: The model's performance depends on factors such as the number of RNN units and the dimensionality of the embeddings. These parameters will be tuned to achieve the best results.

4. Training and Evaluation

- **Training Data**: The model will be trained using different sizes of training data (100, 1,000, 3,000, 5,000, 10,000, 20,000 reviews).
- **Validation Set**: A constant validation set of 10,000 samples will be used to evaluate the model’s performance after each training run.
- **Performance Metrics**: Accuracy and loss will be monitored and compared across different training sample sizes and model configurations.

Model Implementation

1. Baseline Model (RNN with Learned Embeddings)

- **Model Overview**: The baseline approach trains an RNN with an embedding layer that learns word embeddings directly from the training data.

- **Evaluation Metrics**: The model's performance will be assessed based on validation accuracy, test accuracy, and loss.

2. Model with Pretrained Word Embeddings

- **Pretrained Embedding Integration**: Instead of learning embeddings from scratch, pretrained embeddings (such as GloVe) will be used to initialize the embedding layer. This allows the model to benefit from word vectors that have already captured semantic relationships between words from large external datasets.
- **Model Evaluation**: This model's performance will also be evaluated on validation and test data, with performance compared to the baseline model.

3. Impact of Training Data Size

- **Varying Training Samples**: Training data sizes will be varied (100, 1,000, 3,000, 5,000, 10,000, and 20,000 samples) to analyze how the volume of data influences model performance.
- **Performance Tracking**: For each training subset, the model's performance (accuracy and loss) will be tracked to understand how increasing the amount of training data affects the model.

4. Comparison of Embedding Layer vs. Pretrained Embeddings

- **Analysis**: The final comparison will evaluate the impact of using learned embeddings (trained from scratch) versus pretrained embeddings (like GloVe) in the context of varying training data sizes.
- **Insights**: This will reveal whether pretrained embeddings improve the model's understanding and sentiment prediction, especially when large training sets are used.

MODEL RESULTS

MODEL	ACCURACY	LOSS	VALIDATION ACCURACY	VALIDATION LOSS
One Hot model	79.0%	0.46	79.4%	0.45
Trainable Embedding Layer	79.8%	0.43	80%	0.43
Masking Padded Sequences in the Embedding Layer	79.8%	0.43	79.8%	0.43
Model with Pretrained GloVe Embeddings	79.8%	0.45	79.7%	0.45

MODEL	ACCURACY	LOSS
Embedding Layer of 100 Training Samples	75.1%	0.52
Pretrained Embedding Layer of 100 Training Samples	77.2%	0.47
Embedding Layer of 1000 Training Samples	81.0%	0.43
Pretrained Embedding Layer of 1000 Training Samples	78.9%	0.44
Embedding Layer of 3000 Training Samples	79.8%	0.47
Pretrained Embedding Layer of 3000 Training Samples	79.1%	0.44
Embedding Layer of 5000 Training Samples	79.6%	0.44
Pretrained Embedding Layer of 5000 Training Samples	79.2%	0.44
Embedding Layer of 10000 Training Samples	79.4%	0.45
Pretrained Embedding Layer of 10000 Training Samples	78.3%	0.46
Embedding Layer of 20000 Training Samples	80.6%	0.48
Pretrained Embedding Layer of 20000 Training Samples	78.4%	0.45

Conclusion:

The project evaluates the performance of various models for binary sentiment classification using the IMDB dataset. The results reveal that trainable embedding layers generally performed better than pretrained GloVe embeddings across most training data sizes. Specifically, trainable embeddings achieved higher accuracy and lower loss, especially with larger training sets. The pretrained embeddings provided competitive results, but their benefits diminished as the training data size increased.

As the dataset size grew, the performance differences between trainable embeddings and pretrained embeddings became less pronounced, suggesting that with enough data, models can effectively learn meaningful representations without relying on pretrained embeddings.

Recommendation:

- For smaller datasets (100-1,000 samples), pretrained embeddings such as GloVe are recommended, as they provide a solid foundation and improve model performance.
- For larger datasets (3,000+ samples), trainable embedding layers should be considered, as they can capture more dataset-specific nuances and achieve better performance.
- Additionally, fine-tuning the model's architecture and hyperparameters can further optimize performance, especially when working with larger datasets.