

# NLP

# Vectorization Techniques

## Week 1: Natural Language Processing

This file is meant for personal use by rameshmckv@gmail.com only.  
Sharing or publishing the contents in part or full is liable for legal action.



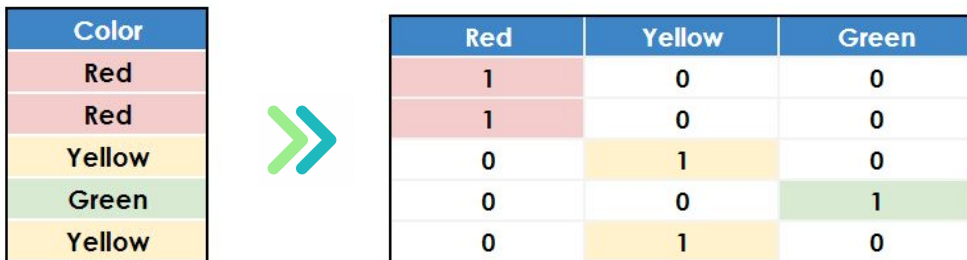
# Agenda

- **Vectorization Techniques**


# Vectorization Techniques

1. **Bag of Words (BoW):** It is used to vectorize a document. A document is represented as a vector. The number of elements in the vector is equal to the size of the vocabulary.

- We have seen the **one-hot encoding** method which marks the presence of a label as present if the value is 1 and absent if the value is 0.



Color
Red
Red
Yellow
Green
Yellow



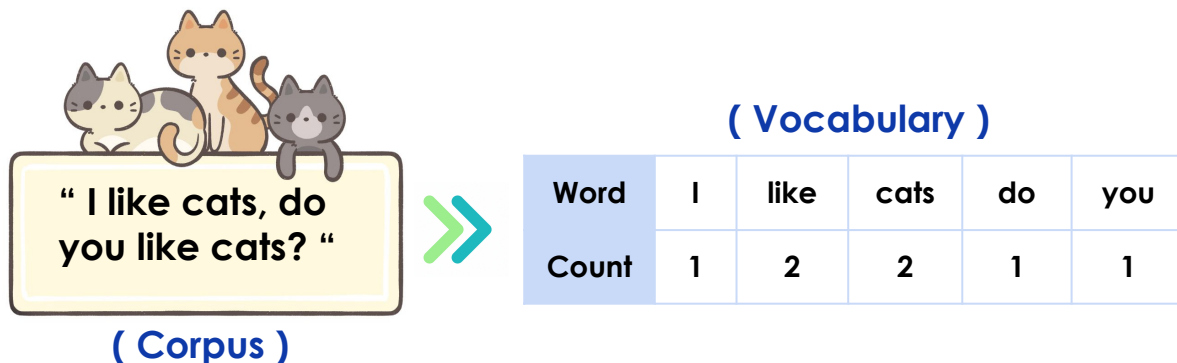
Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1
0	1	0

- In Bag of Words, instead of storing the status of the word as present or absent we **store the count of the word occurrence in a given text data.**

# Vectorization Techniques

## 1. Bag of Words (BoW)

- A **BoW vector spans the complete vocabulary**: The collection of unique words in the **corpus**. Let's look at an example to understand the Bag of Words technique better:



- The **values of the vector represent how frequently each word appears** in a specific corpus/document.

# Vectorization Techniques

- **Limitations of the Bag of Words technique**

- The Bag of Words model is perhaps the simplest text vectorization method to grasp and apply, but it comes with several limitations and drawbacks.
  1. **Vocabulary:** The vocabulary necessarily requires careful design, particularly to manage the size, which affects the sparsity of the document representations.
  2. **Sparse:** It is difficult to model sparse representations for computational and information reasons, as the models must use very little information from a large representational space.

# Vectorization Techniques

- **Limitations of the Bag of Words technique**

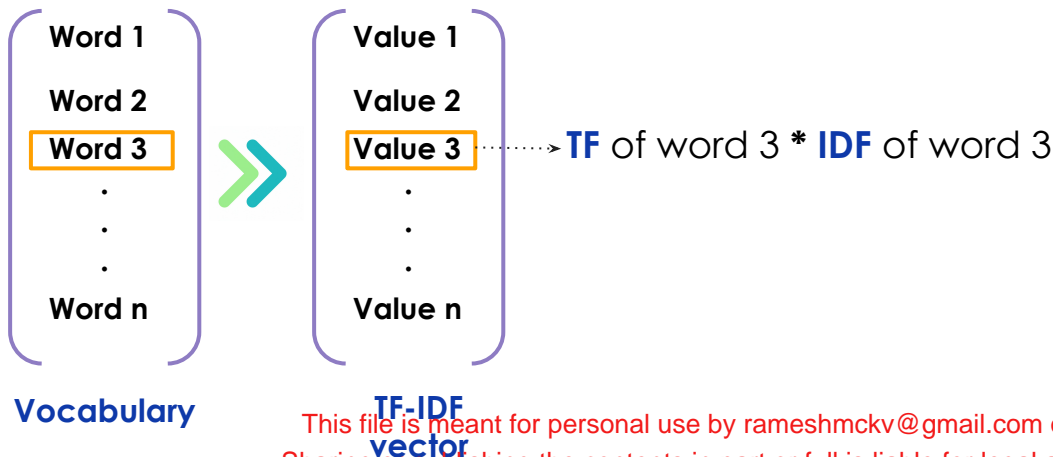
- The Bag of Words model is perhaps the simplest text vectorization method to grasp and apply, but it comes with several limitations and drawbacks.

**3. Meaning:** The meaning of the sentence is lost if the order of the words is changed. Context and meaning can provide a lot to the model, which if modeled can tell the difference between the same words arranged differently ("this is good vs is this good"), synonyms ("old car" vs "used car"), and much more.

# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

- The Bag of Words approach simply uses the count as a feature of the vector. On the other hand, **TF-IDF** takes the game up a level by using statistical measures to **evaluate how important a word is in a corpus**.
- The **value** in the resulting **vector** corresponding to each word is the **product of TF (Term Frequency) and IDF (Inverse Document Frequency)**.

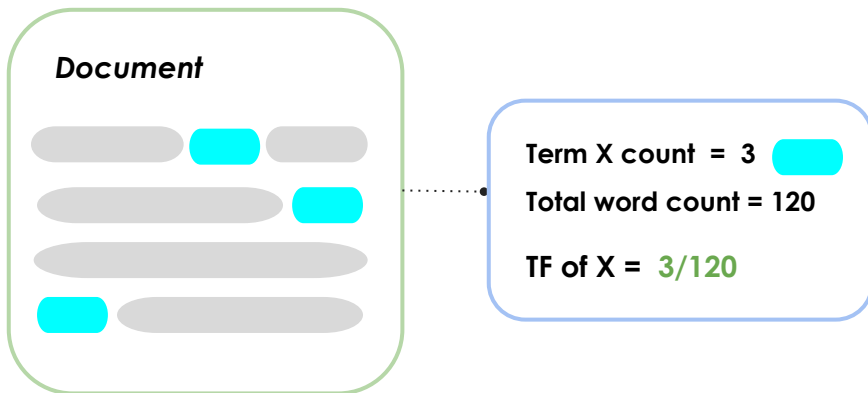


# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

Let's try to understand the meaning of Term Frequency and Inverse Document Frequency:

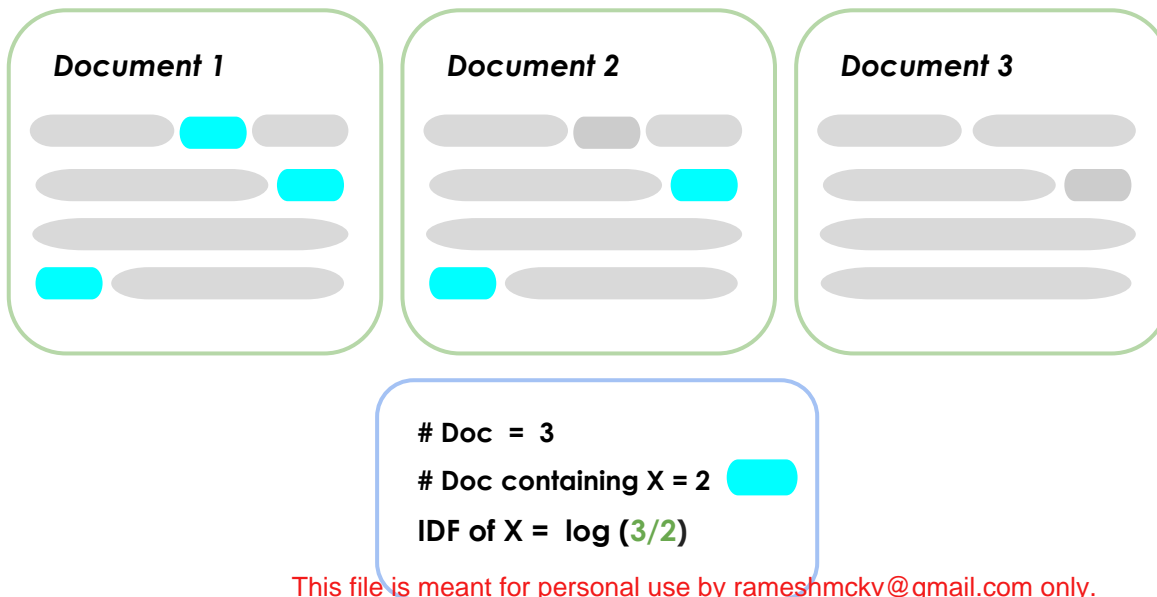
- **Term Frequency (TF)** - It is the **ratio of number of times the term “X” appears in a raw text** and the **total number of terms in the raw text**.



# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

- **Inverse Document Frequency (IDF)** - It is the **log of the ratio of total number of documents and the number of documents where the term “X” appears.**



# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

Let's look at an example and understand how to compute TF-IDF for each word in each sentence.

### ***Document 1***

It is going to be sunny today.

### ***Document 2***

Today I am not going to the office.

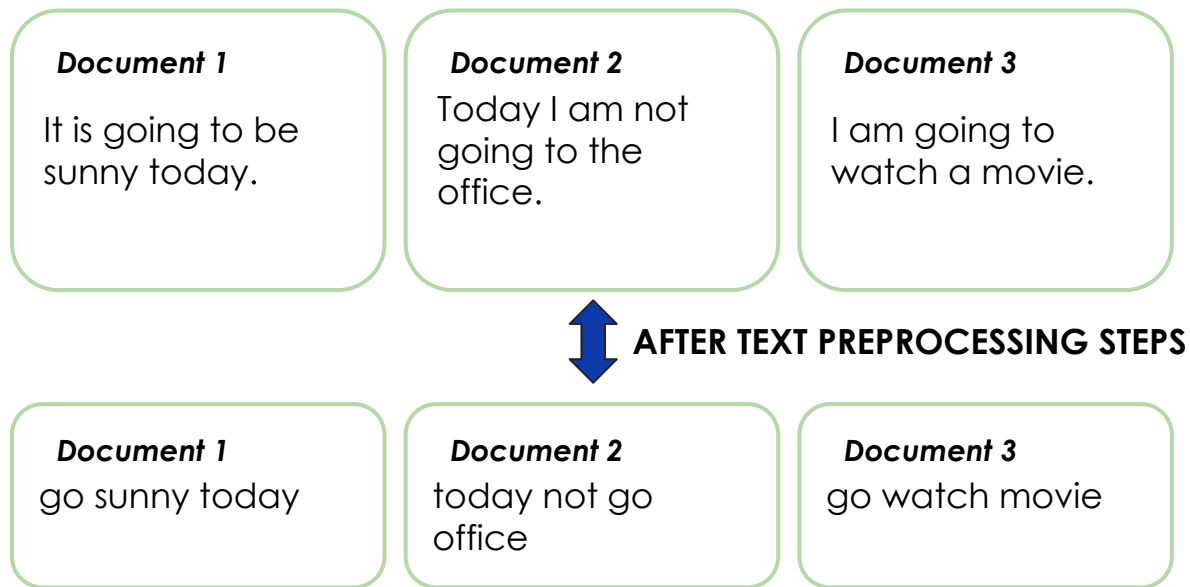
### ***Document 3***

I am going to watch a movie.

# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

After applying all the pre-processing steps on all three documents:



# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

Let's calculate the Term Frequency for document-1 - '**go sunny today**'

$$\text{TF for the word 'go'} = \frac{\text{number of times 'go' appears in document-1}}{\text{number of terms in document-1}} = \frac{1}{3}$$

Similarly,

- TF (sunny) = 1/3
- TF (today) = 1/3

# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

Let's now calculate the Term Frequency (TF) for all the unique terms in each of the documents.

Term	TF - Doc 1	TF - Doc 2	TF - Doc 3
go	1/3	1/4	1/3
sunny	1/3	0	0
today	1/3	1/4	0
not	0	1/4	0
office	0	1/4	0
watch	0	0	1/3
movie	0	0	1/3

# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

Let's calculate the Inverse Document Frequency (IDF) for document - 1 "**go sunny today**"

$$\text{TF for the word 'go'} = \log \left( \frac{\text{number of documents}}{\text{number of documents containing the word 'go'}} \right) = \log \left( \frac{3}{3} \right) = \log (1) = 0$$

Similarly,

- IDF (sunny) =  $\log(3/1) = \log(3) = 0.48$
- IDF (today) =  $\log(3/2) = 0.18$

# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

Let's calculate the Inverse Document Frequency (IDF) for all the unique terms.

Term	IDF
go	$\log(3/3) = \log(1) = 0$
sunny	$\log(3/1) = \log(3) = 0.48$
today	$\log(3/2) = 0.18$
not	$\log(3/1) = \log(3) = 0.48$
office	$\log(3/1) = \log(3) = 0.48$
watch	$\log(3/1) = \log(3) = 0.48$
movie	$\log(3/1) = \log(3) = 0.48$

# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

Now, let's calculate the **TF-IDF** for **each of the terms in document 1**.

$$\text{TF-IDF}(\text{'go'}, \text{doc 1}) = \text{TF}(\text{'go'}, \text{doc 1}) \times \text{IDF}(\text{'go'}) = 1/3 \times 0 = 0$$

$$\text{TF-IDF}(\text{'sunny'}, \text{doc 1}) = \text{TF}(\text{'sunny'}, \text{doc 1}) \times \text{IDF}(\text{'sunny'}) = 1/3 \times 0.48 = 0.16$$

$$\text{TF-IDF}(\text{'today'}, \text{doc 1}) = \text{TF}(\text{'today'}, \text{doc 1}) \times \text{IDF}(\text{'today'}) = 1/3 \times 0.18 = 0.06$$

Similarly, we can **calculate TF-IDF scores** for **each term** in **each document**.

# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

Term	TF - Doc 1	TF - Doc 2	TF - Doc 3		IDF		TF-IDF - Doc 1	TF-IDF - Doc 2	TF-IDF - Doc 3
go	1/3	1/4	1/3		0		0	0	0
sunny	1/3	0	0		0.48		0.16	0	0
today	1/3	1/4	0		0.18		0.06	0.045	0
not	0	1/4	0	*	0.48	→	0	0.12	0
office	0	1/4	0		0.48		0	0.12	0
watch	0	0	1/3		0.48		0	0	0.16
movie	0	0	1/3		0.48		0	0	0.16

# Vectorization Techniques

## 2. Term Frequency-Inverse Document Frequency (TF-IDF)

How does TF-IDF work?

- **IDF** part in it actually works as a dampening factor to **reduce the importance of terms that are common** to a lot of documents.

For example, terms like “**about**” or “**but**” will be **occurring a lot in an article**. But that **does not mean that the article is about these prepositions**. This is what **IDF is used for**.

- On the other hand, when **a keyword appears only in a small number of documents**, it is deemed **more relevant to the documents in which it appears**.

TF helps us to obtain this information.

In this way, **TF-IDF attempt to give higher relevance scores to words that occur in fewer documents within the corpus**.

# Difference between BoW and TF-IDF

Bag of Words	TF-IDF
1. Converts the words into numbers with <b>no semantic information</b> .	1. Converts the words into numbers with <b>some weighted information</b> .
2. It creates a <b>set of vectors</b> containing the <b>count of word occurrences in the document</b> .	2. It contains information on the <b>more important words</b> and the <b>less important</b> ones as well.
3. It has a disadvantage that it <b>depends on the count of words</b> and <b>emphasizes words with high frequency</b> , so important words which have a lower frequency are considered invaluable.	3. TF-IDF uses a <b>normalized count</b> where <b>each word is divided by the number of documents</b> the word appears in.

# Summary

A brief recap:

- We have learned **vectorization techniques** and gone over **two important early approaches to word vectorization**:
  - **Bag of Words (BoW)**
  - **TF-IDF**
- Both **BOW** and **TF-IDF** are **document-vectorization techniques**.
- In the upcoming section, we will look at a more recent encoding technique that aims to capture not just the lexical but **also the semantic properties of words**, that is word vectorization using the **Word2Vec method**.



# Happy Learning !

