Summary

	Hello	World
A = Hello, World! ➡	1	1
B = Hello! ➡	1	0

...which is the same value we got when we took the **cos(45°)**.

$$\text{Cosine Similarity} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} = \frac{(1 \times 1) + (1 \times 0)}{\sqrt{1^2 + 1^2}\sqrt{1^2 + 0^2}} = 0.71$$

World 1

= 45° ⟶ cos(45°) = 0.71 = Cosine Similarity

1   2

Hello

	I	Love	Troll	2	Gymkata
A = I love Troll 2! ➡	1	1	1	1	0
B = I love Gymkata! ➡	1	1	0	0	1

Then we do the math and we get **0.58**.

$$\text{Cosine Similarity} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

$$= \frac{(1 \times 1) + (1 \times 1) + (1 \times 0) + (1 \times 0) + (0 \times 1)}{\sqrt{1^2 + 1^2 + 1^2 + 1^2 + 0^2}\sqrt{1^2 + 1^2 + 0^2 + 0^2 + 1^2}}$$

$$= 0.58$$

```apache
           user1   user2   user3   user4   user5
book1        5       4       3       2       1
book2        4       3       2       1       5
book3        3       2       1       5       4
book4        2       1       5       4       3
book5        1       5       4       3       2
```

```apache
[[1.         0.13018891 0.30618622 0.         0.         ]
 [0.13018891 1.         0.27575882 0.         0.07715167]
 [0.30618622 0.27575882 1.         0.         0.09090909]
 [0.         0.         0.         1.         0.         ]
 [0.         0.07715167 0.09090909 0.         1.         ]]
```

book1:book1	book1:book2	book1:book3	book1:book4	book1:book5
book2:book1	book2:book2	book2:book3	book2:book4	book2:book5
book3:book1	book3:book2	book3:book3	book3:book4	book3:book5
book4:book1	book4:book2	book4:book3	book4:book4	book4:book5
book5:book1	book5:book2	book5:book3	book5:book4	book5:book5

TF IDF MATRIX

```python
corpus = ["This is the first document.", "This document is the seco
nd document.", "And this is the third one."]

# Create a TfidfVectorizer object
vectorizer = TfidfVectorizer()

# Transform the corpus into a TF-IDF matrix
tfidf_matrix = vectorizer.fit_transform(corpus)

# Print the shape of the matrix
print(tfidf_matrix.shape)

# Output:                  each row represent a description
# (3, 20)                  each coloumn represent a word

# Print the matrix
print(tfidf_matrix.toarray())

# Output:
# [[0.37437956 0.         0.37437956 0.37437956 0.37437956 0.
#   0.         0.         0.         0.         0.37437956 0.
#   0.         0.         0.         0.         0.         0.374379
```