DSN

Large Language model

Word Embeddings

- Understanding word embeddings
- Word2Vec
- Lab: Implementing word embeddings using Gensim

Word Embedding is a way of representing words as numbers, in a continuous vector space.

Instead of treating words as isolated units (like just "cat", "dog", "run"), embeddings give each word a mathematical representation that captures meaning and context.

dense vector: most values are non zeros eg [0.2, 0.6, 0.3]

spare vector: mostly filed with zeros eg [0, 0, 1, 0, 0]

Word embeddings are dense vectors of real numbers.

Where do Dense Vector comes from

- Word2Vec, GloVe
- Embedding Layers in deep learning model (PyTorch embeddings)
- Transformers like BERT and GPT

How Are Vectors Learned?

The idea is simple but brilliant:

- Initialize word vectors randomly.
- Use a shallow neural network (just 1 hidden layer).
- Train it using CBOW or Skip-Gram on a large corpus.
- The model learns to predict well by adjusting the word vectors.
- Once training is done, the weights of the hidden layer are the word embeddings.

Implementing word embeddings using Gensim

Neural Networks

- Introduction to neural networks
- Basic architectures: Feedforward, RNNs
- Sequence modeling: Long Short-Term Memory (LSTM), GRU
- Lab: Sentiment analysis using RNN
- Paper Reading: Sequence to Sequence Learning with Neural Network

Introduction

Neural network is a computational model composed of layers of interconnected nodes (neurons), designed to learn patterns from data by approximating complex functions using a combination of linear transformations and non-linear activations.

Feed forward neural network

Input Layer → Hidden Layer(s) → Output Layer

data flows in one direction only, from input to output

no loops, no cycle, no feedback connections

show simple feed forward NN

Recurrent Neural Network

These are neural network designed to process sequential data, like text or time series, by maintaining a "memory" of past inputs to influence current outputs

"Memory" or Hidden State:

RNNs have a hidden state that captures information from previous inputs, allowing them to learn from the context of the sequence.

Recurrent Connections:

The network's output at one time step is fed back as input to the next time step, creating a recurrent connection.

Types of RNN:

1. Vanilla RNN
2. Gated Recurrent Units (GRU)
3. Long Short-Term Memory (LSTM)

Sentiment Analysis using RNN

Paper Review and Implementation:

Sequence to Sequence Learning with Neural Networks

Thank you