SEIS 764 Project

Topic modeling for news articles

Final Report

by Yiding Weng

May 23, 2019

Problem

With the rapid growth of internet and online information services, printing medias, more and more information is available and accessible. Explosion of information has caused a well recognized information overload problem. There is no time to read everything and yet we have to make critical decisions based on whatever information is available. Without absorbing the text information line by line, how to know what is the significant writing worth paying attention? What is the central theme?

One of the primary applications of natural language processing is to automatically extract what topics people are discussing from large volumes of text. Knowing what people are talking about and understanding their problems and opinions is highly valuable to businesses, administrators, political campaigns. It will be very helpful to have an automated algorithm that can read through the text documents and automatically output the topics discussed.

I will use news articles as the source of information, and extract the naturally discussed topics.

Data

Proposed Data Collection:

All the news: 143,000 articles from 15 American publications

Data Location: https://www.kaggle.com/snapcrack/all-the-news

Approach

I will be using the Latent Dirichlet Allocation (LDA) from Gensim package along with the Mallet's implementation (via Gensim). Mallet has an efficient implementation of the LDA. It is known to run faster and gives better topics segregation.

We will also extract the volume and percentage contribution of each topic to get an idea of how important a topic is.

Data wrangling

Download nltk stop words and spacy model

We will need the stopwords from NLTK and spacy's en model for text pre-processing. Then we will be using the spacy model for lemmatization. Lemmatization is converting a word to its root word.

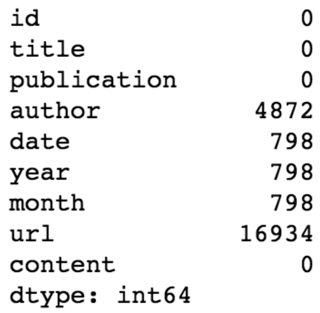
For example: the lemma of the word 'machines' is 'machine'. Likewise, 'walking' –> 'walk', 'mice' –> 'mouse' and so on.

Data sampling

This version of the dataset contains about 142k news articles from 15 different medias. This is available as https://www.kaggle.com/jannesklaas/analyzing-the-news/data. Due to the limited computing power, I will take a sample of 30% of the total data for our study in this project.

Data cleaning

'title' and 'content' are the important features that have the text information we are interested to study. Since this title dataset does not have empty fields in 'title' or 'content' information, we will not remove any rows of data.



```
id                 0
title              0
publication        0
author          4872
date             798
year             798
month            798
url            16934
content            0
dtype: int64
```

Remove emails and newline characters

There are many emails sign (@), newline and extra spaces that is quite distracting. So get rid of them using regular expressions. After removing the emails and extra spaces, we need to break down each sentence into a list of words through tokenization, while clearing up all the messy text in the process.

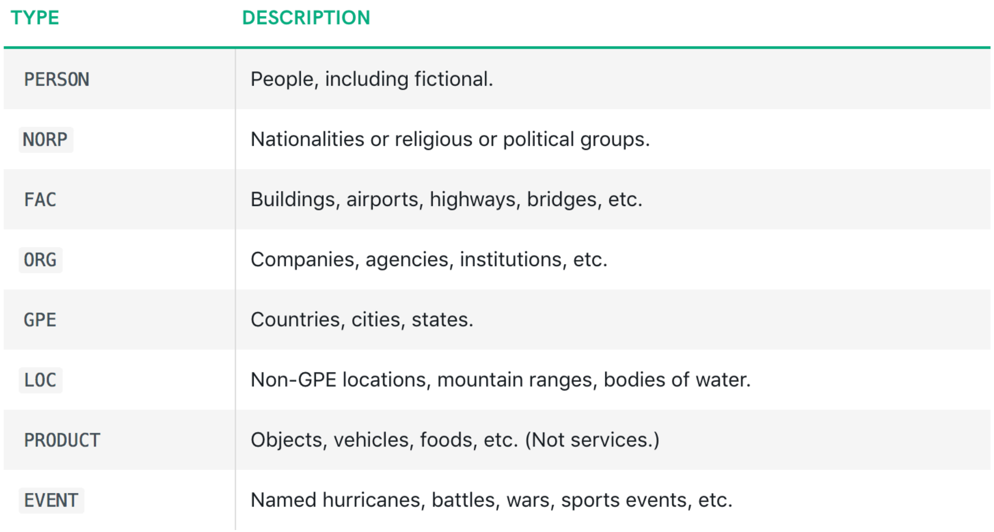
Creating Bigram and Trigram Models

Bigrams are two words frequently occurring together in the document. Trigrams are 3 words frequently occurring. Gensim’s Phrases model can build and implement the bigrams, trigrams, quadgrams and more. The two important arguments to Phrases are min_count and threshold. The higher the values of these param, the harder it is for words to be combined to bigrams.

Some examples in our example are: ‘health_insurance’, ‘health_care’, ‘white_house’ etc.

Exploratory analysis

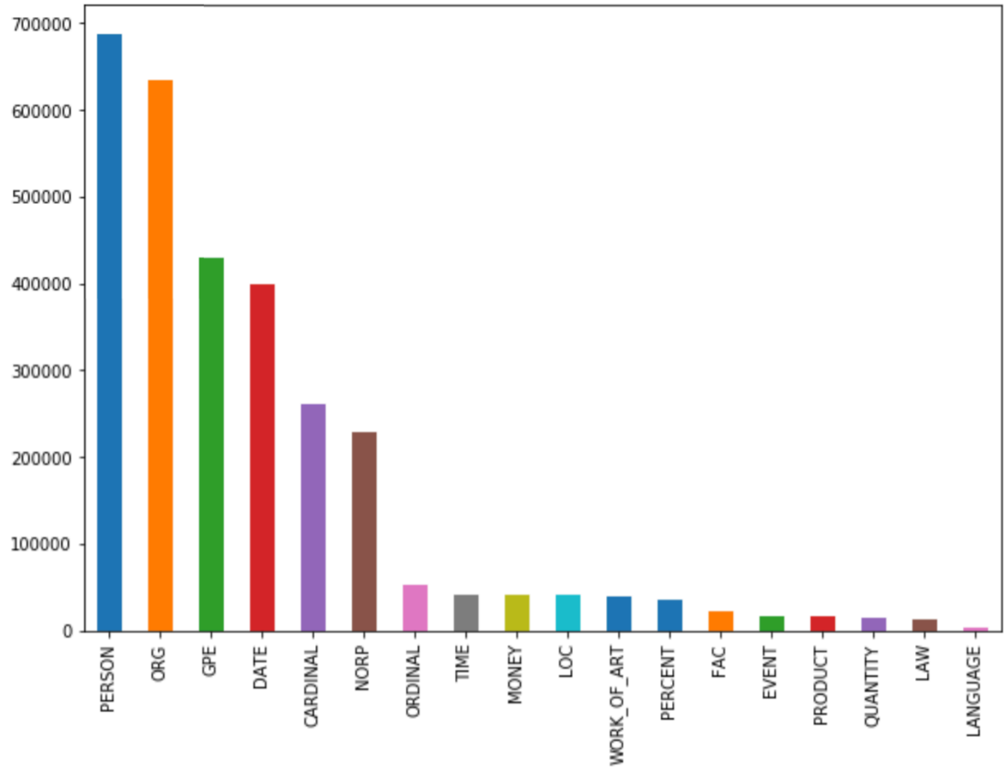
As of v2.0, spaCy supports models trained on more than one language, in this case is English. This is especially useful for named entity recognition.
To list a few examples: (Annotation Specifications https://spacy.io/api/annotation#named-entities)



TYPE	DESCRIPTION
PERSON	People, including fictional.
NORP	Nationalities or religious or political groups.
FAC	Buildings, airports, highways, bridges, etc.
ORG	Companies, agencies, institutions, etc.
GPE	Countries, cities, states.
LOC	Non-GPE locations, mountain ranges, bodies of water.
PRODUCT	Objects, vehicles, foods, etc. (Not services.)
EVENT	Named hurricanes, battles, wars, sports events, etc.

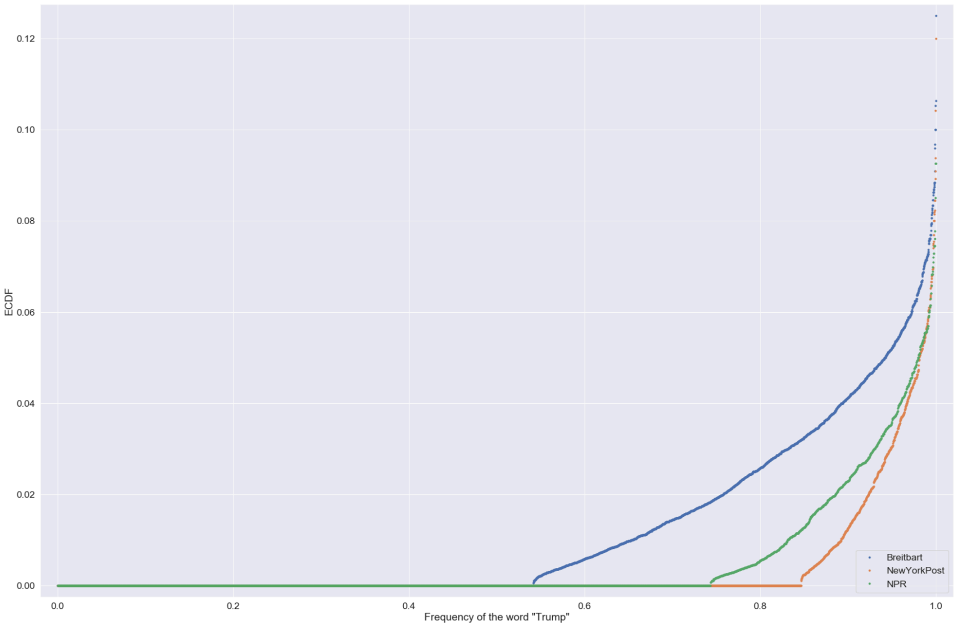
I first looked at the distribution of the number of the articles from different publications
And found out Breitbart, New York Post and NPR has the most number of news articles, so I use them for the target of comparison.

After that I observed the distribution of types of predefined words by spaCy,
Which is listed below:



And I decided to look at some most occurring words in each media source.
From Breitbart: trump, state, people, president, news
From New York Post: year, time, people, day, trump
From NPR: people, year, trump, time, state

Since ‘Trump’ in all three medias, but the frequencies of occurrence are different, I want to observe if such difference are statistically significant to notice such fact.



Statistical analysis

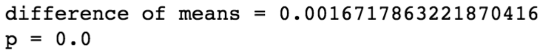
We can see that the differences of the occurring frequency of the word 'trump' from three different publications are much clearer in the ECDF. Breitbart has almost half of the articles contains the word 'trump'.

New York Post and NPR has more similar frequency for the word 'trump', if we can conclude that the frequency difference between them are statistically significant, then we can also conclude the difference between these two(New York Post, NPR) and Breitbart are much more statistically significant given its wider margin.

Preform hypothesis test

H0: p = 0 H1: p ≠ 0
Null hypothesis:
There is no difference between the occurring frequency of the word 'trump' between New York Post and NPR (frequency difference is 0, p = 0)

Alternate hypothesis:
There is difference between the occurring frequency of the word 'trump' between New York Post and NPR (frequency difference is not 0, p ≠ 0)



```
difference of means = 0.0016717863221870416
p = 0.0
```

We get a p-value of 0.0, which suggests that there is a statistically significant difference. We got a difference of 0.00167 between the means. You should combine this with the statistical significance. The difference of 0.00167 in frequency is substantial.

Cluster modeling

Create the Dictionary and Corpus needed for Topic Modeling

The two main inputs to the LDA topic model are the dictionary(id2word) and the corpus.
To list, to show a part of the corpus:

```
[[(0, 1), (1, 1), (2, 1), (3, 1), (4, 1), (5, 1), (6, 1), (7, 1), (8,
1), (9, 1), (10, 1), (11, 1), (12, 1), (13, 1), (14, 3), (15, 1), (16,
1), (17, 1), (18, 1), (19, 2), (20, 3), (21, 1), (22, 2), (23, 1),
(24, 1), (25, 1), (26, 1), (27, 1), (28, 1), (29, 1), (30, 1), (31,
1), (32, 1), (33, 1), (34, 1), (35, 1), (36, 1), (37, 1), (38, 1),
(39, 2), (40, 1), (41, 1), (42, 1), (43, 1), (44, 1), (45, 1), (46,
2), (47, 1), (48, 1), (49, 1), (50, 1), (51, 1), (52, 1), (53, 1),
(54, 2), (55, 1), (56, 2), (57, 7), (58, 1), (59, 1), (60, 1), …
```

Gensim creates a unique id for each word in the document. The produced corpus shown above is a mapping of (word_id, word_frequency). For example, (0, 1) above implies, word id 0 occurs once in the first document. Likewise, word id 14 occurs three times and so on. This is used as the input by the LDA model. To see what word a given id corresponds to, pass the id as a key to the dictionary.

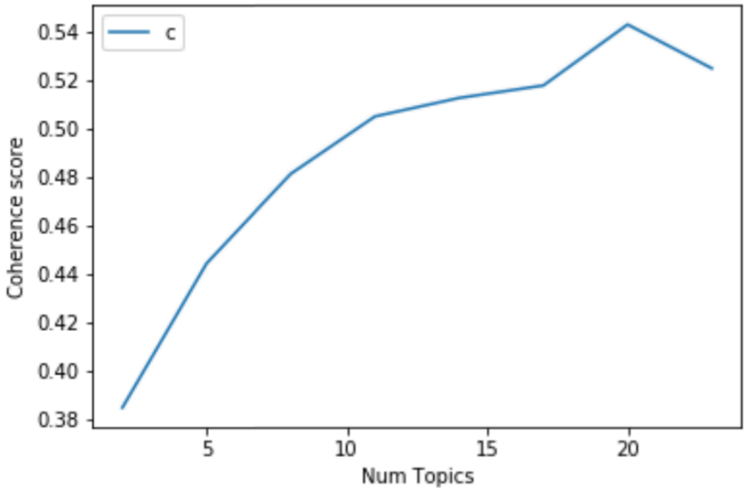
You can see a human-readable form of the corpus itself.

```
[[('ability', 1),
  ('accomplished', 1),
  ('acknowledge', 1),
  ('address', 1),
  ('adviser', 1),
  ('advocacy', 1),
  ('aim', 1), … ]…[…]]
```

How to find the optimal number of topics for LDA

My approach to finding the optimal number of topics is to build many LDA models with different values of number of topics (k) and pick the one that gives the highest coherence value. Choosing a ‘k’ that marks the end of a rapid growth of topic coherence usually offers meaningful and interpretable topics. Picking an even higher value can sometimes provide more granular sub-topics.
If you see the same keywords being repeated in multiple topics, it’s probably a sign that the ‘k’ is too large.
The compute_coherence_values() trains multiple LDA models and provides the models and their corresponding coherence scores.

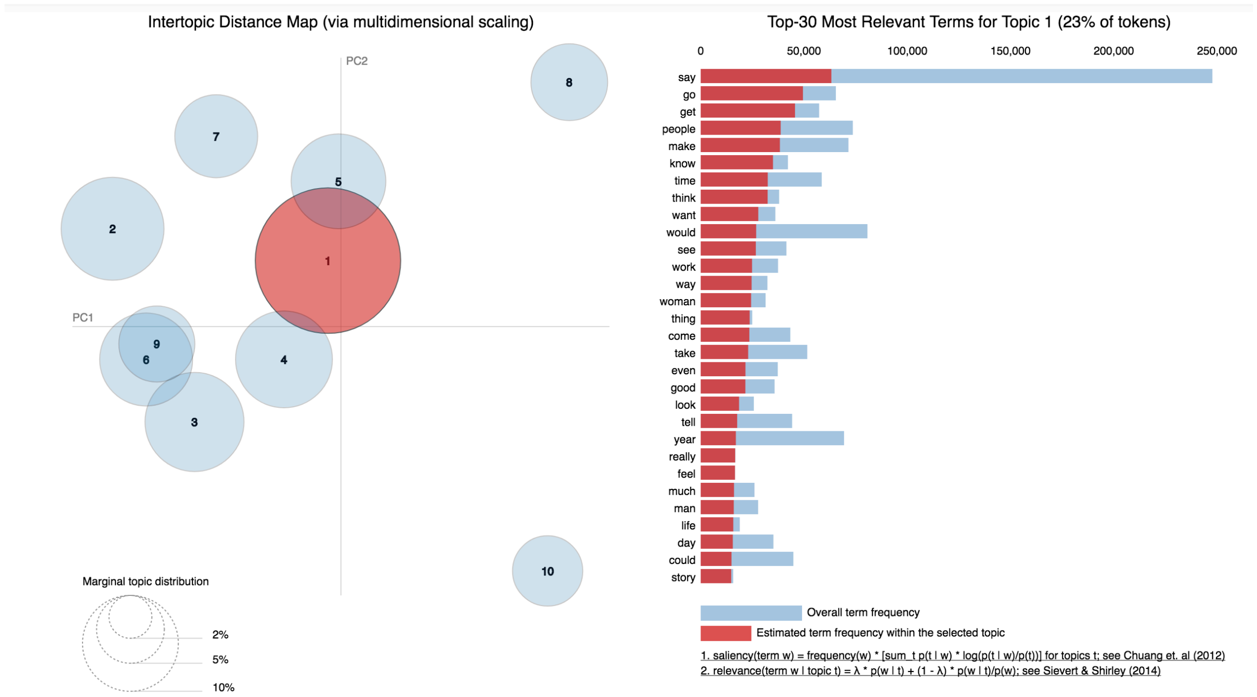
(add what is coherence scores)



If the coherence score seems to keep increasing, it may make better sense to pick the model that gave the highest CV before flattening out. This is exactly the case here.
So for further steps I will choose the model with 10 topics itself.

Visualize the topics-keywords

Now that the LDA model is built, the next step is to examine the produced topics and the associated keywords. There is no better tool than pyLDAvis package’s interactive chart and is designed to work well with jupyter notebooks.



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

	Document_No	Dominant_Topic	Topic_Perc_Contrib	Keywords	Text
0	0	0.0	0.1914	make, woman, people, show, time, life, thing, ...	[ivanka, trump, trouble, convince, world, fath...
1	1	1.0	0.3047	american, people, policy, president, bill, hou...	[jar, regulatory, action, week, large, bank, b...
2	2	2.0	0.5457	make, company, facebook, time, year, system, w...	[tale, planet, begin, year, ago, planet, earth...
3	3	0.0	0.6515	make, woman, people, show, time, life, thing, ...	[british, costumer, sandy, powell, already, os...
4	4	3.0	0.3391	country, attack, government, military, war, st...	[king_salman, push, aside, crown_prince, nayef...
5	5	6.0	0.6002	percent, company, year, business, market, pay,...	[dollar, steady, monday, recover, friday, loss...
6	6	3.0	0.1893	country, attack, government, military, war, st...	[thursday, sovereign, citizen, temporarily, en...
7	7	5.0	0.4234	trump, president, news, house, report, white, ...	[may, avoid, criminal, charge, sear, rebuke, e...
8	8	9.0	0.4120	trump, republican, campaign, party, vote, clin...	[come, house, democrat, retreat, friday, heapi...
9	9	5.0	0.3829	trump, president, news, house, report, white, ...	[document, release, friday, federal, bureau, i...

Now we can see the data has been divided into 10 groups, from the keyword we are able to tell the theme of each topic.

Topic 1: General News
Topic 2: Election
Topic 3: Sports
Topic 4: Education & Social work
Topic 5: Legal issues
Topic 6: Finance & Economy
Topic 7: Crime
Topic 8: Health & Environment
Topic 9: Foreign affairs
Topic 10: Entertainment