Natural Language Processing (NLP)

NLP

- Machine translation.
- Information retrieval (e.g., search engines).
- Sentiment analysis (e.g., positive, negative, happiness, sadness, etc.).
- Information extraction (e.g., summary, keywords, etc.).
- Text generation.

Text processing techniques

- Remove stopwords: *a*, *the*, *it*, *is*, *etc.*
- Keep the most *K* “important” words.
- Stemming: chop words to its root. E.g., swimmer, swimming… → swim.

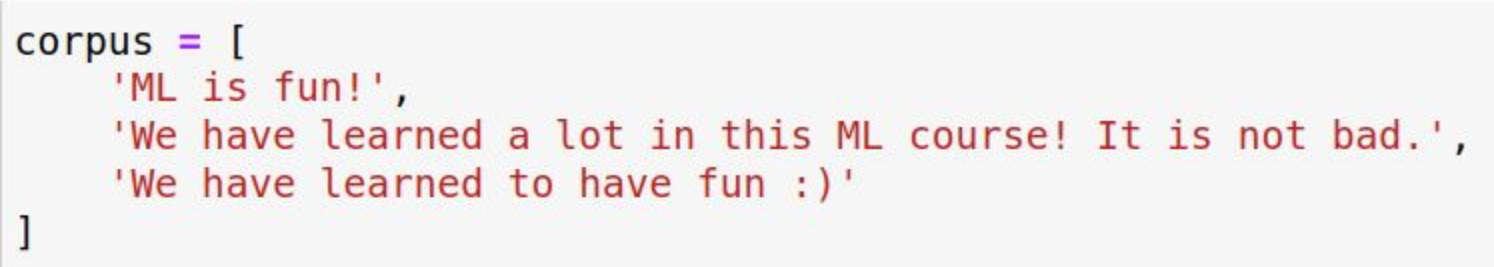
Corpus

“ML is fun!”

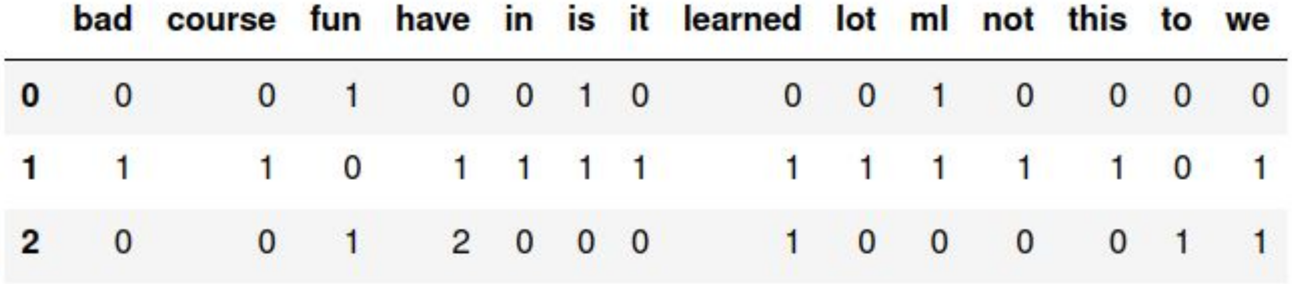
“We have learned a lot in this ML course! It is not bad.”

“We have learned to have fun :)”

Bag-of-Words



```
corpus = [
    'ML is fun!',
    'We have learned a lot in this ML course! It is not bad.',
    'We have learned to have fun :)'
]
```

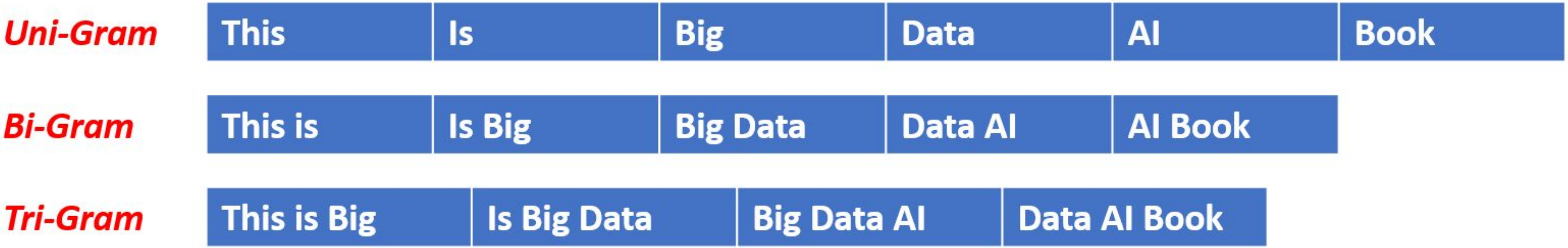


	bad	course	fun	have	in	is	it	learned	lot	ml	not	this	to	we
0	0	0	1	0	0	1	0	0	0	1	0	0	0	0
1	1	1	0	1	1	1	1	1	1	1	1	1	0	1
2	0	0	1	2	0	0	0	1	0	0	0	0	1	1

Bag-of-Words

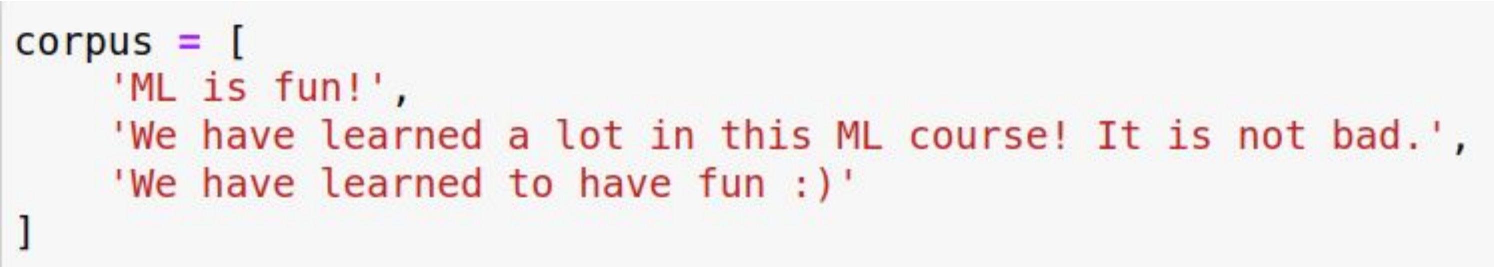
- **Problem**: we lose semanting meaning of words (we lose context).
- **Example**:
    - “not bad” means “decent” or even “good”, which is a positive thing.
    - In a bag-of-words we separate “not” and “bad” in different columns.
    - The model learns that it says “bad”, which is negative.

N-Gram model

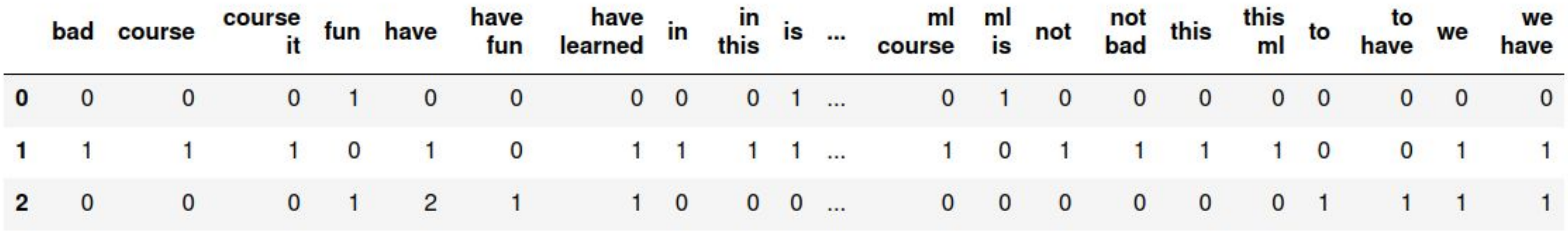


Uni-Gram	This	Is	Big	Data	AI	Book
Bi-Gram	This is	Is Big	Big Data	Data AI	AI Book	
Tri-Gram	This is Big	Is Big Data	Big Data AI	Data AI Book		

Bag-of-2-Grams



```
corpus = [
    'ML is fun!',
    'We have learned a lot in this ML course! It is not bad.',
    'We have learned to have fun :)'
]
```



	bad	course	course it	fun	have	have fun	have learned	in	in this	is	...	ml course	ml is	not	not bad	this	this ml	to	to have	we	we have
0	0	0	0	1	0	0	0	0	0	1	...	0	1	0	0	0	0	0	0	0	0
1	1	1	1	0	1	0	1	1	1	1	...	1	0	1	1	1	1	0	0	1	1
2	0	0	0	1	2	1	1	0	0	0	...	0	0	0	0	0	0	1	1	1	1

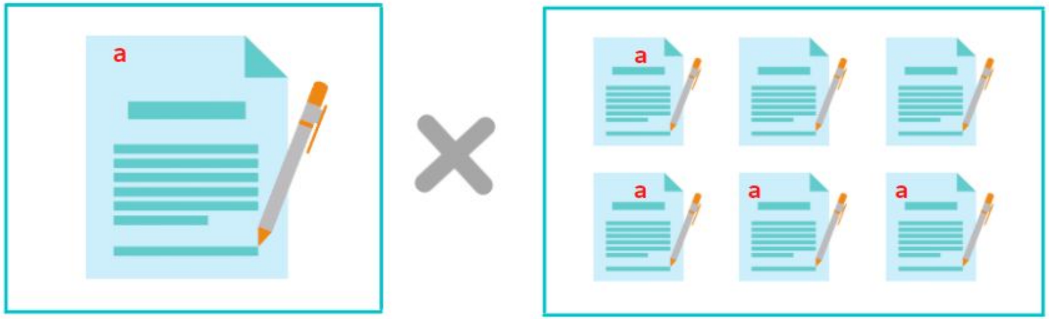
Bag-of-N-Grams

- **Problem**: Increase in feature space.
    - With a very big corpus it may become infeasible.

TF-IDF (Term Frequency-Inverse Document Frequency)

TF

IDF



Frequency of a word
within the document

Frequency of a word
across the documents

Term Frequency

TF = number of times the term appears in the document / total number of terms in the document.

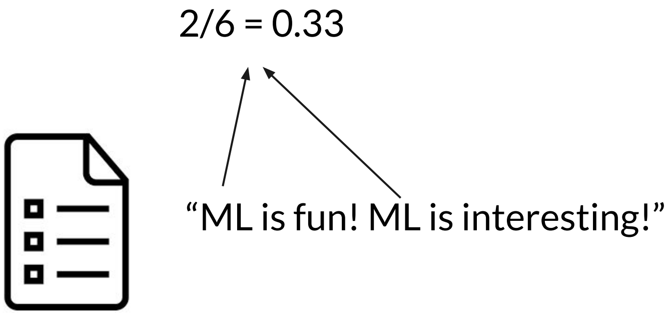
Term Frequency

TF = number of times the term appears in the document / total number of terms in the document.

“ML is fun! ML is interesting!”

Term Frequency

TF = number of times the term appears in the document / total number of terms in the document.

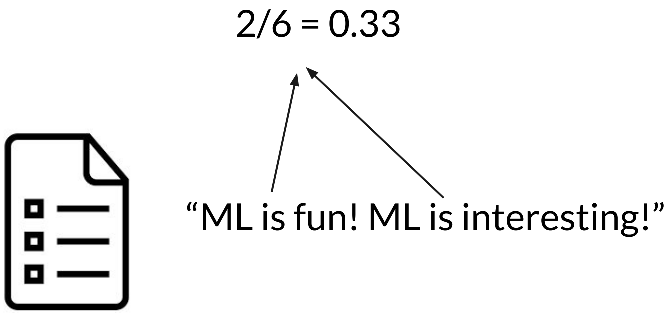


2/6 = 0.33

“ML is fun! ML is interesting!”

Term Frequency

TF = number of times the term appears in the document / total number of terms in the document.



2/6 = 0.33

“ML is fun! ML is interesting!”

Term Frequency

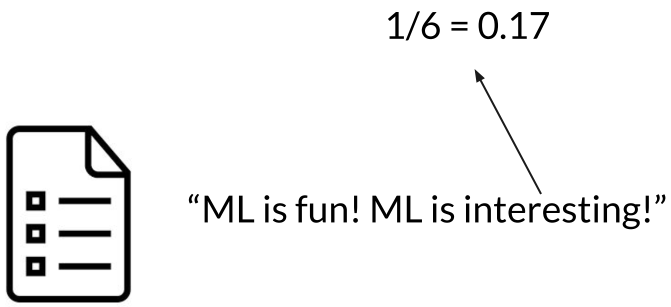
TF = number of times the term appears in the document / total number of terms in the document.

1/6 = 0.17

“ML is fun! ML is interesting!”

Term Frequency

TF = number of times the term appears in the document / total number of terms in the document.



1/6 = 0.17

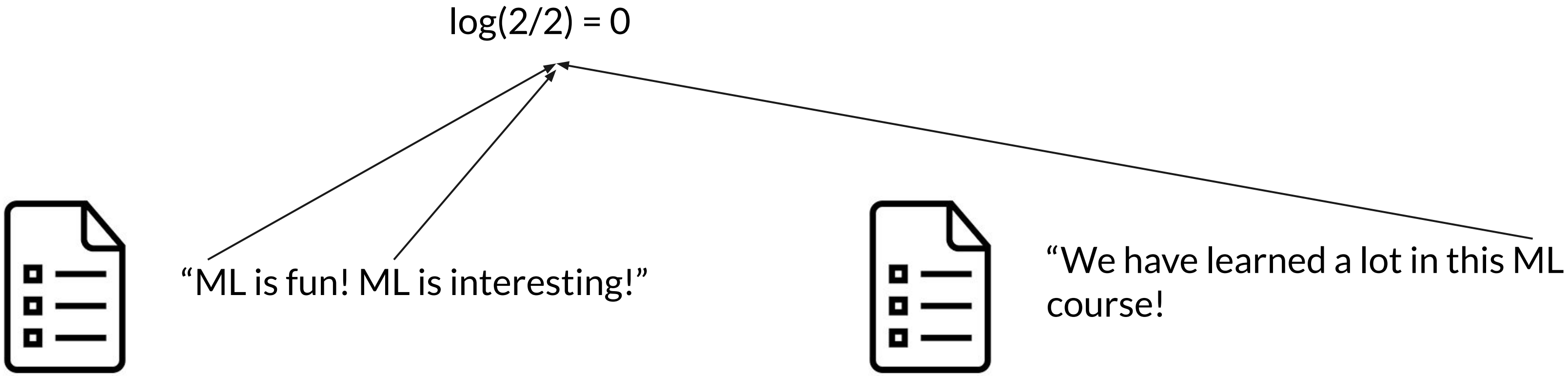
“ML is fun! ML is interesting!”

Inverse Document Frequency

IDF = log(number of documents in the corpus / number of documents in the corpus that contain the term)

Inverse Document Frequency

IDF = log(number of documents in the corpus / number of documents in the corpus that contain the term)



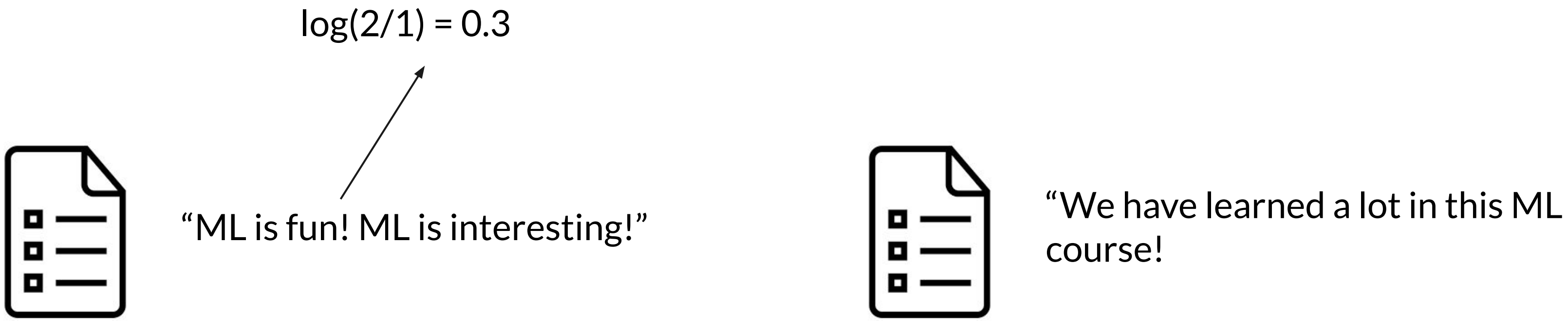
log(2/2) = 0

“ML is fun! ML is interesting!”

“We have learned a lot in this ML course!

Inverse Document Frequency

IDF = log(number of documents in the corpus / number of documents in the corpus that contain the term)



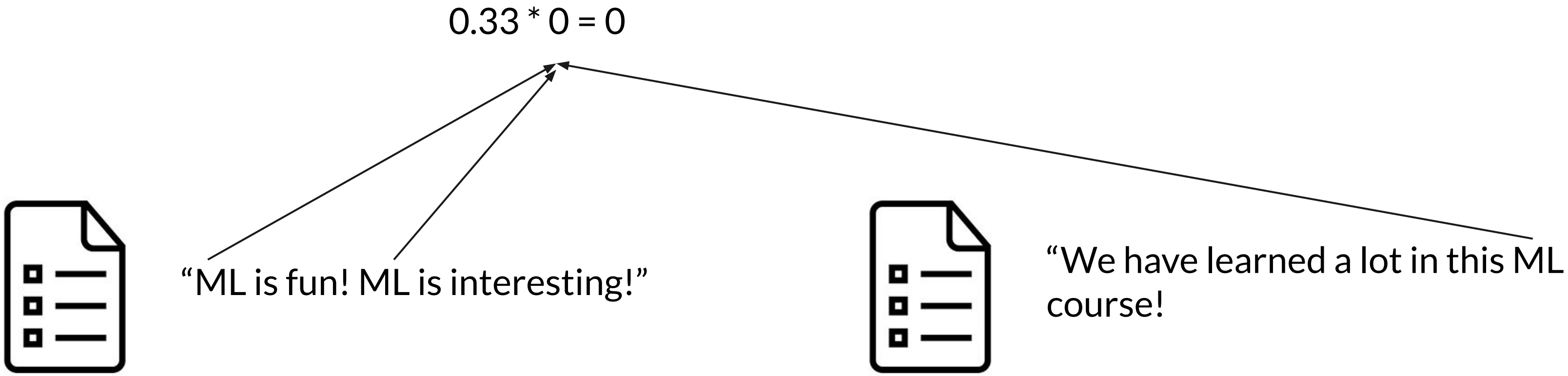
log(2/1) = 0.3

“ML is fun! ML is interesting!”

“We have learned a lot in this ML course!

Term Frequency-Inverse Document Frequency

TF-IDF = TF * IDF



0.33 * 0 = 0

“ML is fun! ML is interesting!”

“We have learned a lot in this ML course!

Term Frequency-Inverse Document Frequency

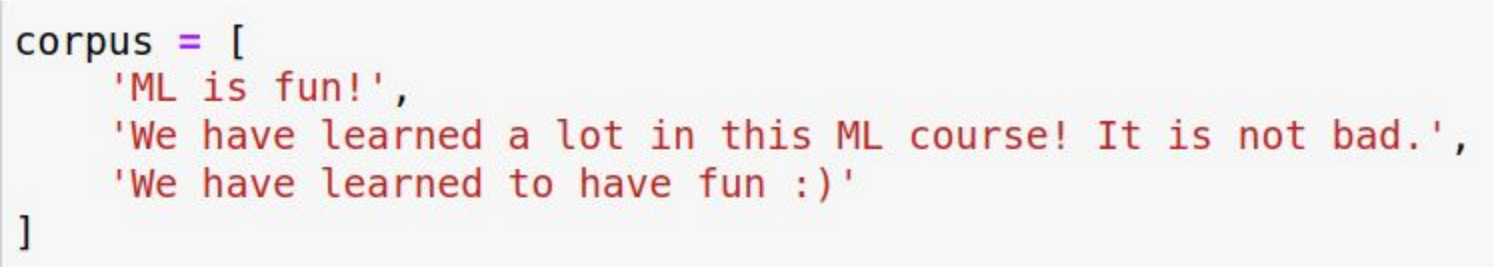
TF-IDF = TF * IDF

0.17 * 0.3 = 0.05

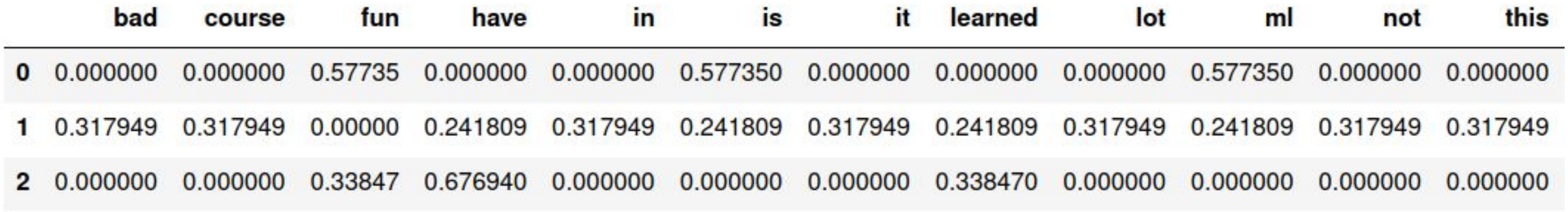
“ML is fun! ML is interesting!”

“We have learned a lot in this ML course!”

TF-IDF

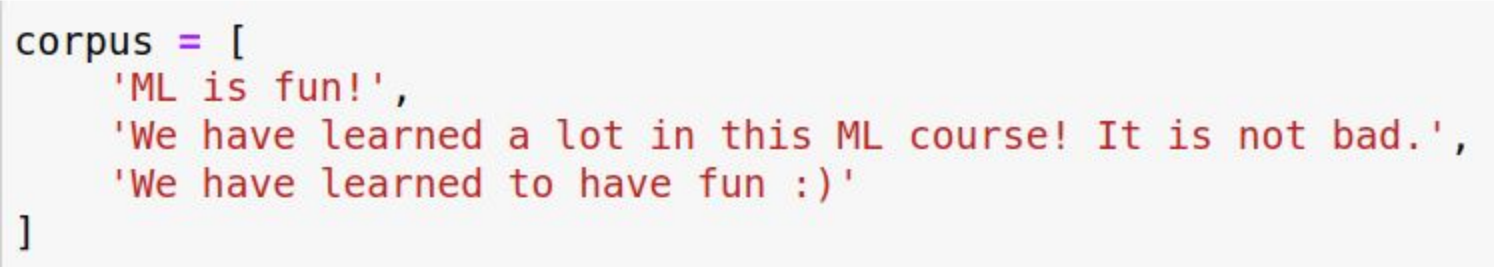


```
corpus = [
    'ML is fun!',
    'We have learned a lot in this ML course! It is not bad.',
    'We have learned to have fun :)'
]
```

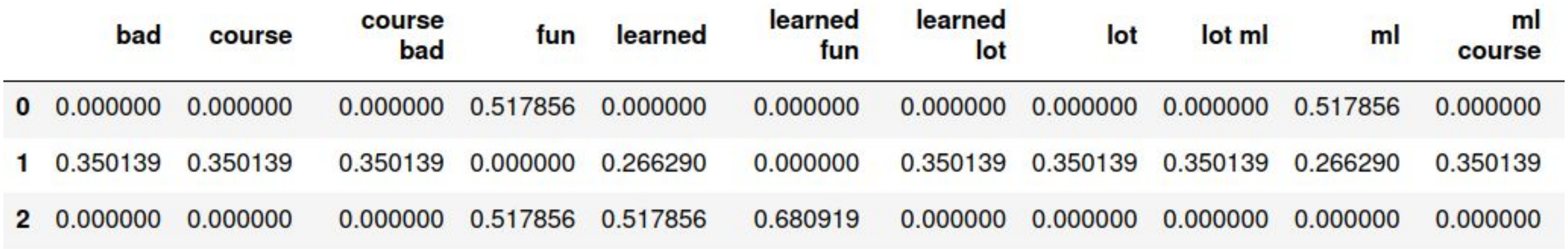


	bad	course	fun	have	in	is	it	learned	lot	ml	not	this
0	0.000000	0.000000	0.57735	0.000000	0.000000	0.577350	0.000000	0.000000	0.000000	0.577350	0.000000	0.000000
1	0.317949	0.317949	0.00000	0.241809	0.317949	0.241809	0.317949	0.241809	0.317949	0.241809	0.317949	0.317949
2	0.000000	0.000000	0.33847	0.676940	0.000000	0.000000	0.000000	0.338470	0.000000	0.000000	0.000000	0.000000

TF-IDF 2-Gram



```
corpus = [
    'ML is fun!',
    'We have learned a lot in this ML course! It is not bad.',
    'We have learned to have fun :)'
]
```



	bad	course	course bad	fun	learned	learned fun	learned lot	lot	lot ml	ml	ml course
0	0.000000	0.000000	0.000000	0.517856	0.000000	0.000000	0.000000	0.000000	0.000000	0.517856	0.000000
1	0.350139	0.350139	0.350139	0.000000	0.266290	0.000000	0.350139	0.350139	0.350139	0.266290	0.350139
2	0.000000	0.000000	0.000000	0.517856	0.517856	0.680919	0.000000	0.000000	0.000000	0.000000	0.000000