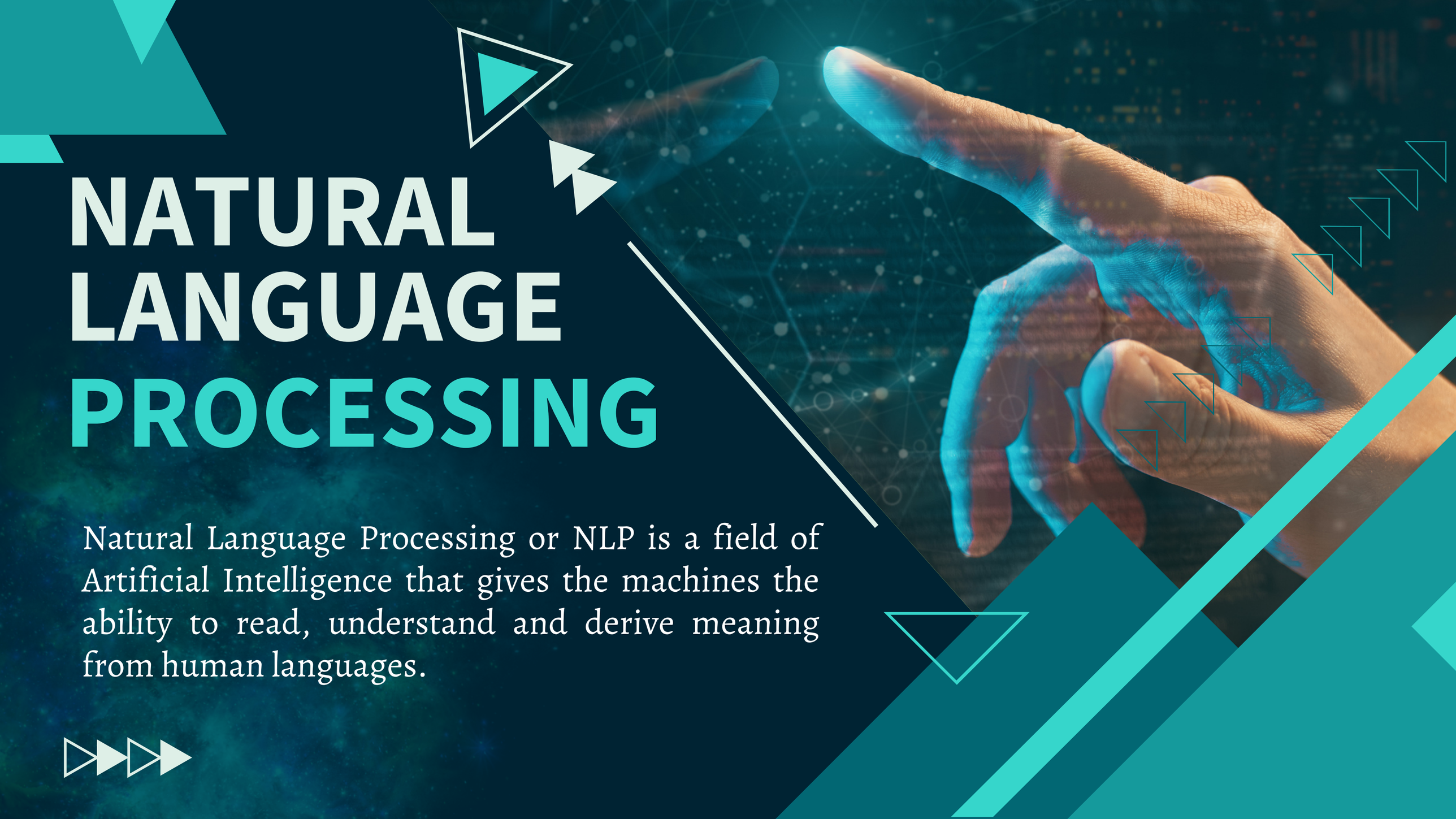
NATURAL LANGUAGE PROCESSING



Natural Language Processing or NLP is a field of Artificial Intelligence that gives the machines the ability to read, understand and derive meaning from human languages.

APPLICATIONS OF NLP

Sentiment Analysis

It basically means to analyze and find the emotion or intent behind a piece of text or speech or any mode of communication.

Speech Recognition

Speech Recognition is a technology that enables the computer to convert voice input data to machine readable format.

E-Mail Filtering

NLP makes use of a technique called text classification to filter emails. It refers to the process of classification of a piece of text into predefined categories.

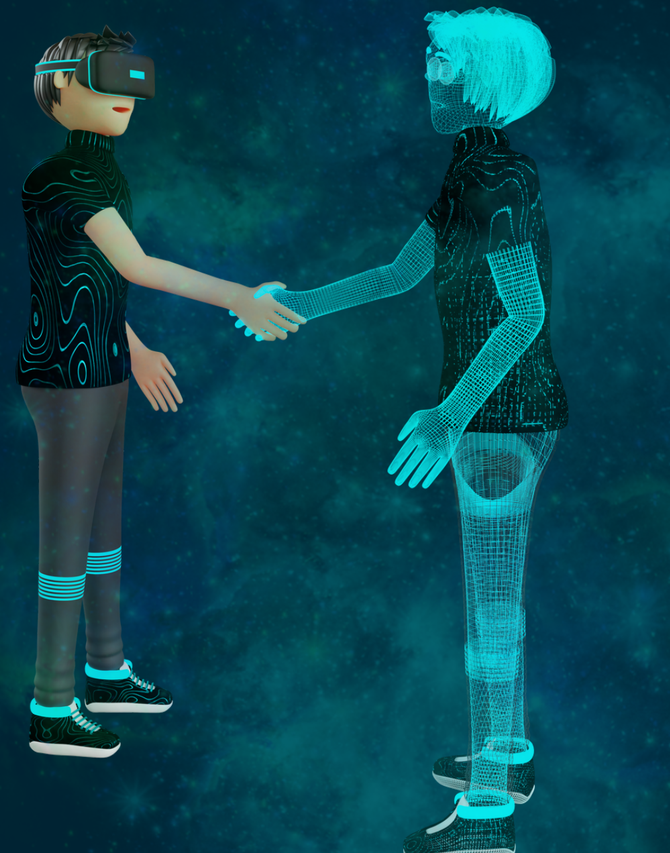
Advertisement To Target Audience

Through NLP, keywords that are searched by the user are matched with the keywords of the product ad. If they are similar, the user gets an advertisement. This process is called keyword matching

OpenAI's GPT-3

Generative Pre-trained Transformer 3 (GPT-3; stylized GPT·3) is an autoregressive language model that uses deep learning to produce human-like text. GPT-3's full version has a capacity of 175 billion machine learning parameters

Two AIs Have An Existential Crisis (GPT-3)



CODING WITH GPT 3

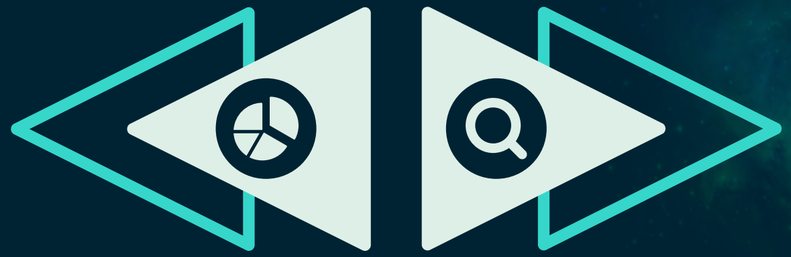

```
def fib(n):
    if n <= 1:
        return 1
    return fib(n-1) + fib(n-2)

fib(10)
```

Basic Terminologies in NLP

Stemming:

Stemming is used to normalize words into its base form or root form.The big problem with stemming is that sometimes it produces the root word which may not have any meaning.

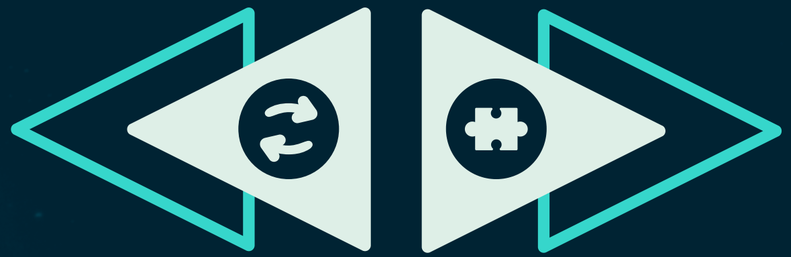


Corpus:

A Corpus is defined as a collection of text documents for example a data set containing news is a corpus or the tweets containing Twitter data is a corpus

Lemmatization:

It is used to group different inflected forms of the word, called Lemma. The main difference between Stemming and lemmatization is that it produces the root word, which has a meaning.

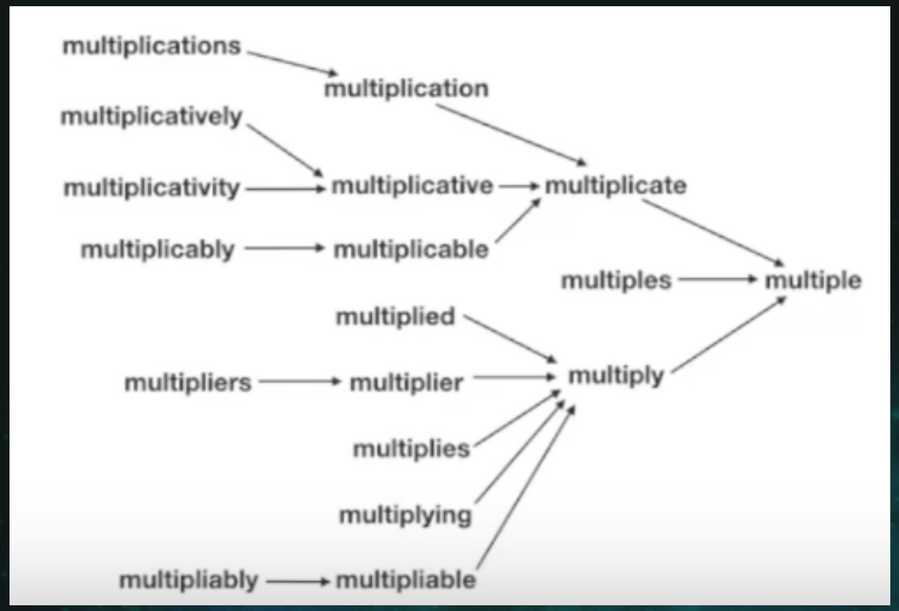


Tokenization:

Tokenization is a process of splitting a text object into smaller units which are also called tokens

Corpus > Documents > Paragraphs > Sentences > Tokens

A detailed explanation of how Lemmatization works by the step-by-step process to remove inflection forms of a word-

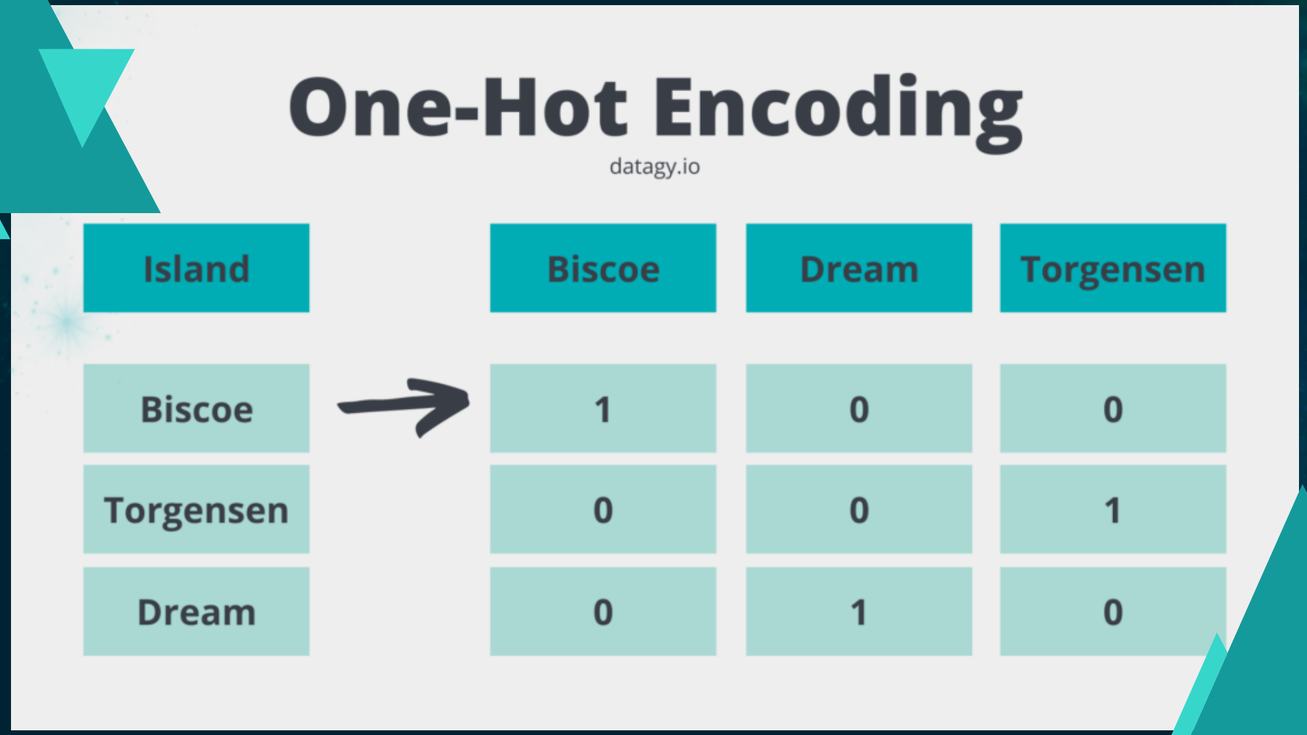


Most natural language processing tasks involve syntactic and semantic analysis, used to break down human language into machine-readable chunks.

- Semantic Analysis: Synatactic analysis is defined as analysis that tells us the logical meaning of certain given sentences or parts of those sentences.
- Syntactic Analysis: Suntactic analysis is the process of analyzing natural language with the rules of a formal grammar. It identifies the syntactic structure of a text and the dependency relationships between words.

One Hot Encoding

In this strategy, each category value is converted into a new column and assigned a 1 or 0 (notation for true/false) value to the column.Size of the one hot vector is equal to the size of the vocabulary.



BRIDGE-TYPE (TEXT)	BRIDGE-TYPE (Arch)	BRIDGE-TYPE (Beam)	BRIDGE-TYPE (Truss)	BRIDGE-TYPE (Cantilever)	BRIDGE-TYPE (Tied Arch)	BRIDGE-TYPE (Suspension)	BRIDGE-TYPE (Cable)
Arch	1	0	0	0	0	0	0
Beam	0	1	0	0	0	0	0
Truss	0	0	1	0	0	0	0
Cantilever	0	0	0	1	0	0	0
Tied Arch	0	0	0	0	1	0	0
Suspension	0	0	0	0	0	1	0
Cable	0	0	0	0	0	0	1

SAFETY-LEVEL (TEXT)	SAFETY-LEVEL (None)	SAFETY-LEVEL (Low)	SAFETY-LEVEL (Medium)	SAFETY-LEVEL (High)	SAFETY-LEVEL (Very High)
None	1	0	0	0	0
Low	0	1	0	0	0
Medium	0	0	1	0	0
High	0	0	0	1	0
Very-High	0	0	0	0	1

Word Embeddings

word embedding is a term used for the representation of words for text analysis, typically in the form of a real-valued vector that encodes the meaning of the word such that the words that are closer in the vector space are expected to be similar in meaning.

Word2Vec:

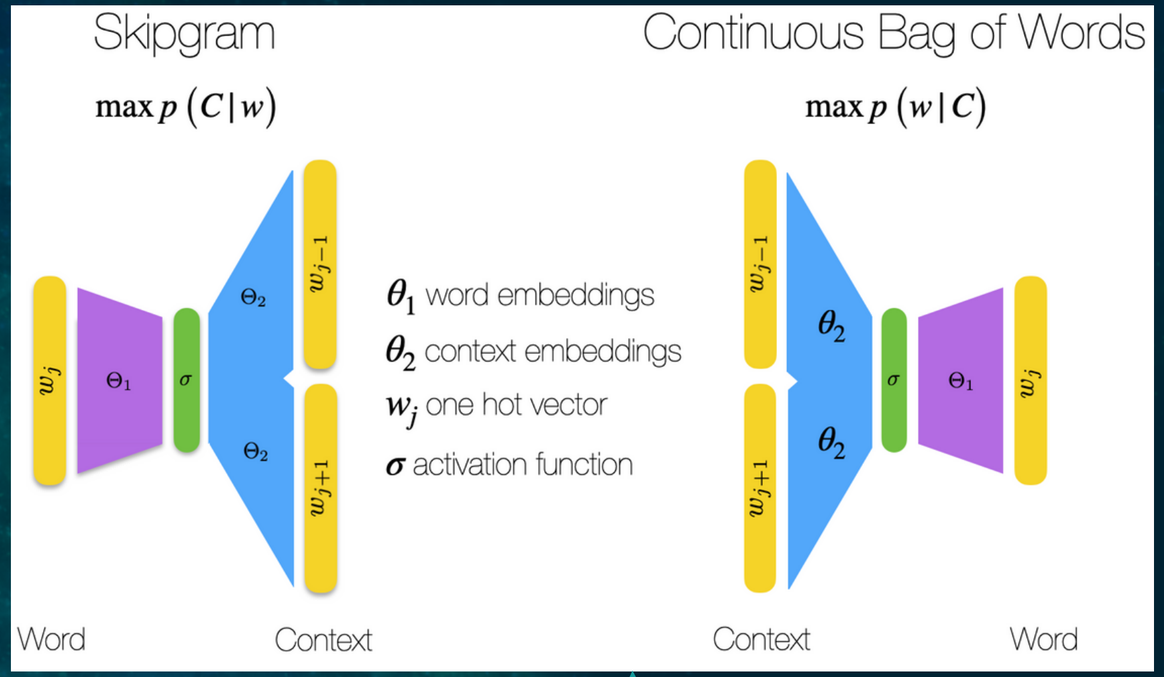
The **word2vec** tool takes a text corpus as input and produces the word vectors as output. It embeds words in a lower-dimensional vector space using a shallow neural network. The result is a set of word-vectors where vectors close together in vector space have similar meanings based on context, and word-vectors distant to each other have differing meanings.
It can be obtained using two methods:

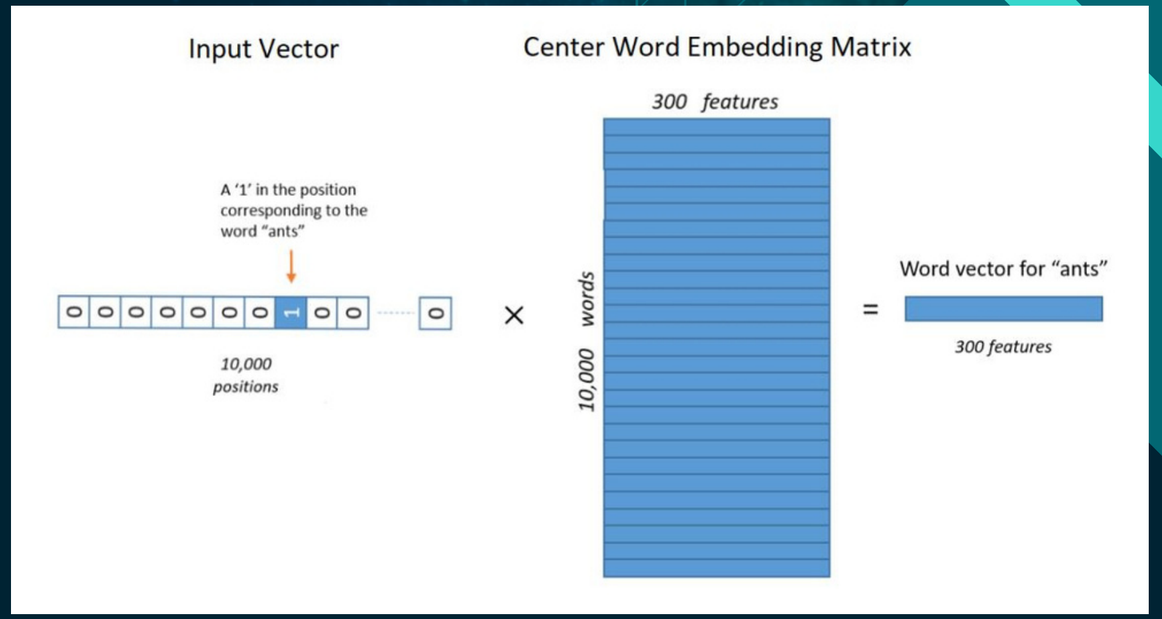
Continuous bag of words:

The **CBOW** model architecture tries to predict the current target word (the center word) based on the source context words (surrounding words).

Skipgram Model:

The **SkipGram** Model tries to predict the source context words (surrounding words) given a target word (the center word).


Skipgram
Continuous Bag of Words
max p (C | w)
max p (w | C)
θ1 word embeddings
θ2 context embeddings
wj one hot vector
σ activation function
Word
Context
Context
Word


Input Vector
Center Word Embedding Matrix
300 features
A '1' in the position corresponding to the word "ants"
10,000 positions
10,000 words
Word vector for "ants"
300 features

TF-IDF (Term Frequency-Inverse Document Frequency)

Term frequency-inverse document frequency is a text vectorizer that transforms the text into a usable vector. It combines 2 concepts, Term Frequency (TF) and Document Frequency (DF).

Term Frequency

The term frequency is the number of occurrences of a specific term in a document. Term frequency represents every text from the data as a matrix whose rows are the number of documents and columns are the number of distinct terms throughout all documents.

Document Frequency

Document frequency is the number of documents containing a specific term. Document frequency indicates how common the term is.

Inverse Document Frequency

Inverse document frequency (IDF) is the weight of a term, it aims to reduce the weight of a term if the term's occurrences are scattered throughout all the documents.

If you search something on the search engine, with the help of TFIDF values, search engines can give us the most relevant documents related to our search.The main limitation is that it does not capture the semantic meaning of the words.

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

$tf_{i,j}$ = number of occurrences of i in j
df_i = number of documents containing i
N = total number of documents

	a	an	the	football	sachin	Messi	sports
Document-1	1000	1001	1100	100	20	80	76
Document-2	1002	900	982	0	78	0	80
Document-3	1100	999	900	80	0	90	30
Document-4	1000	1000	1000	0	67	0	20
Document-5	1200	1004	1200	0	0	0	110
Document-6	1008	1100	1000	0	0	0	90

Example of term frequency matrix

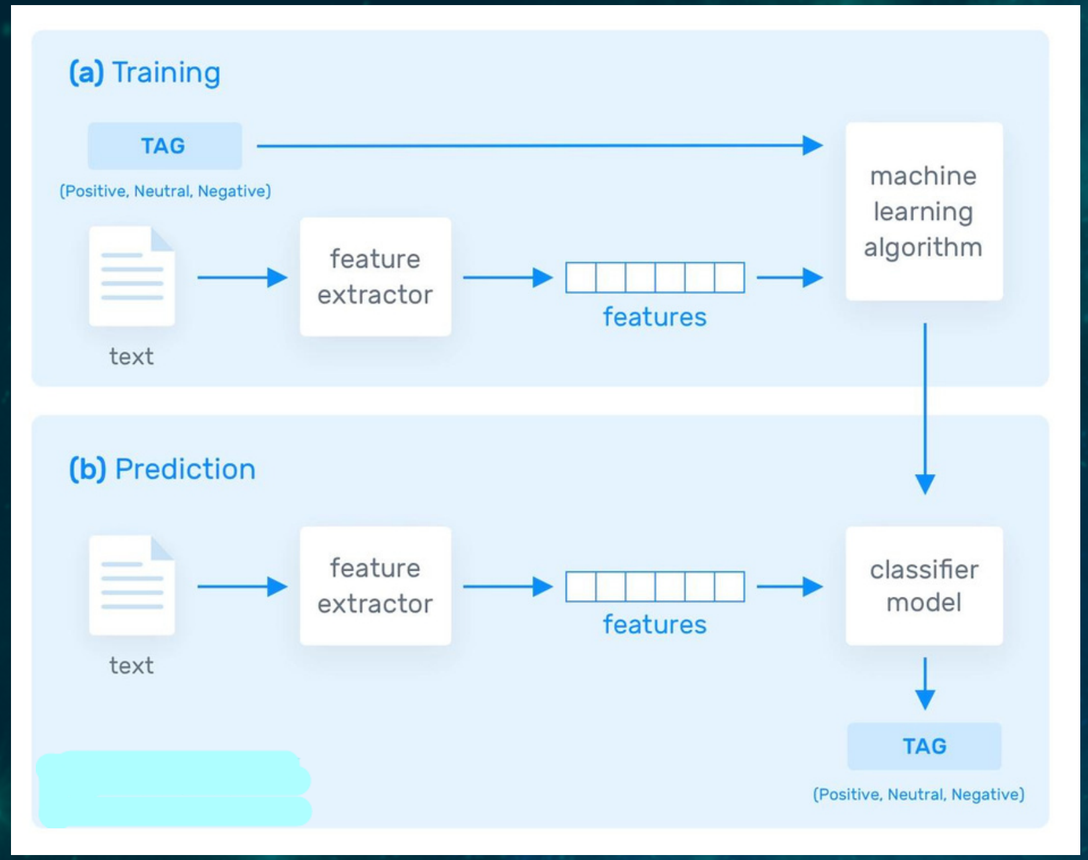
Text 1	i love natural language processing but i hate python
Text 2	i like image processing
Text 3	i like signal processing and image processing

	and	but	hate	i	image	language	like	love	natural	processing	python	signal
Text 1	0	1	1	2	0	1	0	1	1	1	1	0
Text 2	0	0	0	1	1	0	1	0	0	1	0	0
Text 3	1	0	0	1	1	0	1	0	0	2	0	1

Term	and	but	hate	i	image	language	like	love	natural	processing	python	signal
IDF	0.47712	0.47712	0.4771	0	0.1760913	0.477121	0.1760913	0.477121	0.47712125	0	0.477121	0.477121

	and	but	hate	i	image	language	like	love	natural	processing	python	signal
Text 1	0	0.47712	0.4771	0	0	0.477121	0	0.477121	0.47712125	0	0.477121	0
Text 2	0	0	0	0	0.1760913	0	0.1760913	0	0	0	0	0
Text 3	0.47712	0	0	0	0.1760913	0	0.1760913	0	0	0	0	0.477121


Intelligent Chatbots
Text Summarization
Topic Modeling
Text Generation
Sentiment Analysis
Compliance and Risk Management
Entity Extraction
Synthetic Data Generation


(a) Training
TAG
(Positive, Neutral, Negative)
machine learning algorithm
feature extractor
features
text
(b) Prediction
feature extractor
features
classifier model
text
TAG
(Positive, Neutral, Negative)


do a little text search and nobody bats an eye
do a little
NLP and everybody loses their minds
quickmeme.com