Natural Language Processing (NLP) is a multidisciplinary field that focuses on enabling machines to understand, interpret, and generate human language in a way that is both meaningful and contextually relevant. It combines principles from linguistics, computer science, artificial intelligence, and cognitive psychology to bridge the gap between human communication and machine understanding.

To process language, Natural Language Processing (NLP) converts human text into numerical representations called vectors, using techniques like Bag-of-Words (BoW) for frequency counts, TF-IDF to weigh word importance, and Word Embeddings (e.g., Word2Vec)and Transformer-based models (e.g., BERT, GPT) to capture semantic meaning and context through dense vectors.

Machines do not understand raw text like humans because text is a discrete, symbolic input that must be converted into a numerical format for processing. Humans comprehend language through context and real-world experience, understanding the nuances and meanings of words, while computers interpret text as vectors of numbers and struggle with common sense, emotional context, and the vast ambiguity of human language.

BAG OF WORDS

In**Bag-of-Words (BoW)**, text is represented by treating each word as a feature in a vector. The model completely ignores word order, syntax, and semantics. Each word in the corpus corresponds to a dimension in the vector, and the value represents how many times the word appears in the document.

The bag-of-words model is commonly used in methods of document classification where the (frequency of) occurrence of each word is used as a feature for training a classifier.

The bag-of-words model is simple to understand and implement and has seen great success in problems such as language modeling and document classification.

**tf–idf** or **TFIDF**,

short for **term frequency-inverse document frequency**, is a numerical statistic that is intended to reflect how important a word is to a document in document.The tf–idf value increases proportionally to the number of times a word appears in the document and is offset by the number of documents in the corpus that contain the word, which helps to adjust for the fact that some words appear more frequently in general. *Count the number of times each word appears in a document. Calculate the frequency that each word appears in a document out of all the words in the document.*

TF-IDF also doesn’t take the semantic meaning of the words.

**Word2Vec :**

Word2Vec model is used for learning vector representations of words called “word embeddings”. This is typically done as a preprocessing step, after which the learned vectors are fed into a discriminative to generate predictions and perform all sorts of interesting things. It takes the semantic meaning of words

With Word2Vec, a word is transformed into a numerical dense vector (not sparse vector as retrieved in BoW and TF-IDF) with d-dimensions typically 50, 100, 200, 300, etc.  
If two words are semantically similar, then the vectors of these words are closer geometrically.  
For example, if we have 3 words: tasty, delicious, and baseball. After Word2Vec transformation, the Vectors for the words viz., “tasty” and “delicious” will be close in the d-dimension space, and the Vector for the word “baseball” will be farther.

**Doc2Vec:**

Doc2Vec stands for Document to Vector.

Instead of creating a vector for each word and then taking the average to find the vector representation of text, this technique creates a vector for each document or collection of texts.

The goal is the same as Word2Vec.

Just as the Word2Vec pre-trained Models, we do have pre-trained Doc2Vec models trained on a very large corpus of texts like Wikipedia or Google News.

This is a much easier process to use for Machine Learning than Word2Vec.