Understanding Word Embeddings (Word2vec, GloVe, FastText)

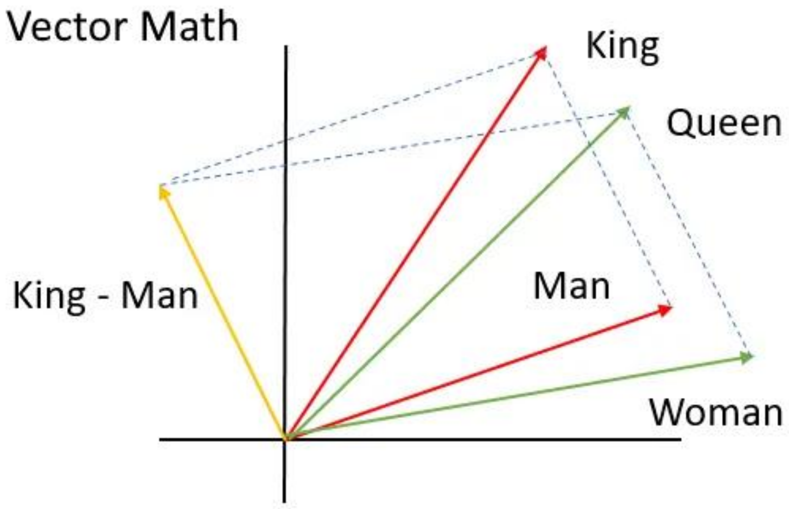
Author: Edward Leen

Student ID: 23090780

Github: Github Repository

1) What is NLP?

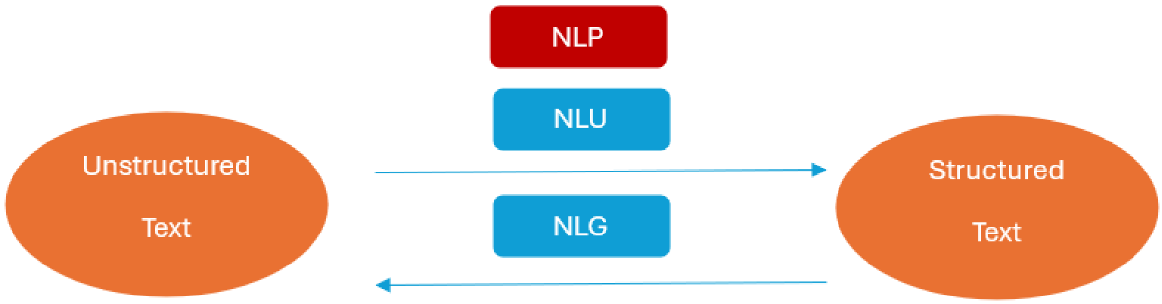
Before we jump into Word Embeddings, we need to first understand what NLP is. NLP stands for Natural Language Processing. NLP is a subfield of computer science and AI used for building machines that are able to manipulate human language or, data that resembles human language in the way that it is written, spoken, and organized.



Alt Text Figure 1: "Visualization of word vector relationships showing the analogy 'King - Man + Woman ≈ Queen,' demonstrating how vector arithmetic captures semantic relationships in word embeddings."

2) Logic behind NLP

NLP has high utility value for AI applications. It starts with unstructured text and ends with structured text, NLP sits between unstructured and structured text. When text gets translated from unstructured to structured it is called **NLU (Natural Language Understanding)** and when it gets translated from structured to unstructured it is called **NLG (Natural Language Generation)**. Something to note is that, NLP is not just one particular algorithm, but more like a bag of tools where a specific tool will be best suited for a specific task. The chart below represents the flow of the NLP model.



Alt Text Figure 2: "Flowchart of an NLP model transitioning from unstructured text to structured text, with red and blue rectangles representing stages in the process.

3) Use cases of NLP

Machine translation is a scenario which helps in the understanding contexts in sentences. For example, if a sentence like “The spirit is willing but the flesh is weak” is translated from English to Russian and then back from Russian to English the sentence might come out as “The Vodka is good but the meat is rotten”, which is not the intended meaning of the phrase. In this case, NLP can be helpful in understanding contexts of sentences.

Virtual Assistant and Chatbot are places where NLP can be quite useful. A virtual assistant is something like SIRI or Alexa that takes human utterances and then deriving a command to be executed based upon that. A chatbot is similar but in written language that is used to traverse a decision tree in order to take an action.

Sentiment Analysis, a tool in NLP, used to assess the sentiment behind a product review or an email. For example, it can be used to check whether a product review is positive or negative sentiment.

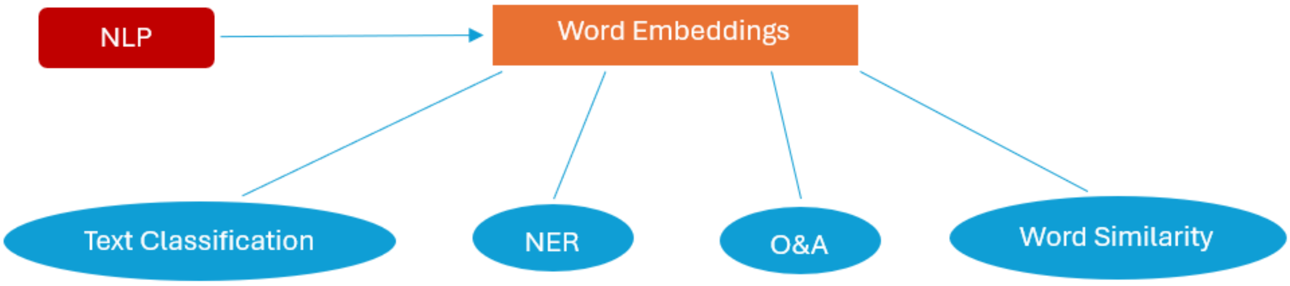
Spam Detection, yet another useful scenario for assessing, if an email is spam or not. Here, NLP will check for grammatical errors, over used words or if there is a claim of urgency to the email, these factors are some criteria’s for assessing real and spam emails.

What are Word Embeddings?

They represent words as numbers, specifically as numeric vectors, in a way that captures the semantic relationship and contextual information. This means that words with similar meaning are positioned close to each other, and the distance and direction between vectors encodes the degree of similarity between words.

Why do we need to transform words to numbers?

Most machine learning algorithms are incapable of processing plain text in its raw form. These require input as numbers to perform any tasks, and that’s where embeddings come in. Word embeddings are used in various NLP tasks.

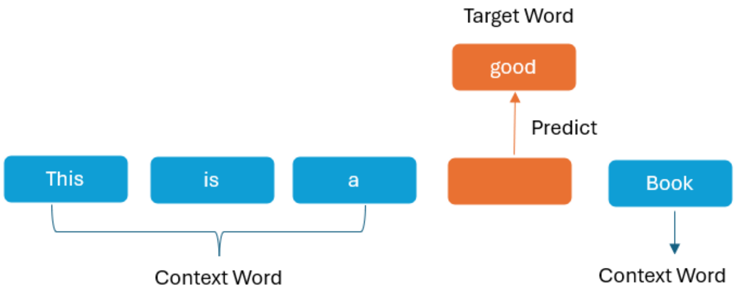


Alt Text Figure 3: "A conceptual diagram illustrating the role of word embeddings in NLP, showing how words are transformed into numerical vectors to capture semantic relationships."

Word Embedding Methods and How it Works?

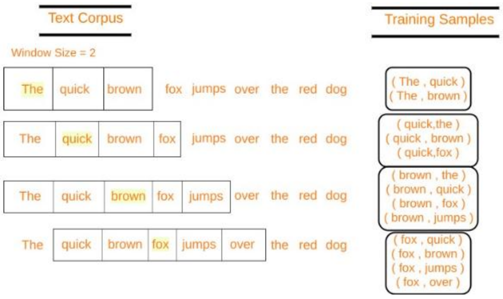
- **Frequency – based**:- Word representation that are derived from the frequency of words in a corpus. It is based on the idea that, the importance of a word depends on how frequently a word is used in the text. One type of frequency based embedding is:
  - **TF-IDF (Term Frequency – Inverse Document Frequency)**: This type of embedding highlights words that are frequent within a specific document but are rare across the entire corpus.
    Eg: For a document related to coffee TF-IDF would emphasise words like espresso and cappuccino, which might appear in that document but rarely in other documents about different topics. Commonly used words like “the” and “for” would receive a low TF-IDF score.

- **Prediction – based**:- These capture semantic relationships and contextual relationships between words. A prediction based embedding excels at separating words with close meanings and can manage various senses in which a word may be used.
  Eg: “A dog is barking or wagging its tail” closely relates dog with words like bark, wag and tail.
  - **Word2Vec**: This foundational technique developed by Google, is a type of embedding similar words occur more frequently together than dissimilar words. That does not mean that just because a word occurs within the vicinity of another word it has similar meaning, we also have to consider the frequency of which the words are found together. These consist of two main architectures:
    - **CBOW (Continuous Bag Of Words) –** This architecture predicts the target word based on the surrounding context words. The example below will help you understand it a bit better.



Alt Text Figure 4:"Diagram illustrating the CBOW (Continuous Bag of Words) architecture, where context words are used to predict a target word."

    - **SKIP-GRAM –** This does the opposite. It predicts the context words based on the given target word. Considering an array of words W, if W(i) is the input word, then W(i-2), W(i-1), W(i+1), and W(i+2) are the context words if the sliding window size is 2.



Alt Text Figure 5: "Diagram showing the Skip-Gram architecture, where a target word is used to predict surrounding context words within a sliding window." Source

  - **GloVe (Global Vectors for Word Representation)**: This technique of Word Embedding makes use of co-occurrence statistics to create word vectors. This takes a broad view approach by analysing how often words appear together across the entire corpus, then uses this information to create word vectors.
- **Subword Embeddings**:- This method represents words in NLP by breaking them into smaller units, such as character sequences or morphemes. Unlike traditional word embeddings that assign a single vector per word, subword methods decompose words into parts, making it easier to handle rare words, misspellings, and morphological variations. For instance, “unhappiness” could be split into “un-”, “happy”, and “-ness”. The technique FastText leverages this approach to enhance to enhance generalisation across languages and vocabularies.
  - **Fast-Text**: This technique was developed by Facebook AI research. It is an extension of Word2Vec and represents words as a collection of n-grams. This method is especially effective for managing unique words and recognising morphological variations.

Comparison of Techniques

Feature	Word2Vec	GloVe	FastText
Architecture	Predictive (neural net)	Count-based (matrix factorization)	Predictive with subwords
Training Speed	Fast	Slower (needs co-occurrence matrix)	Fast
Memory Usage	Moderate	High (stores co-occurrence matrix)	Moderate
Handles OOV (Out-of-Vocabulary)	No	No	Yes (via subwords)
Word Similarity	Good	Excellent	Good
Morphology	Limited	Limited	Excellent

When to Use Each

1. **Word2Vec**: Good general-purpose embeddings, especially with limited data
2. **GloVe**: When you have large corpora and want to capture global statistics
3. **FastText**: For morphologically rich languages or when dealing with misspellings/OOV words

Conclusion

Word embeddings are a cornerstone of modern NLP, transforming words into numerical vectors to capture semantic and contextual relationships. Techniques like Word2Vec, GloVe, and FastText each offer unique strengths—Word2Vec excels in predictive tasks, GloVe leverages global co-occurrence statistics, and FastText handles morphological variations through subword embeddings. The choice of method depends on specific needs, such as training speed, memory usage, or handling out-of-vocabulary words. As NLP continues to evolve, these embeddings remain essential for applications like machine translation, sentiment analysis, and virtual assistants, bridging the gap between human language and machine understanding.

References

Date: 25/03/2025

What is NLP (Natural Language Processing)? - IBM Technologies

What are Word Embeddings? - IBM Technologies

A Complete Overview of Word Embeddings - AssemblyAI

What Is NLP (Natural Language Processing)? | IBM - IBM

Word Embeddings in NLP | Word2Vec | GloVe | fastText - Medium

Advanced Word Embeddings: Word2Vec, GloVe, and FastText - Medium

What are word embeddings? - IBM

Implement your own word2vec(skip-gram) model in Python - GeeksforGeeks

What are subword embeddings? – Milvus

Word Embedding A Powerful Tool — How To Use Word2Vec GloVe, FastText - Spotintelligence