Co-occurrence Matrix Word Embeddings

Deepak Singh
22125

August 29, 2025

1 Introduction

Word embeddings are numerical vector representations of words that capture semantic meaning. In this project, we implement **count-based word embeddings** using a **co-occurrence matrix**, apply **dimensionality reduction**, and visualize the embeddings in 2D space. Unlike prediction-based models such as Word2Vec, this approach relies on the distributional hypothesis: *"You shall know a word by the company it keeps."*
The complete source code and implementation are available on GitHub:
https://github.com/DeepakSingh/Co-occurenceMatrix

2 Methodology

2.1 Input Corpus

I used the following small text corpus on artificial intelligence and machine learning:

> Artificial intelligence and machine learning are transforming the world.
> Machine learning helps computers learn patterns from data.
> Natural language processing enables computers to understand human language.

2.2 Preprocessing the Corpus

The raw text corpus is cleaned by converting it to lowercase, removing punctuation, and splitting into tokens (words).

2.3 Distinct Words

The vocabulary is extracted as the set of unique tokens. Each word is assigned an integer ID for indexing.

2.4 Co-occurrence Matrix

We construct a $|V| \times |V|$ co-occurrence matrix, where each entry (i, j) represents how often word j occurs within a window size n around word i. For this experiment, the default window size was set to $n = 4$.

2.5 Dimensionality Reduction

The co-occurrence matrix is typically large and sparse. To obtain compact embeddings, we applied **Singular Value Decomposition (SVD)** and projected the high-dimensional vectors into a lower-dimensional space ($k = 2$).

2.6 Visualization

We plotted the reduced embeddings using Matplotlib. Each point in 2D corresponds to a word, and nearby words tend to have related meanings due to similar co-occurrence contexts.

3 Results

The resulting plots showed clustering of semantically related terms. For example, *"machine"*, *"learning"*, *"data"* appeared close to each other, while *"language"*, *"processing"*, *"human"* formed a separate cluster. This demonstrates how co-occurrence captures contextual similarity.

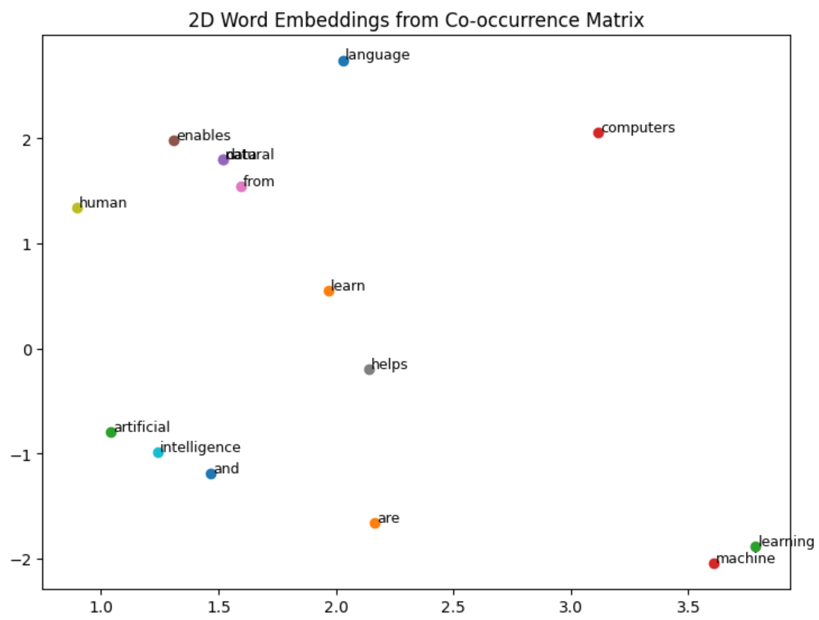

Figure 1: 2D visualization of word embeddings from the co-occurrence matrix

4 Discussion

4.1 Advantages

- Simple to implement and interpret.
- Captures global co-occurrence statistics.

4.2 Limitations

- High-dimensional and sparse.
- Does not capture word order or deeper semantics compared to Word2Vec or GloVe.

4.3 Possible Improvements

- Use larger corpora (news datasets, Wikipedia).
- Apply weighting schemes such as **Positive Pointwise Mutual Information (PPMI)**.
- Use advanced visualization methods (t-SNE, UMAP).

5 Conclusion

This project demonstrated the construction of word embeddings using a count-based co-occurrence matrix, dimensionality reduction, and visualization. Even with a small corpus, semantically related words clustered together, validating the effectiveness of the distributional hypothesis.