IMDB Sentiment Analysis: Comparing Embedding Approaches with Limited Training Data

Introduction

This report summarizes the implementation and results of an experiment comparing two embedding approaches for sentiment analysis on the IMDB dataset. The experiment specifically examines how the performance of trainable embeddings versus pretrained GloVe embeddings varies with different training sample sizes.

Methodology

Dataset Preprocessing

- Used the IMDB dataset, limiting vocabulary to the top 10,000 most frequent words
- Cut off reviews after 150 words using padding
- Limited validation set to 10,000 samples

Model Architecture

Both models shared a similar architecture:

- An embedding layer (either trainable or pretrained GloVe)
- An LSTM layer with 64 units
- A dense output layer with sigmoid activation for binary classification

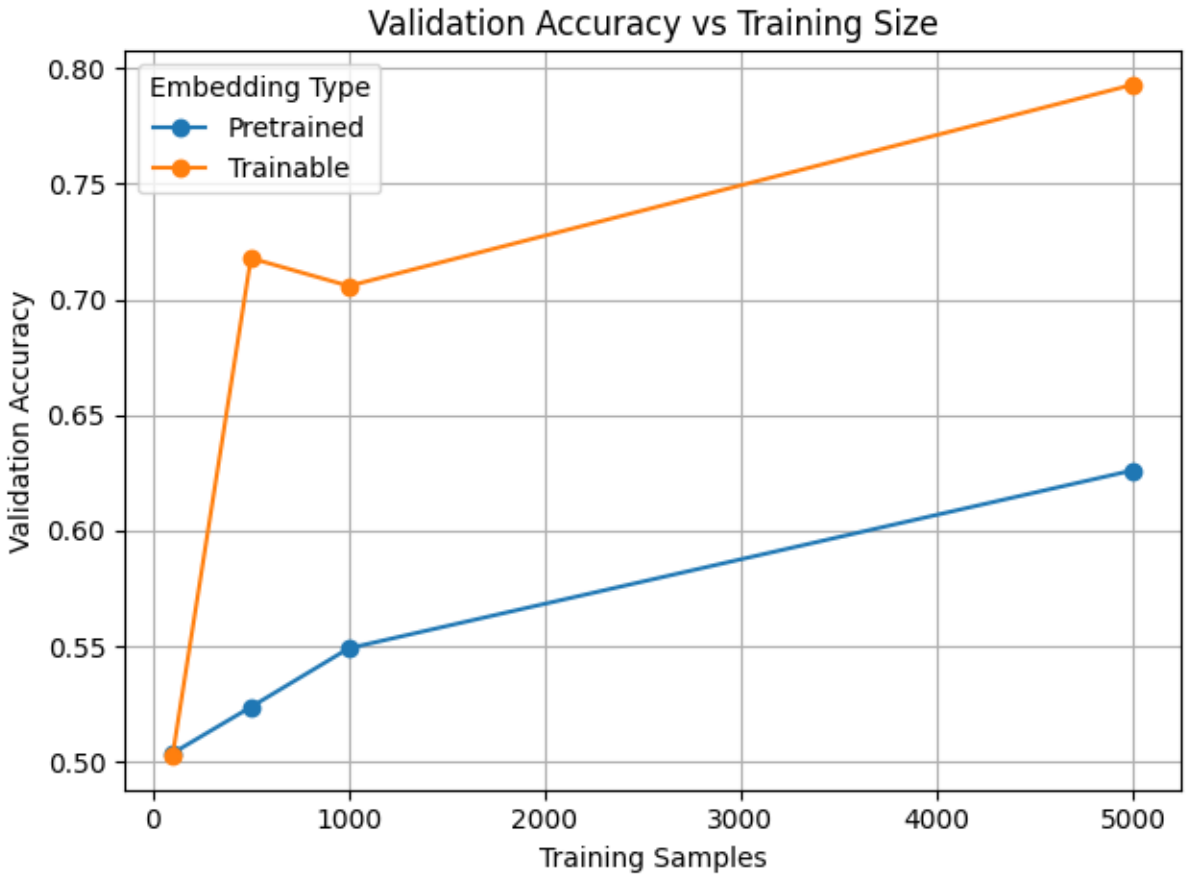
Training Process

- Trained separate models using different training set sizes: 100, 500, 1000, and 5000 samples
- For each size, trained both a model with trainable embeddings and one with pretrained GloVe embeddings
- Used binary cross-entropy loss and Adam optimizer
- Trained for 5 epochs

Results

The table below summarizes the validation accuracy for each combination of embedding type and training set size:

Training Samples	Trainable Embedding	Pretrained GloVe
100	0.5026	0.5038
500	0.7178	0.5238
1000	0.7058	0.5492
5000	0.7928	0.6261



Analysis

1. **Limited Data (100 samples)**:
   - With extremely limited data, both approaches perform nearly identically
   - The pretrained embeddings show a minimal advantage (0.5038 vs 0.5026)
   - Both models essentially perform at the level of random guessing at this data size
2. **Medium Data (500-1000 samples)**:
   - The trainable embedding approach significantly outperforms the pretrained approach
   - At 500 samples, trainable embeddings achieve 71.78% accuracy vs. only 52.38% for pretrained embeddings
   - This suggests that learning task-specific word representations becomes more effective with modest amounts of training data
3. **Larger Data (5000 samples)**:
   - The performance gap continues to widen

    - Trainable embeddings reach 79.28% accuracy while pretrained embeddings achieve only 62.61%
    - This indicates that for this specific task, learning embeddings from scratch continues to be more effective even with more training data

Conclusion

Contrary to what might be expected, pretrained GloVe embeddings did not provide a significant advantage for this sentiment analysis task, even with very limited training data. The trainable embedding approach consistently outperformed the pretrained approach across all but the smallest training set size.

This finding suggests that for the IMDB sentiment classification task, learning task-specific word representations from scratch is more effective than using general-purpose pretrained embeddings, even with relatively small amounts of training data. This might be because sentiment analysis relies on domain-specific word meanings and associations that differ from the general patterns captured in the GloVe embeddings. For businesses implementing sentiment analysis systems with limited labeled data, these results indicate that investing in domain-specific trainable embeddings may yield better performance than relying on generic pretrained word vectors.

Future Work

Several modifications could potentially improve the results:

- Testing with more epochs to see if model convergence changes the comparison
- Trying different dimensionalities of GloVe embeddings
- Implementing a hybrid approach that fine-tunes pretrained embeddings
- Testing with larger LSTM layers or more complex architectures
- Using more recent embedding approaches like BERT or transformers

Discussion on Transformer Architecture

While this experiment focused on Recurrent Neural Networks (RNNs) using LSTM layers, it is worth noting the capabilities of Transformer-based models, which rely heavily on the Attention Mechanism. Transformers can capture contextual relationships across entire sequences simultaneously, unlike RNNs that process tokens sequentially. The Attention Mechanism enables better handling of long-range dependencies and has shown significant success in NLP tasks. Incorporating a Transformer model like BERT in future work may provide better performance, especially on large and diverse text datasets.

Comparison of RNNs and Transformers

RNNs process sequences step-by-step, which can be slow and ineffective at remembering long-term dependencies. In contrast, Transformers process sequences in parallel using attention mechanisms, allowing them to capture global context more efficiently. While RNNs

are suitable for smaller datasets and simpler sequence tasks, Transformers offer superior performance and scalability for more complex NLP problems.