**TF-IDF**

**TF-IDF** stands for Term Frequency Inverse Document Frequency of records.

* **Term Frequency:**

tf(t,d) = count of t in d / number of words in d

ordering of terms is not significant

* **Document Frequency:**

number of occurrences

df(t) = occurrence of t in documents

:

* **Inverse Document Frequency:**

 locate the appropriate records that fit the demand

df(t) = N(t)

where

df(t) = Document frequency of a term t

N(t) = Number of documents containing the term t

idf(t) = N/ df(t) = N/N(t)

idf(t) = log(N/ df(t))

*sklearn.feature\_extraction.text.TfidfVectorizer(input)*

***Parameters:***

* ***input:****It refers to parameter document passed, it can be a filename, file or content itself.*

***Attributes:***

* ***vocabulary****\_****:****It returns a dictionary of terms as keys and values as feature indices.*
* ***idf\_:****It returns the inverse document frequency vector of the document passed as a parameter.*

***Returns:***

* ***fit\_transform():****It returns an array of terms along with tf-idf values.*
* ***get\_feature\_names():****It returns a list of feature names.*

1. =Is the unique value

**Example 3:**In this program, tf-idf values are computed from a corpus having similar documents.

# assign documents

d0 **=** 'Geeks for geeks!'

d1 **=** 'Geeks for geeks!'

