Deadline: **Saturday** 1 October (“end of day of class”)

*Proposal*

Each group is required to submit a proposal (up to 3 pages) to indicate the data set to be used and the intended tasks and analyses. Use some use cases to illustrate why you believe the analytics tasks you intend to perform are useful. You can add unlimited number of pages for appendix.

Note: The proposal counts towards 5% of the final grade.

*List of Analytics Tasks*

1. Document Retrieval
2. Document Classification
3. Document Clustering
4. Topic Analysis
5. Information Extraction
6. Sentiment Analysis
7. You can propose any other text mining or NLP tasks and the corresponding details. Examples: Question answering, summarization, discourse analysis for relationship extraction, semantic analysis in social media data etc.,

**Shortlisted**

1. Movie character emotion

* Build general code that could be applied across movies or
  + possible to look at movies with sequels e.g., avengers, harry potter
  + or movies with “strong” characters (in terms of emotion)

2. Political speech analysis

* Topic Analysis or Document summarization to get a summary of the speech
* Retrieve the topics or most frequent words and perform Sentiment Analysis on Twitter by crawling tweets that contain these topics/frequent words

For 2022 speech – manually tag the topics, to do a comparison of speaker’s sentiment vs public sentiment on a topic level

For other speeches, do sentiment analysis on para level -> create heat map

3. Fake reviews detection [Backup]

* Classification of product / restaurant reviews and identify fake reviews
* Sentiment analysis on real reviews

**--------**

**Ideas for Project so far:**

1. Scam Detection

2. Political Speech analysis

3. Boring money-making topic: Stock price prediction based on news and tweets

4. Review analysis

-> To know the positive and negative points of a product so that a company can focus and improve

[Amelia's ideas]

**5. Sentiment Analysis**

**Web data: Amazon Fine Foods reviews** <https://snap.stanford.edu/data/web-FineFoods.html>

This dataset consists of reviews of fine foods from [amazon](http://www.amazon.com/). The data span a period of more than 10 years, including all ~500,000 reviews up to October 2012. Reviews include product and user information, ratings, and a plaintext review.

More Amazon data (other categories) for sentiment analysis:<http://snap.stanford.edu/data/web-Amazon.html>

**6. [Document Classification] Fake reviews detection**

a. Fake Review Dataset – Amazon Reviews

**Reference**: Creating and detecting fake reviews of online products <https://www.sciencedirect.com/science/article/pii/S0969698921003374>  
Data: <https://jonisalminen.com/fake-reviews-dataset-and-generation/>

* Generated fake product reviews based on an Amazon e-commerce dataset using GPT-2 (language model)
* Data set: contains 20k **computer-generated** fake reviews and 20k human-created product reviews

b. Fake restaurant reviews

**References** (seems like a student assignment)  
Paper: <http://cs229.stanford.edu/proj2019aut/data/assignment_308832_raw/26647583.pdf>Code: <https://github.com/ganasank/CS229-Project>   
Data: <https://github.com/darshandagly/Fake-Review-Detection> (not by the same authors, but should be the same/similar dataset..)

Another source:

Data: <https://www.kaggle.com/datasets/yelp-dataset/yelp-dataset/discussion/156218>   
For citation: " got this dataset from Shebuti Rayana, and she told me to cite her papers if I plan to use this dataset in any of my projects. So **please**, if you use this data set to write a paper/thesis about your project, cite her following paper:

Collective Opinion Spam Detection: Bridging Review Networks and Metadata. Shebuti Rayana, Leman Akoglu, ACM SIGKDD, Sydney, Australia, August 10-13, 2015"

General reference papers on the subject

[**https://arxiv.org/abs/2112.14343**](https://arxiv.org/abs/2112.14343)

<https://dr.ntu.edu.sg/bitstream/10356/107209/1/a88-banerjee.pdf>

<https://scoredata.com/how-to-detect-fake-online-reviews-using-machine-learning-2/>

**7. Clustering / Topic Analysis**

Was researching and chanced upon news dataset meant for recommender task ; but likelihood is the data would be suited for clustering or topic analysis too  
<https://competitions.codalab.org/competitions/24122#learn_the_details>  
https://msnews.github.io/

[Wesley ideas]

**8. Movie character emotion analysis using subtitle text**

Some papers:

* <https://www.stanfurrer.ch/Doc/Emotion_analyse_On_OpenSubtitle.pdf>
* https://link.springer.com/chapter/10.1007/978-3-030-49161-1\_36

Some libraries that we can use

* EmoBERT / EmoRoBERTa / GoEmotions
* NRCLex (<https://www.youtube.com/watch?v=HujhnpzT24c>)

Dialog Extraction from IMDB: <https://www.youtube.com/watch?v=z2ntS7_CNpk>

Movie script database: <https://imsdb.com/>

[Zoe’s ideas]

**9. Analysis on Medium Data Science Articles**

Some possible analysis that could be done: document retrieval, document classifications (the dataset have tags), document clustering/ topic analysis to find out what are the trendy topics in the data science area

Limitation: The dataset only has URL of the articles but not the free texts.

Dataset: <https://www.kaggle.com/datasets/viniciuslambert/medium-2021-data-science-articles-dataset>

Some other possible datasets that could apply the same analytics techniques could be newspaper articles: <https://www.kaggle.com/datasets/tumanovalexander/nyt-articles-data>

**10. Employer Review Analysis (I think would be more interesting)**

Dataset: <https://www.kaggle.com/datasets/muhammedabdulazeem/employer-review-about-their-organization>

Available information: Review Title (word/phrase/short sentence), Review Body (e.g., former employee), Review Rating (numerical score), Reviewed Company (entity), Review description (free text)

Limitation: JSON file

Some possible analysis that could be done: sentiment analysis, information extraction, document retrieval

Applications: To help the company understand what employees are looking for when seeking jobs, improve on their current flaws, offer packages and work environment, learn from other popular companies in the industry. On the other hand, to help the jobseekers to understand more about the pros and cons about the companies based on the reviews and make informed decisions.

11. **Political Speech analysis – Retrieve National Day Rally Speech**

Perform Topic Analysis (Topic Modelling) or Document summarization to get a summary of the speech

Retrieve the topics or most frequent words, and perform Sentiment Analysis on social media (Twitter) by crawling tweets that contain these topics/frequent words.

**12. Report topic sentence generation or summary generation**

* Can apply to news articles or scientific reports or patents or book texts etc.
* Extractive text summarisation
  + Key sentences add to summary
  + Exact sentences ranked and selected
* Abstractive text summarisation
  + Key phrases, understand context, reproduce
  + More advanced
* Dataset

<https://paperswithcode.com/dataset/cnn-daily-mail-1>

<https://paperswithcode.com/dataset/newsroom>

<https://metatext.io/datasets-list/summarization-task>

------------------

---------

Draft section

Get corpus -> Union of BoW

# **2.** **Document Retrieval**

Steps:

* Tokenisation
* Normal text pre-processing
  + Lowercase
  + remove punctuation
  + Remove stop words
  + lemmatization/stemming
* TF, TFIDF
* Classification
  + Naïve Bayes
  + Cosine Similarity
  + Logistic Regression
* Rank and return results

----- To be deleted ----

**Document retrieval Task**

Source: Prime Minister’s Office [<https://www.pmo.gov.sg/Topics/National-Day-Rally>]

Take all languages

**By Friday**

Name your file: YYYY\_xxx.txt e.g., 2004\_eng.txt

* Years:
  + PMO: 2004 to 2022
    - Wesley: 2004 – 2008
    - Zoe: 2009 - 2013
    - Corryne: 2014 - 2018
    - Amelia: 2019 to 2022
  + **Include other languages (translated in Eng) too – in separate txt files**

**Topic Analysis**

* Split into Paragraph (nested list)
* Tokenisation
* Text pre-processing
  + Lowercase
  + remove punctuation
  + Remove stop words
  + lemmatization/stemming
* Unsupervised Learning
  + LDA
  + Clustering
    - Find best number (elbow)
* Paragraph Level Topic Modelling / Document Level Topic Modelling

1. **Sentiment Analysis**

* Split into Paragraph (nested list)
* Tokenisation
* Normal text pre-processing
  + Lowercase
  + remove punctuation
  + Remove stop words
  + lemmatization/stemming
* Analyse Speech: Sentiment (+/-)
  + VADER
  + Textblob
  + Flair
  + SpaCy
* Scrape Twitter Data
* Analyse Reactions: Sentiment (+/-) and Emotions
  + text2emotions
  + + other emotion tools

Source: National Archives [<https://www.nas.gov.sg/archivesonline/speeches/>]

* National day rally speeches every 10 years 1972 to 2022? (1972, 1982, 1992, 2002, 2012, 2022) à Jim to gather
* Combined different language speech

**-------------- Discussion -----------**

**Document Retrieval**

* Use TFIDF at the document level

**Speech Sentiment Analysis** **– Paragraph Level**

* Split the document into paragraphs using regex (instead of splitting by word or sentence)
* Result should be a nested list of paragraphs
* Use Sentiment Analyser like VADER that will assign sentiment score on word-level, and then aggregate by sentence. But in our case, our ‘sentence’ is at the paragraph level.
* Result will be a diagram like in here: <https://towardsdatascience.com/sentiment-analysis-of-political-speeches-using-hugging-faces-pipeline-feature-3109c121d351>

**Social Media Sentiment Analysis**

* Want to compare topic level sentiment from speech vs topic level sentiment from twitter
* 2022 speeches, since Twitter existed from 2006 onwards and not much in 2012

**For Topic Modelling**

* Conclusion is to try both methods
  + Single document
  + All document in a corpus
* See which methods gives a better story for our project.