Topic Modeling for Economic News Articles

Image Source: https://www.pixeden.com/psd-mock-up-templates/daily-newspaper-psd-mockup

Springboard Data Science Career Track Capstone Project II

Rueben Yang

1. Introduction

This project deals with the problem of building a machine learning model for news article classification – clustering the articles into a number of groups based on the underlying topic of each article. Therefore, the clients this project serves are the new agencies. The dataset used in this project is obtained from the *Data For Everyone Library*[1], which contains 8000 news articles related to economics ranging from 1951 to 2014. The sources of the articles are Wall Street Journal and Washington Post. Topic modeling is used to classify the economic articles into different categories and determine the membership of them by discovering the structure in the corpus.

2. Data Preprocessing

Preprocessing the text data is the essential step in building a good topic model. The following steps are performed:

1) Remove numbers and special characters; all the remaining words are transformed to lowercase.
2) Remove superfluous content – at the beginning of some articles, there is author information (e.g. "*Author: Author Name*"), news source (e.g. "The Wall Street Journal Online"), or location information of the news (e.g. New York or Washington, which is often reiterated in the news content).
3) Lemmatize the tokens to group the inflected forms of a word into a single term. A lemmatizer is used in this case instead of a stemmer since the words after lemmatization remain readable.
4) To use unigrams in topic modeling, only the nouns are kept as they are more useful in revealing the subject of the articles compared with the other forms of words, for instance, verbs and adjectives.
5) The set of English stop words from *nltk.corpus* are filtered out from the list of tokens. After step 4, most of the stop words have already been removed, and this step ensures that the remaining tokens do not contain stop words.
6) To further refine the quality of the remaining tokens, the 50 most frequent words in the collection of these 8000 articles are summarized and reviewed by me:

No.	Word	Occurrence	No.	Word	Occurrence	No.	Word	Occurrence
1	year	10189	2	market	8498	3	rate	8151

[1] https://www.crowdflower.com/data-for-everyone/

4	stock	7669	5	price	5592	6	economy	4097
7	interest	3985	8	bank	3978	9	month	3794
10	company	3762	11	week	3526	12	percent	3488
13	inflation	3256	14	investor	3048	15	time	2988
16	point	2952	17	government	2831	18	dollar	2818
19	president	2806	20	tax	2723	21	bond	2690
22	yesterday	2621	23	growth	2581	24	day	2567
25	reserve	2463	26	share	2454	27	fund	2439
28	state	2399	29	index	2344	30	business	2326
31	york	2296	32	quarter	2194	33	money	2148
34	increase	2106	35	job	1984	36	consumer	1929
37	fed	1904	38	report	1894	39	deficit	1869
40	policy	1855	41	sale	1845	42	economist	1805
43	trading	1797	44	analyst	1768	45	budget	1751
46	term	1750	47	today	1713	48	people	1699
49	security	1646	50	dow	1639			

Table 1: Word Count for the Top 50 Words

The following list of words from the 50 most popular words are excluded as they are considered superfluous in delivering the subjects:

[*'day', 'today', 'yesterday', 'week', 'month', 'quarter', 'time', 'percent', 'rate', 'point', 'economy', 'economist', 'growth', 'increase', 'york', 'report', 'analyst', 'term', 'people'*]

7) Eventually, rare words that appear in less than 5 articles and frequent words that appear in more than 50% of the articles are filtered out. On top of step (6), this step ensures the common words (based on document frequency) are removed.

After applying the above steps, 5293 words remain in the resulting dictionary.

3. Methodology

David M. Blei, Andrew Y. Ng, and Michael I. Jordan proposed latent Dirichlet allocation (LDA), a generative probabilistic model for collections of discrete data such as text corpora[2]. LDA is a hierarchical

[2] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research 3* (2003) 993-1022.

Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics.

A high-level illustration is provided in this section. Consider the following set of sentences:

1. NFL playoff picture: How the field looks right now
2. Projected 2018 NFL draft order: Browns take two-game lead in race for No. 1 pick
3. FIFA World Cup: Which teams do you think will make it to the round of 16?
4. The Senate tax bill would allow oil drilling in Alaskan wilderness
5. Trump slams DOJ and FBI in weekend tweetstorm

LDA is a way of discovering topics under which these sentences can be grouped and each topic has a list of words associated with it. For the given sentences, if we ask for 2 topics, LDA might produce the following outcomes:

- **Sentences 1, 2, and 3**: 100% topic A, i.e. these sentences are solely topic A.
- **Sentences 4 and 5**: 100% Topic B, i.e. these sentences are solely topic B.
- **Topic A** is implicitly defined by the words: "NFL", "playoff", "draft", "FIFA", "World", and "Cup". One could interpret topic A to be "Sports".
- **Topic B** is implicitly defined by the words: "Senate", "Trump", "DOJ", "FBI". One could interpret topic B to be "Politics".

It is worth noting that one must figure out what the topics refer to as LDA only provides lists of words that implicitly define the topics but does not automatically label them explicitly.

To remedy the problem that topic models give no guarantee on the interpretability of their output, measuring coherence of topics has been recently studied by some researchers. Michael Röde, Andreas Both, and Alexander Hinneburg have provided a comprehensive discussion on the concept known as "topic coherence"[3]. Additionally, Palmetto[4] is a tool for measuring the quality of topics, which provides high-level descriptions for popular coherence measures. In this project, coherence metrics know as C_v and C_{Umass} are used. The definitions offered by the Palmetto website are transcribed here:

[3] M. Röde, A. Both, and A. Hinneburg. Exploring the Space of Topic Coherence Measures. WSDM'15, pages 399-408, 2015.
[4] http://palmetto.aksw.org/palmetto-webapp/

- "*Cv* is based on a sliding window, a one-set segmentation of the top words and an indirect confirmation measure that uses normalized pointwise mutual information (NPMI) and the cosine similarity. This coherence measure retrieves co-occurrence counts for the given words using a sliding window of size 110. The counts are used to calculate the NPMI of every top word to every other top word, thus, resulting in a set of vectors – one for every top word. The one-set segmentation of the top words leads to the calculation of the similarity between every top word vector and the sum of all top word vectors."
- "C_{Umass} is based on document co-occurrence counts, a one-preceding segmentation and a logarithmic conditional probability as confirmation measure. The main idea of this coherence is that the occurrence of every top word should be supported by every preceding top word. Thus, the probability of a top word to occur should be higher if a document already contains a higher order top word of the same topic. Therefore, for every word, the logarithm of its conditional probability is calculated using every other top word that has a higher order in the ranking of top words as condition. The probabilities are derived using document co-occurrence counts."

4. Results

This project applies *genism*[5], an open-source topic modeling toolkit implemented in Python.
Two parameters in *gensim*'s *LdaModel* are tuned – *num_topics* (the number of requested latent topics to be extracted from the corpus) and *iterations* (the number of steps the variational inference is allowed without convergence). The candidates for *iterations* are 100, 300, and 500; and any integer in the closed interval [1, 30] for *num_topics*. The entire dataset is used for training to produce classifications for all 8000 articles.

4.1 C_{Umass}

The C_{Umass} scores are plotted in **Figure 1**.

[5] https://radimrehurek.com/gensim/

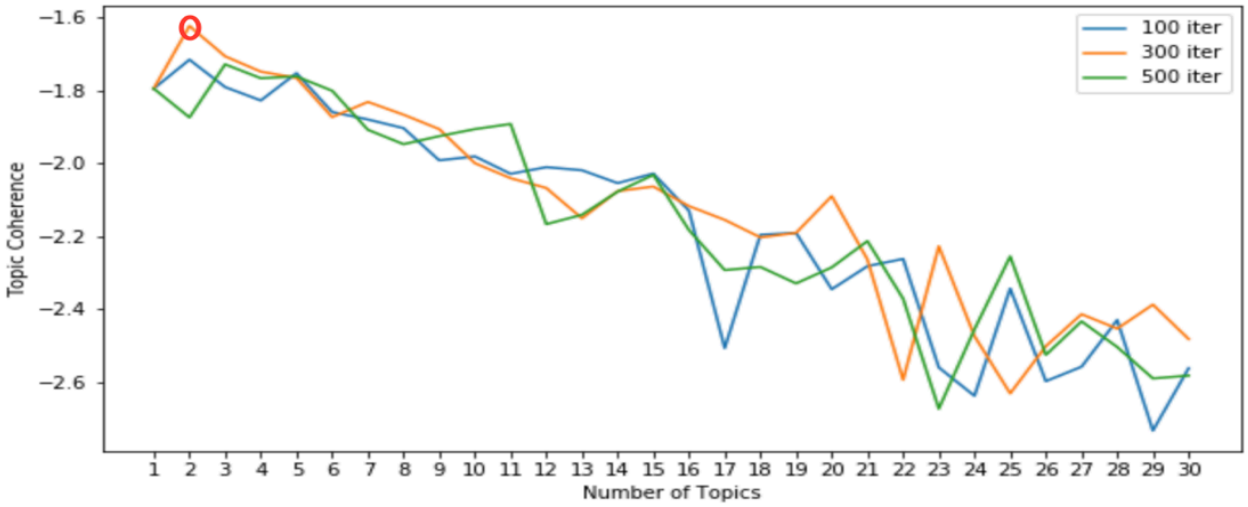

Figure 2: Topic Coherence based on C_{Umass}

The maximum C_{Umass} score is -1.623, which is attained at *iteration* = 300 and *num_topic* = 2. Hence, for the topic model based on C_{Umass}, the optimal parameters are 300 iterations and 2 topics. The top 10 words for each topic are shown in **Table 2**.

Topic 1	Topic 2
market	tax
stock	president
price	state
interest	government
investor	budget
company	job
bond	house
share	deficit
dollar	bank
index	administration

Table 2: Top Words based on the C_{Umass} model

There are no overlaps between the two sets of top 10 words. One possible interpretation for the topics would be that topic 1 is related to the financial market and topic 2 is related to macro economy.

For each given document, the topic distribution as a list of tuples (*topic_id, topic_probability*) is provided by *genism*. Each article is labeled as the topic that yields the maximum topic probability for the article and the number of articles for each topic is shown in **Table 3**.

Topic ID	1	2
Count	4363	3637

Table 3: Classification of Articles based on the C_{Umass} model

4.2 *Cv*

The *Cv* scores are plotted in **Figure 2**.

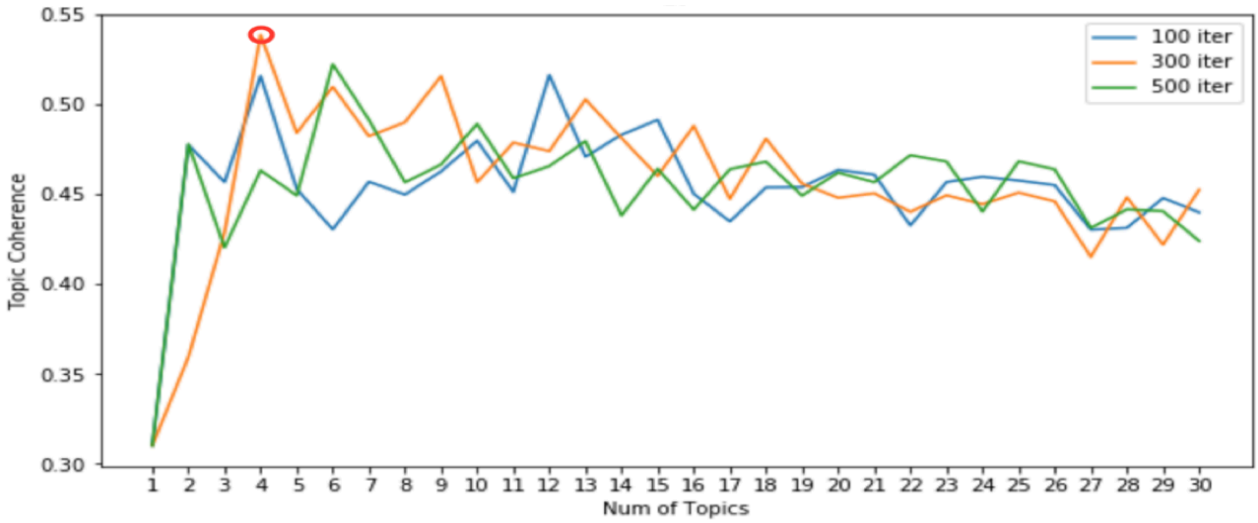

Figure 3: Topic Coherence based on *Cv*

The maximum *Cv* is 0.538, which is attained at *iteration* = 300, and *num_topic* = 4. Hence, for the topic model based on *Cv*, the optimal parameters are 300 iterations and 4 topics. The top 10 words for each topic are shown in **Table 4**.

Topic 1	Topic 2	Topic 3	Topic 4
tax	stock	company	bank
state	market	sale	interest
president	price	business	reserve
job	index	share	fed
budget	investor	industry	fund
government	dollar	firm	loan
deficit	trading	home	market
unemployment	dow	corp	money
house	bond	service	treasury
administration	share	inc	mortgage

Table 4: Top Words based on the *Cv* model

Topic 1 is quite similar to topic 2 from the C_{Umass} model, which is related to macro economy; topic 2 seems to be related to the financial market; for topic 3, one might consider it as corporate news; topic 4 could be related to the housing market and mortgage. **Table 5** shows the number of articles for each topic.

Topic ID	1	2	3	4
Count	2573	2689	1358	1380

Table 5: Classification of Articles based on the *Cv* model

4.3 Evaluation

Given that topic modeling is unsupervised learning, the model evaluation becomes tricky as there are no expected labels that one can compare the predicted labels to. In this project, a somewhat manual approach is used to evaluate the quality of the topic models where the "ground truth" is considered to be the classification made by the human:

1. Randomly select one article from the dataset.
2. Read the picked article and list the 10 keywords from my own perspective. Denote this set as H and treat it as the "ground truth".
3. Classify the article using the *Cv* model and denote the list of 10 keywords as A.
4. Classify the article using the C_{Umass} model and denote the list of 10 keywords as B.
5. Compute *rankA* as the size of the intersection between H and A, and compute *rankB* as the size of the intersection between H and B.
6. The better model (either *Cv* or C_{Umass}) is the one with rank equals to max(*rankA*, *rankB*), i.e. the one has a larger overlap of keywords with H.

The article with ID 2587 was selected from the dataset. The content of the article is shown as follows and the highlighted portion is what I considered important phrases.

Article #2587:

A widely expected Federal Reserve increase in short-term rates sent bond prices falling and knocked stock prices down from their highs to mixed levels. The dollar was mixed. The bond market, practically dormant through much of the morning, began churning once the Fed said at midafternoon that it had raised interest rates. But as economists read through the Federal Open Market Committee statement on the move, bond prices began to slump, especially among shorter-term issues. Some economists noted that not only did the Fed increase the federal funds rate by 0.50%, but also the discount rate by the same margin. The fed funds rate is the bank overnight lending rate and the discount rate is the rate the Fed charges from its discount window. In addition to the rate boosts, the Fed termed signs of slowing as "tentative" and hinted that the economy remains quite vigorous, even as it raised rates for the seventh time in about one year. The language that accompanied the Fed's move, especially about inflation and capacity utilization, along with the unanimous vote, telegraphed a certain sense of concern to the bond market," said Frazier Evans, senior economist for Colonial Investment Services. In the stock market, a mild rally in cyclical issues rapidly escalated into a fierce argument between the two warring constituencies seeking control of the stock market. During the past two sessions, more than 800 million shares have traded hands as investors have wrestled with the recession question.

After reading the article, I came up with my list of the 10 keywords and it is shown in the leftmost column of **Table 6**. The topic probabilities for Article #2587 produced by the two models are shown in **Table 7** (the maximum probabilities are highlighted). For the *Cv* model (300 iteration and 4 topics), the article is classified as its second topic which shares 7 overlapping words with my list (i.e. *rankA* = 7), and the other topics (first, third, and fourth topic) share 0, 1, and 4 words with my list, respectively. The C_{Umass} model (300 iteration and 2 topics) classifies the article as its first topic which shares 8 common words with my list (i.e. *rankB* = 8) and its second topic doesn't share any overlap with my list. The common words found by both models are highlighted in **Table 6**. Since *rankB* > *rankA*, the C_{Umass} model is better than the *Cv* model in classifying Article #2587.

My keywords	*Cv* (Topic 2)	C_{Umass} (Topic 1)
federal	stock	market
reserve	market	stock
bond	price	price
price	index	interest
stock	investor	investor
market	dollar	company
dollar	trading	bond
interest	dow	share
inflation	bond	dollar
share	share	index

Table 6: My Keywords and Top Words from the Most Similar Topics

C_{Umass} model (300 iteration and 2 topics)		*Cv* model (300 iteration and 4 topics)	
Topic ID	Probability	Topic ID	Probability
1	0.894	1	0.051
2	0.106	2	0.521
		4	0.424

Table 7: Topic Probabilities

Based on **Table 7**, the C_{Umass} model classifies Article #2587 as topic 1 with great assurance as the probability for topic 1 is remarkably above the other topic, whereas *Cv* model places the article under its topic 2 with less confidence as the probabilities it yields for topic 2 and topic 4 are not hugely distant from each other.

5. Conclusion

Based on the one-article experiment, both topic models built in this project provide article characterizations based on extracted keywords that have nonempty intersections with keywords extracted by one human reader, and hence, the two topics models show a positive promise of potentially good performance in the context of the practical business problem. Furthermore, the C_{Umass} model achieves a better performance in classifying the random selected article (Article #2587) as it finds more common words with the keywords extracted by human. Since the numbers of the topics (2 and 4) from the two models are significantly smaller than the number of possible words, the dimensionality of the articles is reduced materially.

6. Recommendations for Clients

The preliminary results show that LDA, combined with coherence metrics C_{Umass} and Cv, indeed produces an implicit classification of a given set of articles. Therefore, the target clients, e.g. Wall Street Journal and Washington Post, can consider using the data preprocessing steps and models presented in this project to reduce the complexity of a large set of articles, through a significantly smaller set of abstract topics with their associated keywords. However, the actual effectiveness of this approach with respect to human interpretability remains to be systematically tested.

7. Future Work

1) The evaluation process could be implemented more comprehensively by asking multiple people reading various articles in the dataset and comparing the keywords that are collaboratively determined with the words from the topics produced by the models. This will strengthen the robustness of the evaluation process as it aims to reduce the bias.
2) There are other measures of topic coherence that could be used to train and evaluate the topic models, e.g. *Cp*, *UCI*, *NPMI*, etc.
3) Use n-grams so that the order of words and phrases can help capture the articles' topics.
4) Consider applying other algorithms in topic modeling, e.g. Latent Semantic Indexing (LSI) and Hierarchical Dirichlet Process (HDP).