1. Building recommender systems using NMF

2. Finding similar articles

Suppose that you are an engineer at a large online newspaper. You've been given the task of recommending articles that are similar to the article currently being read by a customer. Given an article, how can you find articles that have similar topics? In this video, you'll learn how to solve this problem, and others like it, by using NMF.

3. Strategy

Our strategy for solving this problem is to apply NMF to the word-frequency array of the articles, and to use the resulting NMF features. You learned in the previous videos these NMF features describe the topic mixture of an article. So similar articles will have similar NMF features. But how can two articles be compared using their NMF features? Before answering this question, let's set the scene by doing the first step.

4. Apply NMF to the word-frequency array

You are given a word frequency array articles corresponding to the collection of newspaper articles in question. Import NMF, create the model, and use the fit_transform method to obtain the transformed articles. Now we've got NMF features for every article, given by the columns of the new array.

5. Strategy

Now we need to define how to compare articles using their NMF features.

6. Versions of articles

Similar documents have similar topics, but it isn't always the case that the NMF feature values are exactly the same. For instance, one version of a document might use very direct language,

7. Versions of articles

whereas other versions might interleave the same content with meaningless chatter. Meaningless chatter reduces the frequency of the topic words overall, which reduces the values of the NMF features representing the topics.

8. Versions of articles

However, on a scatter plot of the NMF features, all these versions lie on a single line passing through the origin.

9. Cosine similarity

For this reason, when comparing two documents, it's a good idea to compare these lines. We'll compare them using what is known as the cosine similarity, which uses the angle between the two lines. Higher values indicate greater similarity. The technical definition of the cosine similarity is out the scope of this course, but we've already gained an intuition.

10. Calculating the cosine similarities

Let's see now how to compute the cosine similarity. Firstly, import the normalize function, and apply it to the array of all NMF features. Now select the row corresponding to the current article, and pass it to the dot method of the array of all normalized features. This results in the cosine similarities.

11. DataFrames and labels

With the help of a pandas DataFrame, we can label the similarities with the article titles. Start by importing pandas. After normalizing the NMF features, create a DataFrame whose rows are the normalized features, using the titles as an index. Now use the loc method of the DataFrame to select the normalized feature values for the current article, using its title 'Dog bites man'. Calculate the cosine similarities using the dot method of the DataFrame.

12. DataFrames and labels

Finally, use the nlargest method of the resulting pandas Series to find the articles with the highest cosine similarity. We see that all of them are concerned with 'domestic animals' and/or 'danger'!