Motivation:

The purpose of this project is to apply web scraping and text analytics in solving real life problem related to business and finance. When an analyst is required to conduct a competitive analysis, it is common for him/her to research on the news of targeted companies, markets or industries, in order to see if there is any past and upcoming M&A, JV or partnerships. Unfortunately, it takes a lot of time to read every single news article and most of them are usually unrelated. Therefore, I would like to implement topic modeling here to classify news topic, which could create 2 benefits. One is knowing the main idea of a news article by just looking at the topic and news title, and the second is classifying news article into groups in order to save reading time.

Scraping Textual Data:

In this project I am interested in the technology infrastructure industry, which is related to data center business, cloud computing and IT solutions. All the text were extracted from the website https://datacenternews.asia/. As the HTML structure of this website is static, I used Python library ***BeautifulSoup*** to do the web scraping. The program goes through the first 300 pages of the news and about 7200 news articles are stored into ***Pandas*** dataframe with the corresponding news title, content and url. I leveraged library ***Concurrent.futures*** to perform multi-threading for speeding up the scraping process, reducing running time from 30 minutes to 4 minutes.

Textual Data Processing:

After creating the nice dataframe using ***Pandas***, I cleaned the unnecessary symbols and turned all letters to lowercase. Then, I tokenized each word to a string and lemmatized them using library ***SpaCy*** with only noun. It is because when all part of speech is included, the topic keys are not efficient as some words like 'say', 'use', 'look' will appear. Martin & Johnson (2015) concluded that removing all words except nouns could improve topics' semantic coherence[i]. After that, I made bigrams for possible adjacent words and cleaned the stop words using ***Gensim***. Finally, the whole list of tokens is turned into a corpus.

LDA Modeling:

In this project I applied Latent Dirichlet Allocation (LDA) to classify news articles to topics. First of all, in order to know the optimal parameter, which is the number of topics to be created, I calculated coherence values for each model with different number of topics. It turned out 8 and 16 topics gave us higher values so 8 topics are picked for simplicity. Therefore, putting corpus into the LDA model will give me the most suitable topic for each news article. I appended the data frame with topic, coherence contribution and topic key words. Now, each news article is assigned to a topic with key words attached.

Conclusion:

The result looks good and topics are well separated but possible improvement is expected in the future, as the overall coherence value is only around 0.45. Sometimes keywords between topics overlap with each other and have similarity. For further improvement, I will consider adding weighted tags and weighted keywords (like adding weight to key word 'acquisition'), which would be useful to improve the overall coherence values, when we know which terms are more important and should carry higher weights.

---

[i] Martin, F., & Johnson, M. (n.d.). More efficient topic modelling through a noun only approach. ACL Anthology. Retrieved March 24, 2022, from https://aclanthology.org/U15-1013/