Natural Language Processing
NLP_CLT_1st_May_11th_2025

Eng. Maytham Ghanoum

Artificial Intelligence & Deep Learning Specialist
MTN Syria – SCS – SVU CLT
+963947222064 - +963982018359
https://www.linkedin.com/in/maytham-ghanoum-697aa5207/
https://www.facebook.com/maytham.ghanoum

Word Embeddings

Word embeddings are a type of word representation that allows words with similar meaning to have a similar representation. They are a way of representing words as vectors in a continuous vector space, where semantically similar words are mapped to nearby points. Unlike traditional methods like Bag-of-Words (BoW) or TF-IDF, word embeddings capture the context of a word in a document, its semantic and syntactic relationships, and its multi-dimensional space.

Word Embeddings

Key Concepts of Word Embeddings

Continuous Vector Space: Words are represented as dense vectors of real numbers, typically in the range of 50 to 300 dimensions.

Semantic Relationships: Words that are used in similar contexts tend to have similar vectors.

Contextual Understanding: Embeddings capture the context in which words appear, enabling better performance in tasks like analogy reasoning, semantic similarity, and syntactic parsing.

Word Embeddings

Algorithms for Word Embeddings:

Word2Vec

Developed by Google, Word2Vec is one of the most well-known algorithms for creating word embeddings. It uses neural networks to learn word associations from a large corpus of text. Word2Vec comes in two flavors:

Continuous Bag of Words (CBOW): Predicts a word given its context. For example, given the context "The cat sat on the ___", it predicts the word "mat".

Skip-gram: Predicts the context given a word. For example, given the word "sat", it predicts the surrounding words "The", "cat", "on", "the".

Word Embeddings

Algorithms for Word Embeddings:
Word2Vec

Advantages of Word2Vec:

Efficient to train on large datasets.
Captures linear relationships between words (e.g., King - Man + Woman ≈ Queen).

Word Embeddings

GloVe (Global Vectors for Word Representation):
Developed by Stanford, GloVe is another popular word embedding method. Unlike Word2Vec, which relies on local context windows, GloVe is based on global word-word co-occurrence statistics from a corpus. It constructs a co-occurrence matrix and then applies matrix factorization to produce word vectors.

Advantages of GloVe:

- Captures both local context and global statistical information.
- Pre-trained models are available, trained on massive datasets like Common Crawl and Wikipedia.

Word Embeddings

GloVe (Global Vectors for Word Representation):

1. **Co-occurrence Matrix**: GloVe constructs a large matrix of word co-occurrence statistics from a given corpus. Each entry X_{ij} in this matrix represents the frequency with which word i appears in the context of word j.

2. **Global Context**: Unlike some other embedding methods that focus on local context (e.g., word2vec's skip-gram and CBOW models), GloVe captures global statistical information by considering the entire corpus.

3. **Weighted Least Squares Objective**: GloVe aims to learn word vectors such that their dot product equals the logarithm of the words' probability of co-occurrence. The objective function is designed to minimize the difference between the dot product of two word vectors and the logarithm of their co-occurrence probability, weighted by the co-occurrence frequency.

Word Embeddings

GloVe (Global Vectors for Word Representation):

The GloVe Model

The objective function for GloVe can be expressed as follows:

$$J = \sum_{i,j=1}^{V} f(X_{ij})(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log(X_{ij}))^2$$

Here:

- V is the vocabulary size.
- w_i and $\tilde{w}_j$ are the word vectors for the main word and the context word, respectively.
- b_i and $\tilde{b}_j$ are bias terms.
- X_{ij} is the co-occurrence count of words i and j.

Word Embeddings

GloVe (Global Vectors for Word Representation):

- $f(X_{ij})$ is a weighting function that gives less importance to extremely common co-occurrences, usually defined as:

$$f(X_{ij}) = \begin{cases} \left(\frac{X_{ij}}{X_{\max}}\right)^{\alpha} & \text{if } X_{ij} < X_{\max} \\ 1 & \text{otherwise} \end{cases}$$

where $X_{\max}$ and α are hyperparameters (typically, $\alpha \approx 0.75$ and $X_{\max}$ around 100).

Word Embeddings

GloVe (Global Vectors for Word Representation):

Advantages of GloVe

1. **Efficiency**: By focusing on word co-occurrences, GloVe is able to leverage the global statistical information of the corpus more effectively.

2. **Performance**: GloVe often provides better word representations for tasks involving semantic similarity and relatedness compared to models that only consider local context.

3. **Interpretability**: The embeddings learned by GloVe capture meaningful linear substructures. For example, the vector difference between the words "king" and "queen" is similar to that between "man" and "woman."

Word Embeddings

GloVe (Global Vectors for Word Representation):

Advantages of GloVe

1. **Efficiency**: By focusing on word co-occurrences, GloVe is able to leverage the global statistical information of the corpus more effectively.

2. **Performance**: GloVe often provides better word representations for tasks involving semantic similarity and relatedness compared to models that only consider local context.

3. **Interpretability**: The embeddings learned by GloVe capture meaningful linear substructures. For example, the vector difference between the words "king" and "queen" is similar to that between "man" and "woman."

Word Embeddings

GloVe (Global Vectors for Word Representation):

Implementation Steps

1. **Build the Co-occurrence Matrix**: Scan through the corpus and construct a co-occurrence matrix, where each element represents how often pairs of words appear together within a given context window.

2. **Compute the Embeddings**: Optimize the objective function to learn word vectors by iterating over the non-zero entries of the co-occurrence matrix.

3. **Usage**: The resulting word vectors can be used in various NLP tasks such as text classification, sentiment analysis, and machine translation.

Word Embeddings

GloVe (Global Vectors for Word Representation):

Applications

GloVe embeddings have been widely used in various NLP applications, including:

- **Semantic Analysis**: Understanding relationships between words and phrases.
- **Information Retrieval**: Enhancing search engines by improving query understanding.
- **Text Classification**: Improving the performance of models for sentiment analysis, topic detection, etc.
- **Machine Translation**: Assisting in translating text by providing robust word representations.

Word Embeddings

Benefits of Word Embeddings

Improved Performance: Word embeddings often lead to better performance in NLP tasks such as text classification, sentiment analysis, and machine translation.

Semantic Understanding: They enable models to understand the semantic meaning of words and phrases, leading to more natural and effective language understanding.

Dimensionality Reduction: They reduce the dimensionality of the text data while preserving the semantic relationships between words

Word Embeddings

Use Cases of Word Embeddings

Sentiment Analysis: Understanding the sentiment of a piece of text based on the context of words.

Machine Translation: Translating text from one language to another while maintaining context and meaning.

Document Similarity: Finding similar documents by comparing the embeddings of their constituent words.

Word Embeddings

Word embeddings are a powerful tool in NLP that transform words into continuous vector spaces, capturing semantic relationships and contextual meanings. Algorithms like Word2Vec and GloVe are widely used to generate these embeddings, providing significant improvements over traditional text representation methods. By leveraging these embeddings, NLP systems can achieve better performance in a variety of tasks, from sentiment analysis to machine translation.