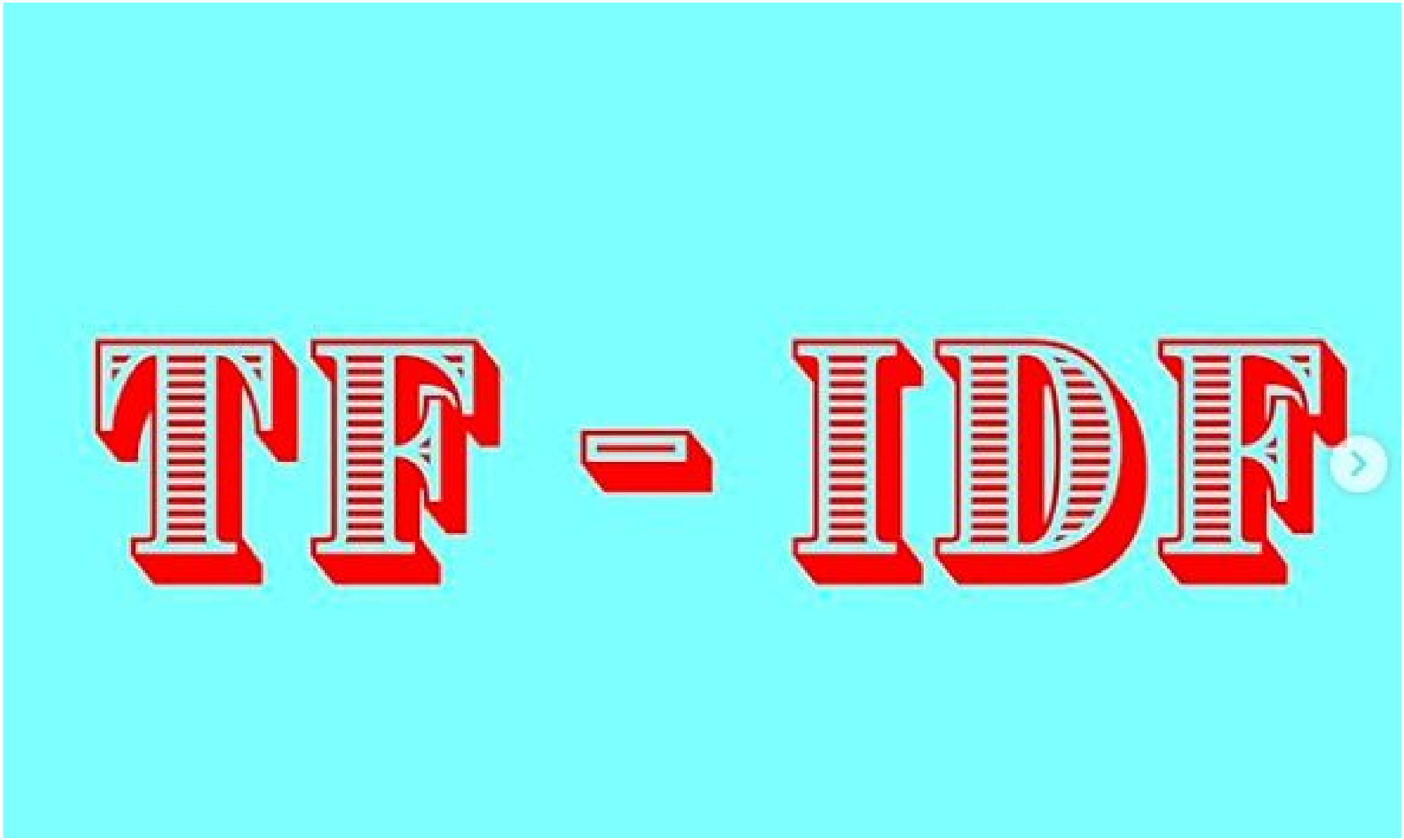
TF - IDF

Why??

Machine learning algorithms cannot work with raw text directly. Rather, the text must be converted into vectors of numbers. And **TF-IDF** is one way to do it. We usually apply bag of word techniques like binary vectorizer or count vectorizer or any similar technique.

Example (Binary Vectorizer)

It gives either 1 or 0 to every word. If it is present in the document then 1 or else 0.

Sentence 1: I love Machine learning
Sentence 2: Machine learning is my passion

I	Love	is	My	Machine	Learning	Passion
1	1	0	0	1	1	0
0	0	1	1	1	1	1

Why??

Problems with Binary Vectorizer is it gives same weight to every word. either it is commonly occurring (The, is, and etc...) or rarely occurring words (Passion, Machine etc...). To overcome this problem we use **TF - IDF**

TF - IDF

Term frequency and Inverse document frequency

Word = TF * log(IDF)

Term frequency

The number of times a word appears in a document divided by the total number of words in the document. It says how common a word is

TF = Word count / Total words

Why??

Inverse document frequency

Total document count divided by count of documents that word occurred. It gives how unique or rare a word is

IDF = log(Total document count / count of word occurring documents)

Example

Sentence 1: I love Machine learning
Sentence 2: I love Deep learning

I	Love	Artificial	Machine	Learning	Intelligence
0	0	0	0.075	0.075	0
0	0	0.075	0	0	0.075

You can the words which doesn't occur and which occur frequently have the same score (I and Love)

Applications

he higher the numerical weight value, the rarer the term. The smaller the weight, the more common the term.

- **Information retrieval**
- **Keyword Extraction**
- **Word embedding**