Count Vectorizer ⇒ Count

① We are learning Data Science.

② Data Science is a combination of Deep learning and Machine learning.

⇓

After Preprocessing

⇓

① learning data science

② data science combination deep learning machine learning

⇓

unique words

[learning, data , science , combination, deep, machine]

	learning	data	science	combination	deep	machine	oov
①	1	1	1	0	0	0	
②	2	1	1	1	1	1	
③	2	0	0	0	1	1	+1 +1 = 2

→ Representation of a single document

Document term matrix

oov → out of vocabulary

Test data : ③ NLP needs both machine learning and deep learning.

Test data : ③ NLP needs both machine learning and deep learning.

⇓

After preprocessing

nlp needs machine learning deep learning

Drawbacks of count Vectorizer :

① Curse of dimensionality

② Order is not maintained

③ It is not considering actual meaning of words.

TF - IDF ⟶ Combination of 2 things
↳ Term frequency ⟶ Inverse Documents frequency

$$\text{Term frequency} = \frac{\text{Frequency of term 't' in a document}}{\text{Total no. of words in that document}}$$

Data Science is a combination of Machine learning and Deep learning.

⇓

After preprocessing

data science combination machine learning deep learning

$$\text{Term frequency (learning)} = \frac{2}{7}$$

$$TF(\text{data}) = \frac{1}{7}$$

IDF -> Inverse Documents frequency

$$\text{Documents frequency} = \frac{\text{No. of documents containing term 't'}}{\text{Total no. of documents}}$$

① NLP needs deep learning and machine learning.

② data science contains lots of things like NLP.

$$\text{Documents frequency}(NLP) = \frac{2}{2} = 1$$

$$DF(\text{learning}) = \frac{1}{2}$$

$$IDF = \log\left(\frac{1}{\text{Documents frequency}}\right)$$

$$= \log\left(\frac{\text{Total no. of documents}}{\text{No. of documents containing term 't'}}\right)$$

$$IDF(NLP) = \log\left(\frac{1}{1}\right) = \log(1) = 0$$

$$IDF(\text{learning}) = \log\left(\frac{1}{\frac{1}{2}}\right) = \log(2)$$

$$TF\text{-}IDF = TF \times IDF$$

① We are learning NLP.

② Data Science is a combination of Deep learning and Machine learning.

③ NLP needs both machine learning and deep learning

(3) NLP needs both machine learning and deep learning.

⇓

After preprocessing

(1) learning nlp

(2) data science combination deep learning machine learning

(3) nlp needs machine learning deep learning

⇓

Unique words

	learning	nlp	data	science	combination	deep	machine	needs
(1)	0	$\frac{1}{2}\log\left(\frac{3}{2}\right)$	0	0	0	0	0	0
(2)	$\frac{2}{7}\log\left(\frac{3}{3}\right)$	0	$\frac{1}{7}\log\left(\frac{3}{1}\right)$	$\frac{1}{7}\log\left(\frac{3}{1}\right)$	$\frac{1}{7}\log\left(\frac{3}{1}\right)$	$\frac{1}{7}\log\left(\frac{3}{2}\right)$	$\frac{1}{7}\log\left(\frac{3}{2}\right)$	0
(3)	$\frac{2}{6}\log\left(\frac{3}{3}\right)$	$\frac{1}{6}\log\left(\frac{3}{2}\right)$	0	0	0	$\frac{1}{6}\log\left(\frac{3}{2}\right)$	$\frac{1}{6}\log\left(\frac{3}{2}\right)$	$\frac{1}{6}\log\left(\frac{3}{1}\right)$

1^{st} document

$$\text{TF-IDF (learning)} = \frac{1}{2} \times \log\left(\frac{3}{3}\right) = \frac{1}{2} \times \log(1)$$

$$= \frac{1}{2} \times 0 = 0$$

TF-IDF ⟶ Weightage

frequency of word ↑ ⇒ Weightage of word ↓

Movie Reviews ⇒ (1) $\overset{\checkmark}{\text{Bad}}$ Movie ✓

Movie Reviews ⇒ ① Bad Movie ✓

② Awesome Movie ✓

③ Fabulous Movie ✓

Drawbacks of TFIDF

① Curse of dimensionality

② Order is not maintained

③ It is not considering actual meaning of words.

Similarities between C.V. and TF-IDF :

① Drawbacks of both are same

② When we initiate their models, the parameters used are same.

Difference between C.V. and TFIDF

Count Vectorizer ⟶ Count

TFIDF ⟶ Weightage