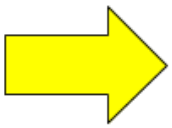
Count Vectorizer vs TFIDF

1) Count Vectorizer:

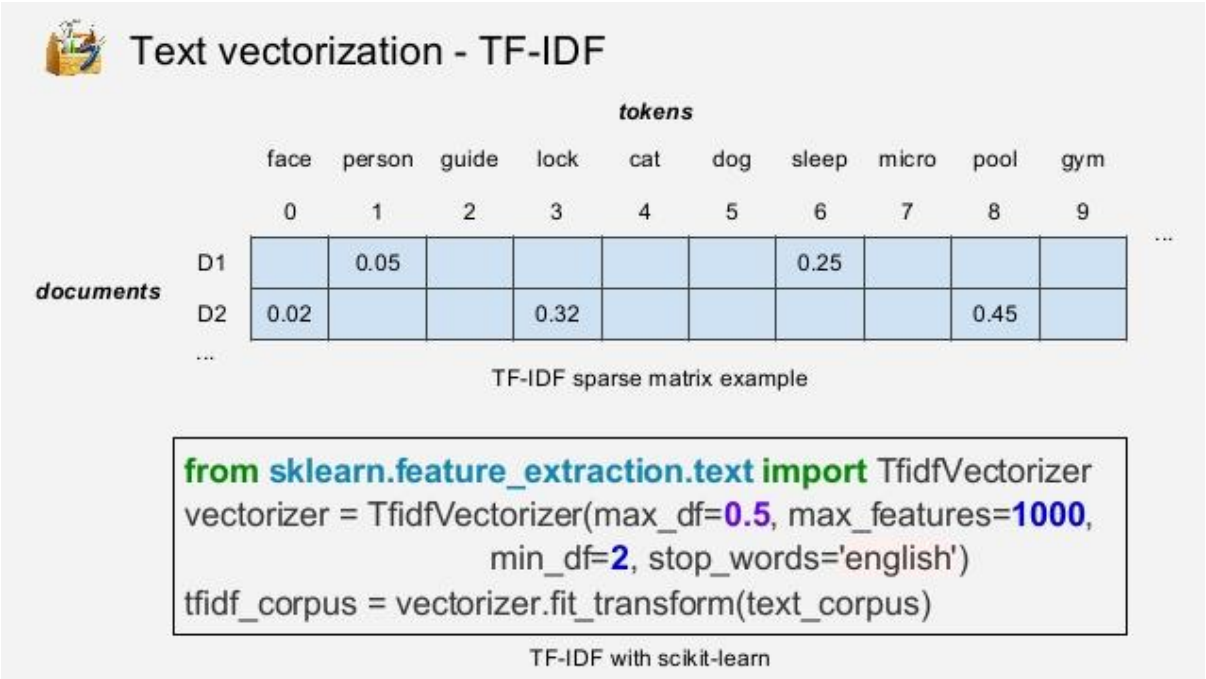
Color
Red
Red
Yellow
Green
Yellow



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

So as you can see the count vectorizer is merely count the frequency of given word in particular line. So how many unique words is in our corpus that will be our column and in given line how man times a given one is repeated.

2) Tfidf Vectorizer:



So, In **TfidfVectorizer** we consider **overall document weightage** of a word. It helps us in dealing with most frequent words. Using it we can penalize them. TfidfVectorizer weights the word counts by a measure of how often they appear in the documents.

- **Term Frequency**: This summarizes how often a given word appears within a document.
- **Inverse Document Frequency**: This downscales words that appear a lot across documents.

TF-IDF are word frequency scores that try to highlight words that are more interesting, e.g. frequent in a document but not across documents.
The TfidfVectorizer will tokenize documents, learn the vocabulary and inverse document frequency weightings, and allow you to encode new documents. Alternately, if you already have a learned CountVectorizer, you can use it with a TfidfTransformer to just calculate the inverse document frequencies and start encoding documents.

The same create, fit, and transform process is used as with the CountVectorizer.

Example

```
sample = ['problem of evil',
          'evil queen',
          'horizon problem']
```

CountVectorizer

	evil	horizon	of	problem	queen
0	1	0	1	1	0
1	1	0	0	0	1
2	0	1	0	1	0

TfidfVectorizer

	evil	horizon	of	problem	queen
0	0.517856	0.000000	0.680919	0.517856	0.000000
1	0.605349	0.000000	0.000000	0.000000	0.795961
2	0.000000	0.795961	0.000000	0.605349	0.000000