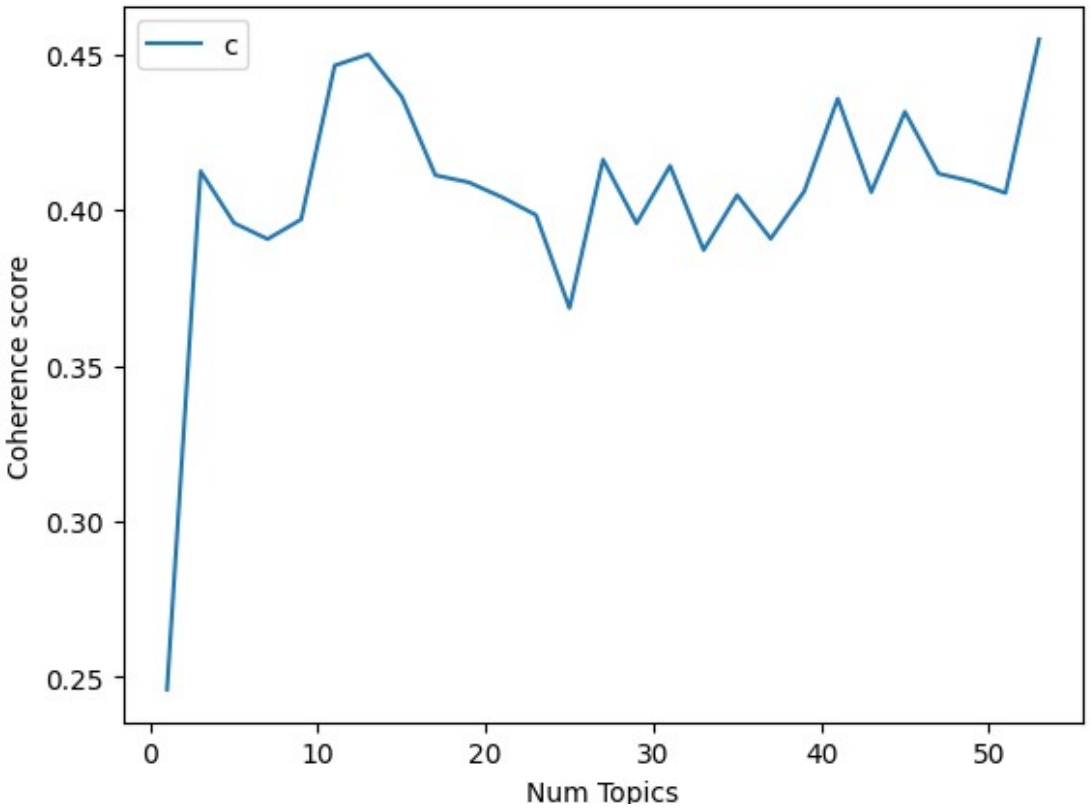
ReadMe

Some important points are listed below :

1) Out of 5 original columns the only columns that are considered are title and content.

2) A new column is added which is the concatenation of title and content in that order.
t_content = title + content

3) To find the number of topics two approaches are used :
a) Hierarchical Dirichlet Process from Gnesim. Using this we get 20 topics and I discard this approach.
b) Coherence score based approach : The number of topics is treated as a hyperparameter and we see with which we get the highest coherence score.



4) Coherence score with 13 topics is just 1% less than max coherence score we saw which was with 53 topics.

5) 13 topics are easier to label than 53 topics hence I choose 53 topics

6) Build the optimal model with 13 topics.

7) This model is then used to find the topic distribution for each of the t_content row we created earlier representing a document

8) Post this we get the topic modelling for each document. We choose the maximum probability

and pick the topic associated with it as the primary topic to be appended.

For example :

```
[(0, 0.070644714),
 (2, 0.043895364),
 (3, 0.15391311),
 (4, 0.021151911),
 (7, 0.013101622),
 (8, 0.01711651),
 (9, 0.010327547),
 (10, 0.6307997),
 (12, 0.02445142)]
```

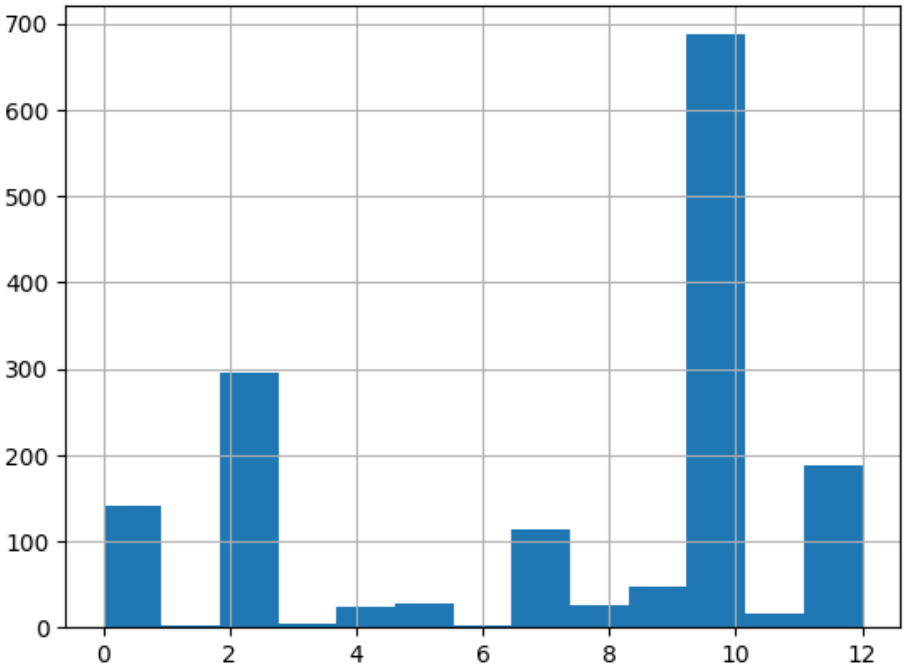
for 13 topics if this is the topic's probability distribution for a document then 0.63 is the maximum probability and 10 is the primary topic associated with this document.

In case we want to choose n primary topics then this algo can be modified .

9) Each document has now 1 primary topics associated to it. Below are each topic's counts :

10	687
2	296
12	187
0	142
7	114
9	48
5	27
8	26
4	23
11	16
3	4
1	3
6	2

10) Below is the histogram for the same



11) Top 10 words corresponding to each topic :

```
0 ['ireland', 'northern', 'said', 'ni', 'uk', 'mr', 'government', 'protocol', 'may', 'office']
1 ['airport', 'used', 'weapon', 'operation', 'track', 'building', 'air', 'station', 'military', 'risk']
2 ['premium', 'ulster', 'back', 'belfast', 'time', 'day', 'club', 'week', 'last', 'since']
3 ['case', 'agreement', 'amid', 'dr', 'eu', 'unionist', 'loyalist', 'instead', 'shortage', 'visitor']
4 ['child', 'head', 'opportunity', 'martin', 'must', 'mr', 'varadkar', 'continue', 'sign', 'taoiseach']
5 ['return', 'title', 'championship', 'action', 'charge', 'win', 'race', 'line', 'series', 'world']
6 ['recently', 'wanted', 'soldier', 'founder', 'rock', 'native', 'drama', 'australia', 'solution', 'sporting']
7 ['man', 'court', 'police', 'old', 'murder', 'death', 'attack', 'belfast', 'year', 'prison']
8 ['ulster', 'rugby', 'podcast', 'adam', 'scotland', 'low', 'province', 'bradley', 'round', 'james']
9 ['job', 'service', 'health', 'school', 'department', 'hour', 'hospital', 'announced', 'executive', 'concern']
10 ['year', 'new', 'one', 'first', 'time', 'world', 'show', 'say', 'said', 'life']
11 ['ukraine', 'war', 'russia', 'parent', 'zelensky', 'ukrainian', 'invasion', 'funding', 'armagh', 'arm']
12 ['year', 'belfast', 'said', 'number', 'city', 'company', 'food', 'car', 'last', 'road']
```

12) Finally we can use this information to fill the dataframe and write it to an excel file.

What more could have been done:

1) I have only used unigrams to feed to the LDA model. There are bigrams present in the text like Northern Ireland, New Year etc. Ideally we should identify these bigrams and pass to the model. Vocabulary should contain bigrams and trigrams as well.

2) I have left the final topics as integers. LDA does not give topics names. We have 13 topics. Upon reading the words that make up a topic and some texts easily names can be given.