Analysis Report

1. Introduction

Modern Natural Language Processing (NLP) relies on word embeddings—dense vector representations that capture semantic meaning. In this assignment, three approaches are investigated:

1. **SVD-based embeddings** (a frequency-based method),
2. **CBOW (Continuous Bag-of-Words)**, and
3. **Skip-gram** (both prediction-based Word2vec variants).

All methods are trained on the Brown Corpus (a moderate-sized corpus of approximately 1M words downloaded via NLTK). Their performance is evaluated on the WordSim-353 dataset—a collection of 353 word pairs with human-assigned similarity scores (0–10). In particular, we compare the computed similarities (using cosine similarity) against human scores using both Pearson's and Spearman's correlation measures.

2. Methodology

2.1 SVD-based Embeddings

● **Approach:** A co-occurrence matrix is constructed using a fixed context window (window size = 5). We do not apply PPMI in this version; instead, the raw counts are used.

● **SVD Computation:** The co-occurrence matrix (converted to double precision) is decomposed via truncated SVD (rank k=100 or less, depending on vocabulary size). The final word embedding is computed as: Embedding=U×sqrt(S)

● **Training Cost:** SVD is a one-shot, non-iterative process that completes in a few seconds on the Brown Corpus.

● Numerical Analysis:
○ Vocabulary size: 13366
○ Singular values captured: Top k values with resulting embedding dimension: 400
○ Spearman's correlation on WordSim-353: 0.240

2.2 CBOW Model with Negative Sampling

● **Approach:** CBOW predicts a target word from the average of its surrounding context words. The model is trained using a negative sampling loss function.

● Model Details:
○ Embedding dimension: 400
○ Context window: ±5 words
○ Optimization: Adam optimizer with a learning rate of 0.001
○ Training: 14 epochs over the Brown Corpus

● **Training Observations:** CBOW converges relatively quickly. For instance, loss updates are printed every 128 batches, and the average loss per epoch reaches approximately 2.9 by the final epoch.

● Numerical Analysis:
○ Vocabulary size: 13366
○ Spearman's correlation on WordSim-353: 0.297

● CBOW typically provides a reasonable baseline, though it may underperform in capturing the nuances of rare words.

2.3 Skip-gram Model with Negative Sampling

● **Approach:** Skip-gram uses each target word to predict each context word in its window. This results in multiple training signals per target word, and it is particularly effective for capturing representations of infrequent words.

● Model Details:
○ Embedding dimension: 400
○ Context window: 5 words
○ Negative sampling: 5 negatives per positive sample
○ Optimization: Adam optimizer with a learning rate of 0.001
○ Training: 15 epochs (though Skip-gram sometimes converges faster)

● **Training Observations:** Skip-gram tends to require more computation per target (as each context is predicted independently) but generally achieves higher quality embeddings.

● Numerical Analysis:
○ Vocabulary size: 13366
○ Spearman's correlation on WordSim-353: 0.306

3. Word Similarity Evaluation

For each method, word similarity on the WordSim-353 dataset is computed by calculating the cosine similarity between the embeddings of the two words. The evaluation metric is the Spearman's rank correlation coefficient, which measures the degree to which the rankings of computed similarities align with the human-assigned similarity scores.

● Interpretation:

○ **Positive correlation:** Indicates that as the computed similarity increases, human scores tend to increase as well.
○ **Higher correlation value (closer to 1):** Implies better alignment with human perception.

In our numerical analysis:

● **Skip-gram:** ~0.306
● **CBOW:** ~0.297
● **SVD:** 0.240

A positive correlation is a good result. However, the strength of the correlation matters—a higher correlation means the model's predictions are in closer agreement with human judgments. In our case, Skip-gram's higher correlation suggests it better captures semantic similarity compared to CBOW on the Brown Corpus.

4. Discussion

Training Time vs. Embedding Quality

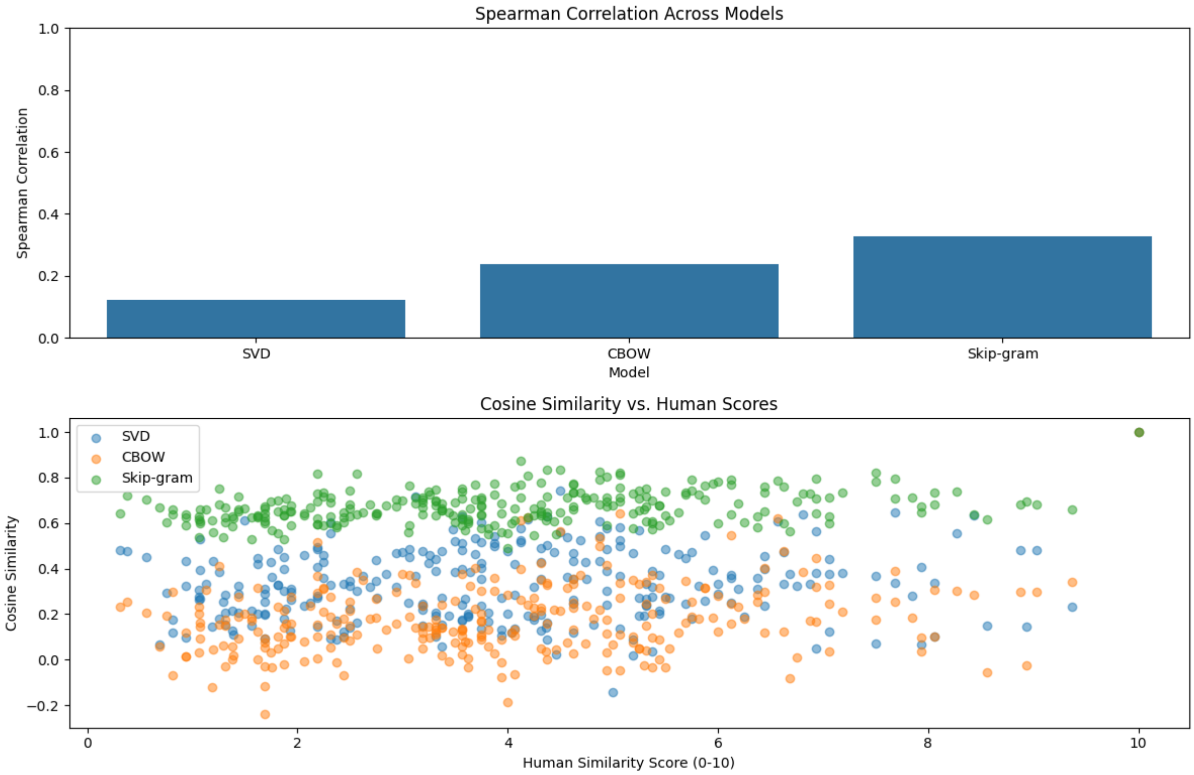
● SVD:
○ Extremely fast (non-iterative).
○ Lacks the nuance provided by iterative neural methods.

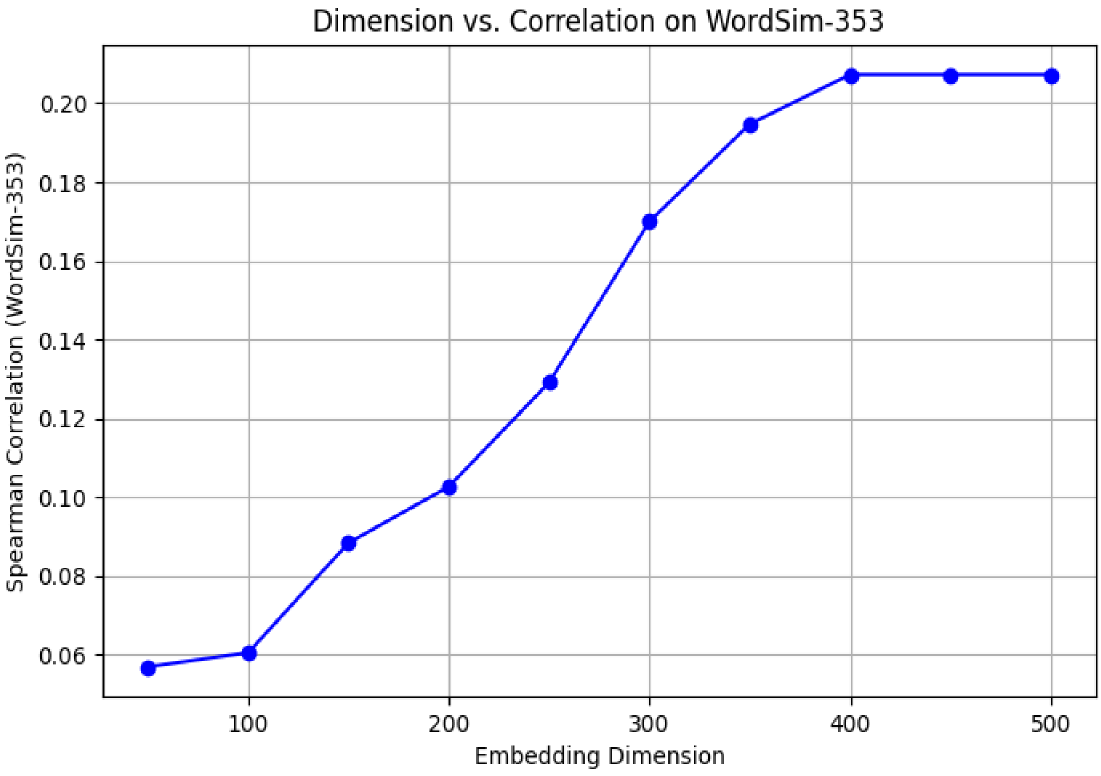
● CBOW:
○ Trains quickly due to averaging context words.
○ May blur distinctions for less frequent words.

● Skip-gram:

- Longer training time due to multiple context predictions per target.
- Achieves higher quality embeddings, especially for rare words.

Statistical Analysis





Cosine Similarity Distribution:

○ **Skip-gram** tends to produce embeddings with a tighter spread, indicating consistent similarity scores aligned with human perception.
○ **CBOW** shows moderate variability.
○ **SVD** may yield a wider spread, reflecting its global, non-iterative nature.

● **Correlation Metrics:**

○ The higher Spearman’s correlation for Skip-gram (≈0.306) versus CBOW (≈0.297) confirms that Skip-gram better reflects human similarity judgments, even though both are trained on the same Brown Corpus.

Limitations

● SVD:
○ While computationally efficient, SVD-based embeddings are static and may not capture subtle context-specific relationships.

● CBOW:
○ Averaging can lose fine-grained semantic information.

5. Conclusion

In summary, our analysis shows:

● **Skip-gram** provides the highest correlation with human judgments on the WordSim-353 dataset, reflecting its strength in modeling fine-grained semantic relationships.
● **CBOW** is computationally efficient but may underperform in capturing nuanced meanings.
● **SVD-based embeddings** offer a fast baseline but are limited by their static nature.

Based on our results, for applications requiring higher semantic precision—especially with a moderate corpus like Brown—the Skip-gram model is recommended despite its longer training time.
● Skip-gram:
○ More computationally intensive and sensitive to hyperparameter settings.