**T1: (PAPER)** Make TfIdf matrix from following term-document coincidence matrix:

and find cosine distance between Document 1 and other 4 documents

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **D1** | **D2** | **D3** | **D4** | **D5** |
| **defender** | 4 | 1 |  | 2 | 4 |
| **shoot** | 3 | 2 | 1 | 3 |  |
| **scored** | 1 |  |  | 1 |  |
| **government** |  | 4 |  |  |  |
| **serious** |  |  | 3 |  |  |
| **different** | 2 | 1 | 5 | 3 |  |
| **president** |  | 5 | 2 |  | 1 |

**T2:** (IMPLEMENT CODE) Install scikit-learn, your code should find TF-IDF matrix for entered sentences. For example

x = ["the cat is running","the dog is running","this is beautiful day for running", "cats and dogs are home animal"]

**T3:** (IMPLEMENT CODE) Use BBC dataset that was used earlier, implement search, it should use TfIDf. User enters some query, and it should find nearest documents (news) and output them. To do this you should implement TFIDF, and cosine distance.