Text Mining - Team 12 Document

| **About**  This can be our living document and white board. New content / ideas will be added on top.  **Pinboard**  📌Google Drive Link: <https://drive.google.com/open?id=1_DgN35hlZGwUYOvL1fgUNfQU4XpNU9EH&authuser=lennard.ong%40gmail.com&usp=drive_fs>  📌 Project Report:  <https://www.overleaf.com/project/6419879881d2a93ca999b774> |
| --- |

---

**FOLDERS**

Folders to submit

1. **data** - datasets we use for training and in various utils
2. **models** - our finetuned models
3. **results** - this is all the final summarization results
4. **Root** - all notebooks / python scripts in root

Folders to clarify

1. notebooks - no notebooks from lenn here. check with DX what should be submitted

Folders to omit

1. archive - this is just old code
2. docs - this is where our gdoc lives
3. references - this is a random assortment of academic papers

**ORDER OF NOTEBOOKS**

In general, each one below takes a folder as input

EDA

1. T5\_00\_EDA\_LO.ipynb (used for EDA of WebNLG Triplet Format)

For Text Preprocessing

1. covar\_00\_bbcBatch\_LO.ipynb (used to batch preprocess documents, replace text with covariance resolution)
2. triplets\_01\_clean\_LO.ipynb (triple generation, greedy)
3. triplets\_02\_tripleFormatting\_LO.ipynb (triple generation, strict)

For LLM Fine Tuning

All these are Kaggle notebooks

1. GPT\_00\_fineTuning.ipynb
2. T5\_01\_train8epoch.ipynb
3. T5\_01\_train16epoch.ipynb

For Text Generation

All export to a folder called “results”

1. GPT\_01\_textGenDistil\_LO.ipynb
2. GPT\_01\_textGenDistilFiltered\_LO.ipynb
3. GPT\_01\_textGenGPT2med\_LO.ipynb
4. ChatGPT\_01\_promptedTextGen.ipynb
5. T5\_02\_textGen8epoch.ipynb
6. T5\_02\_textGen16Epoch\_LO.ipynb

For Evaluation

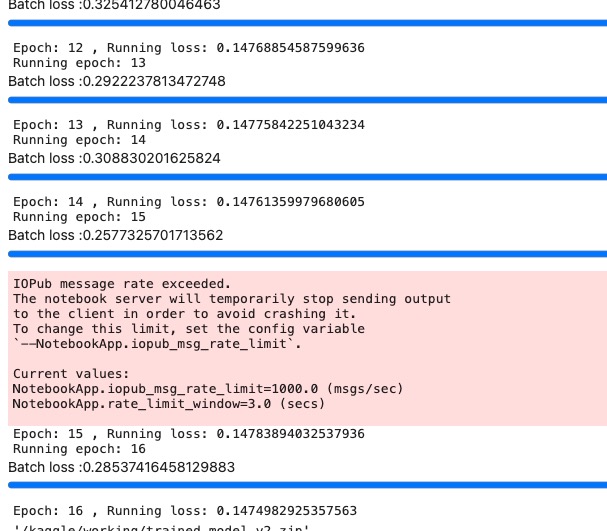
1. scoring\_BatchSimplicityIndex\_LO.ipynb
2. util\_simplicityIndex.py (this is imported into other notebooks)

Others (these can be excluded from upload)

1. triplets\_00\_experiments\_LO.ipynb
2. triplets\_03\_reportGreedyGraphics\_LO.ipynb
3. triplets\_03\_reportStrictGraphics\_LO.ipynb

# 2nd last week

T5 Training Loss



## Notes for KS:

### Outline

What is our preferred outline for the report? Please suggest (I’ve read your old reports, they’re excellent!)

1. ….
   1. ….
2. ..

Arbitrarily, I’m orgainzing by model so its easier. Sections I can writeup:

* Motivation
  + Contribute mock cases of how we think it will work. See section below, “ideal outcomes”
* Metrics
  + Will write up on simplicity index
* Distil GPT
  + Title: usage of dependency tree and entities
  + Triples: Linguistic heuristics to match schema
* Distil GPT, filtered
  + Training Data: Webnlg schema
  + Filtering: generating more sentences, filtering
* T5
  + Training Data: Webnlg schema
  + Triples: Linguistic heuristics to match schema
* Scoring
  + I can score all my models for simplicity
* Observations and Comparisons
  + Create a table comparing a few articles, observing the pros and cons of each model

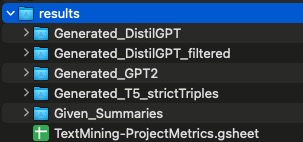
Sections I cannot writeup

* Interpreting the training results of the model.

### Scoring Metrics

* Concern: ROUGE scores will be horrible because we are not basing out training on the summaries themselves.
* Yes, rouge might be useful, but consider some other semantic similarity metric. (e.g. vectors of articles to articles or something?)

### Results & Files



* All results stored in ‘results’ folder.
* Scores should be chucked into the spreadsheet there
* Direct Link: [results](https://drive.google.com/open?id=1PSU7bSLRbyvb3Yit8O3XjlSKhy2ILzQl&authuser=lennard.ong%40gmail.com&usp=drive_fs)

| Outcome | Model & Training | Output |
| --- | --- | --- |
| Generated\_DistilGPT | Same model as below | **Data**  data/BBC/Training/business\_triples  **Notebook**  GPT\_00\_distil\_LO.ipynb  **Notes**  Generative triples are filtered to match keytwords in titles only |
| Generated\_DistilGPT\_filtered | **Training Data**  models/distilgpt2\_trained\_pkg  **Notebook**  [Kaggle](https://www.kaggle.com/code/lennong/triplets-to-sent-generator)  **Notes**  Trained on “relaxed triples”. longer triples, allowing nonverb/non-entity in Subject or Object. | **Data**  data/BBC/Training/business\_triples  **Notebook**  GPT\_01\_distilFiltered\_LO.ipynb  **Notes**  Generative triples are filtered to match keytwords in titles only  For each triple, generate 5 sentences and choose the one with closest vec similarity to title. |
| Generated\_t5\_strictTriples | **Trainig Data**  models/T5\_fulltrain\_8epoch  **Notebook**  [Kaggle](https://www.kaggle.com/code/lennong/textmining-t5)  **Notes**  Trained on WebNLG Data  For WebNLG EDA, see  T5\_00\_EDA\_LO.ipynb | **Data**  data/BBC/Training\_strict/business\_triples  **Notebook**  T5\_01\_textGen.ipynb  **Notes**  Triples are very strict, formatted to match WebNLG schema |
| Generated\_BERT |  |  |
| Generated\_ChatGPT Prompt | **Input Sys Prompt**  You are a triplet-to-paragraph generator.  #Brief:  Anything between [] is the task inputs.  # indicates the article title  "sub | veb | obj" describes a series of verb, object triples related to title.  #Task:  The task is to generate a short summary paragraph with the title at top.  It should be factually based only on the triples.  Inferences should only be made between the triples and the title.  Do not add embellishments.  Do organize the paragraph so it has a logical flow.  Keep it as simple and direct as possible.  #Example Input:  [#China now top trader with Japan  china | overtook | us  change | highlights | chinagrowing importance  trade | was hurt | factors  analysts | see | spurs  Japan trade surplus | grew | trade  Japan trade surplus | accounted | trade  #Example Output:  China now top trader with Japan  China has overtaken the US as Japan's top trading partner. This change highlights China's growing importance in the region. Trade was hurt by various factors, but analysts see this as a spur to further growth. Japan's trade surplus grew as a result, with the surplus accounting for a significant portion of the trade.  **Input User Prompt (example)**  [#Laura Ashley chief stepping down  Laura Ashley | announce | closure  Laura Ashley | found | namesake  Laura Ashley | have | UK\_factory  Laura Ashley | part | company  Laura Ashley | put | focus  Laura Ashley | thank | executive  Laura Ashley\_Laura Ashley | hit | report  Lillian\_Tan | manage | director\_executive  September\_Laura Ashley | report | focus  entrepreneur\_Dr\_Khoo\_Kay\_Peng | deny | intend  entrepreneur\_Dr\_Khoo\_Kay\_Peng | intend | take\_Laura Ashley\_private  owner | shuffle | deckchair] |  |

## Relevant Links for Krishna:

* SI scorer:
  + Utility to import: [util\_simplicityIndex.py](https://drive.google.com/open?id=17oLnB2UhwCXoB7dVsXnKkEeTqLR3iJ_f&authuser=lennard.ong%40gmail.com&usp=drive_fs)
  + Notebook to batch score based on a folder of text files: [scoring\_BatchSimplicityIndex\_LO.ipynb](https://drive.google.com/open?id=1TlJAzZOTjkLi8CMmsm_l15krO6S_Vba5&authuser=lennard.ong%40gmail.com&usp=drive_fs)
  + Scoring gsheet: <https://docs.google.com/spreadsheets/d/11SgIfcxXpyzxHjbOKskInuplvXUjJodlmO6ZiADxMO0?authuser=lennard.ong%40gmail.com&usp=drive_fs>
* Raw biz articles
  + Folder of text files: [business](https://drive.google.com/open?id=11Dnew9YYNl_F-_2ow9M3ILUnY3xuRskX&authuser=lennard.ong%40gmail.com&usp=drive_fs)
  + To generate an array and separate title from body: see [triplets\_01\_clean\_LO.ipynb](https://drive.google.com/open?id=1SayJX1liJQVlGW_bNsmh4Y9n2_FL3wY-&authuser=lennard.ong%40gmail.com&usp=drive_fs)
* Triplets

## LO - Triplet2Text Model Trained in Kaggle

* Using GPT2-med and DistilGPT2 model for speed
* Credit to krish for Kaggle tip
* <https://www.kaggle.com/lennong/triplets-to-sent-generator>

## Ideal Outcomes

LO: I note that the problem isn’t “sentence completion from triples”.

Its 2 things:

1. Inferring context from title
2. Joining triples into compound sentences

Given this, what is an model approach?

Giving a mask china [blank] overtook [blank] us does not serve the right purpose

Study Prompt (all triplets + title to body):

Anything between [] is your summary inputs.

# a article title followed by a series of subject, verb, object triples about the title.

The goal is to generate a short summary paragraph with the title at top.

It should be factually based only on the triples. Do not add embellishments or paraphrase. Keep it as simple and direct as possible. Make only simple inferences.

Example:

[# Lacroix label bought by US firm

group lvmh | has sold | lacroix label

group lvmh | has been shedding | businesses

couture labels | had been purchased | us investment group

us investment group | bought | cosmetics labels

fashion houses | are struggling | to make money from many fashion houses expensive haute couture ranges

us investment group | planned | to expand lacroix label by opening new stores

designer | planned | to stay at lacroix label the french designer 's]

Suggestion:

Total articles: ~2000

Train: 1600

Test: 400

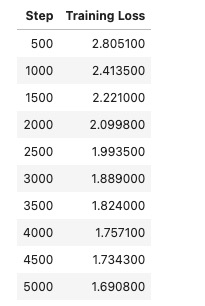
1. Generate a “ideal output” dataset using GPT3.5 api
2. Train a BERT model on it
3. Use it for

| Original Article | Example Input | Ideal Output (not from our model) | Dataset Summary  *(our rouge scores will all suck unless we build our model based on this)* |  |
| --- | --- | --- | --- | --- |
| **China now top trader with Japan**  China overtook the US to become Japan's biggest trading partner in 2004, according to numbers released by Japan's Finance Ministry on Wednesday.  China accounted for 20.1% of Japan's trade in 2004, compared with 18.6% for the US. In 2003, the US was ahead with 20.5% and China came second with 19.2%. The change highlights China's growing importance as an economic powerhouse. In 2004, Japan's imports from and exports to China (and Hong Kong) added up to 22,201bn yen ($214.6bn;£114.5bn). This is the highest figure for Japanese trade with China since records began in 1947. It compares with 20,479.5bn yen in trade with the US.  Trade with the US during 2004 was hurt by one-off factors, including a 13-month ban on US beef imports following the discovery of a cow infected with mad cow disease (BSE) in the US. However, economists predict China will become an even more important Japanese trading partner in the coming years. On Tuesday, figures showed China's economy grew by 9.5% in 2004 and experts say the overall growth picture remains strong. Analysts see two spurs to future growth as being China's membership of the World Trade Organisation and lower trade tariffs. During 2004, Japan's trade surplus grew 17.9% to 12.011 trillion yen, with more than half the surplus, 6.962 trillion yen, accounted for by its trade with the US. In December, the surplus grew 1.8% on a year ago to 1.14 trillion yen thanks to stronger-than-expected exports.  505.txt | **#China now top trader with Japan**  china | overtook | us  change | highlights | chinagrowing importance  trade | was hurt | factors  analysts | see | spurs  Japan trade surplus | grew | trade  Japan trade surplus | accounted | trade  **{#China now top trader with Japan,**  china | overtook | us}  **{#China now top trader with Japan,**  change | highlights | chinagrowing importance}  **{#China now top trader with Japan**  trade | was hurt | factors}  **{#China now top trader with Japan**  analysts | see | spurs}  **{#China now top trader with Japan**  Japan trade surplus | grew | trade}  **{#China now top trader with Japan,**  Japan trade surplus | accounted | trade} | Whole Body to Para  **China now top trader with Japan**  China has overtaken the US as Japan's top trading partner. This change highlights China's growing importance in the region. Trade was hurt by various factors, but analysts see this as a spur to further growth. Japan's trade surplus grew as a result, with the surplus accounting for a significant portion of the trade.  Sentence by sentence  China has overtaken the US as Japan's top trader. The change mentioned in the article highlights the growing importance of China. The factors hurt the trade between China and Japan. China's emergence as the top trader with Japan is being seen as a sign of China's growing importance, according to analysts.  Manual  **China now top trader with Japan**  China overtook the US. The change highlights China’s growing importance. Trade was hurt by various factors. Analysts see spurs. Japan trade surplus grew with trade. Japan trade surplus accounted for trade. | **China now top trader with Japan**  China accounted for 20.1% of Japan's trade in 2004, compared with 18.6% for the US.During 2004, Japan's trade surplus grew 17.9% to 12.011 trillion yen, with more than half the surplus, 6.962 trillion yen, accounted for by its trade with the US.It compares with 20,479.5bn yen in trade with the US.China overtook the US to become Japan's biggest trading partner in 2004, according to numbers released by Japan's Finance Ministry on Wednesday.This is the highest figure for Japanese trade with China since records began in 1947.In 2003, the US was ahead with 20.5% and China came second with 19.2%. |  |
| **India's Maruti sees profits jump**  India's biggest carmaker Maruti has reported a sharp increase in quarterly profit after a booming economy and low interest rates boosted demand.  Net profit surged 70% to 2.39bn rupees ($54.98m; £29.32m) in the last three months of 2004 compared with 1.41bn rupees a year earlier. Total sales were 30.1bn rupees, up 27% from the same 2004 period. Maruti accounts for half of India's domestic car sales, luring consumers with cheap, fuel-efficient vehicles.  Demand in India also has been driven by the poor state of public transport and the very low level of car ownership, analysts said.  Figures show that only eight people per thousand are car owners. Maruti beat market expectations despite an increase in raw materials costs. The company, majority-owned by Japan's Suzuki, said an increase in steel and other raw material prices was partially offset by cost cutting. Sales in the fiscal third quarter, including vans and utility vehicles, rose by 17.8% to 136.069 units. Maruti is not the only company benefiting as Indian's economic growth gives consumer greater spending power. Utility vehicle and tractor maker Mahindra has reported a 52% rise in net profit during the last three months of 2004. Profit was 1.33bn rupees compared with 874.2m rupees a year earlier.  503.txt | **#India's Maruti sees profits jump**  India's biggest carmaker | has reported | increase  India's biggest carmaker | boosted | demand  demand | has been driven | state level  increase | was offset | cost cutting  company | gives | spending power  mahindra | has reported | % rise | **India's Maruti sees profits jump**  India's biggest carmaker has reported an increase in profits, boosted by demand that has been driven at the state level. The increase was offset by cost cutting, but the company has given confidence in its spending power. The carmaker's success comes as rival Mahindra has also reported a % rise in profits. | **India's Maruti sees profits jump**  Net profit surged 70% to 2.39bn rupees ($54.98m; £29.32m) in the last three months of 2004 compared with 1.41bn rupees a year earlier.Profit was 1.33bn rupees compared with 874.2m rupees a year earlier.India's biggest carmaker Maruti has reported a sharp increase in quarterly profit after a booming economy and low interest rates boosted demand.Maruti accounts for half of India's domestic car sales, luring consumers with cheap, fuel-efficient vehicles.Utility vehicle and tractor maker Mahindra has reported a 52% rise in net profit during the last three months of 2004. | Given an input that consists of a title and subject-verb-object triples, I would like to generate a 1 paragraph summary.  It should be factually based only on the triples. Do not add embellishments or paraphrase. Keep it as simple and direct as possible. Make only simple inferences.  I would like to implement this using a LLM. Training data and compute is relatively limited.   What approaches would a data scientist consider?  Example Input:  **India's Maruti sees profits jump**  India's biggest carmaker | has reported | increase  India's biggest carmaker | boosted | demand  demand | has been driven | state level  increase | was offset | cost cutting  company | gives | spending power  mahindra | has reported | % rise  Example Output:  **India's Maruti sees profits jump**  India's biggest carmaker has reported an increase in profits, boosted by demand that has been driven at the state level. The increase was offset by cost cutting, but the company has given confidence in its spending power. The carmaker's success comes as rival Mahindra has also reported a % rise in profits. |
| **Parmalat bank barred from suing**  Bank of America has been banned from suing Parmalat, the food group which went bust in 2003 after an accounting scandal.  The bank - along with investors, auditors and the group's managers - wants damages for being a victim of fraud at the hands of the Italian firm. But a judge has barred Bank of America and two auditors from the case. The bank, and Italaudit - formerly the Italian arm of auditor Grant Thornton - face lawsuits and possible prosecution. A second auditor, Deloitte & Touche, has also been banned from the case. Grant Thornton - now rid of the Italian unit at the centre of the case - is still being permitted to sue, as are Consob, Italy's stock market regulator, hundreds of small investors and Parmalat's new managers. Parmalat collapsed in December 2003 after it emerged that the 4bn euros ($5.2bn; £2.8bn) it supposedly held in a Bank of American offshore account did not in fact exist.  492.txt | **Parmalat bank barred from suing**  bank along | wants | damages  judge | has barred | bank auditors  auditor grant thornton | is being permitted | to sue , | **Parmalat bank barred from suing**  The bank is not allowed to sue for damages, as a judge has ruled that only the auditors, Grant Thornton, can sue. | **Parmalat bank barred from suing**  But a judge has barred Bank of America and two auditors from the case. The bank, and Italaudit - formerly the Italian arm of auditor Grant Thornton - face lawsuits and possible prosecution.A second auditor, Deloitte & Touche, has also been banned from the case. |  |
| **353** | hurricane mitch | caused | damage  damage | covers | fishing farming businesses  thousands | destroyed | giant waves  insurers | are struggling | to assess the cost of the damage  players | believe | to assess the cost of the damage |  |  |  |

Training Notes

DistilGPT2

TrainOutput(global\_step=5112, training\_loss=2.0347088230234545, metrics={'train\_runtime': 307.4895, 'train\_samples\_per\_second': 49.823, 'train\_steps\_per\_second': 16.625, 'total\_flos': 500383277383680.0, 'train\_loss': 2.0347088230234545, 'epoch': 8.0})

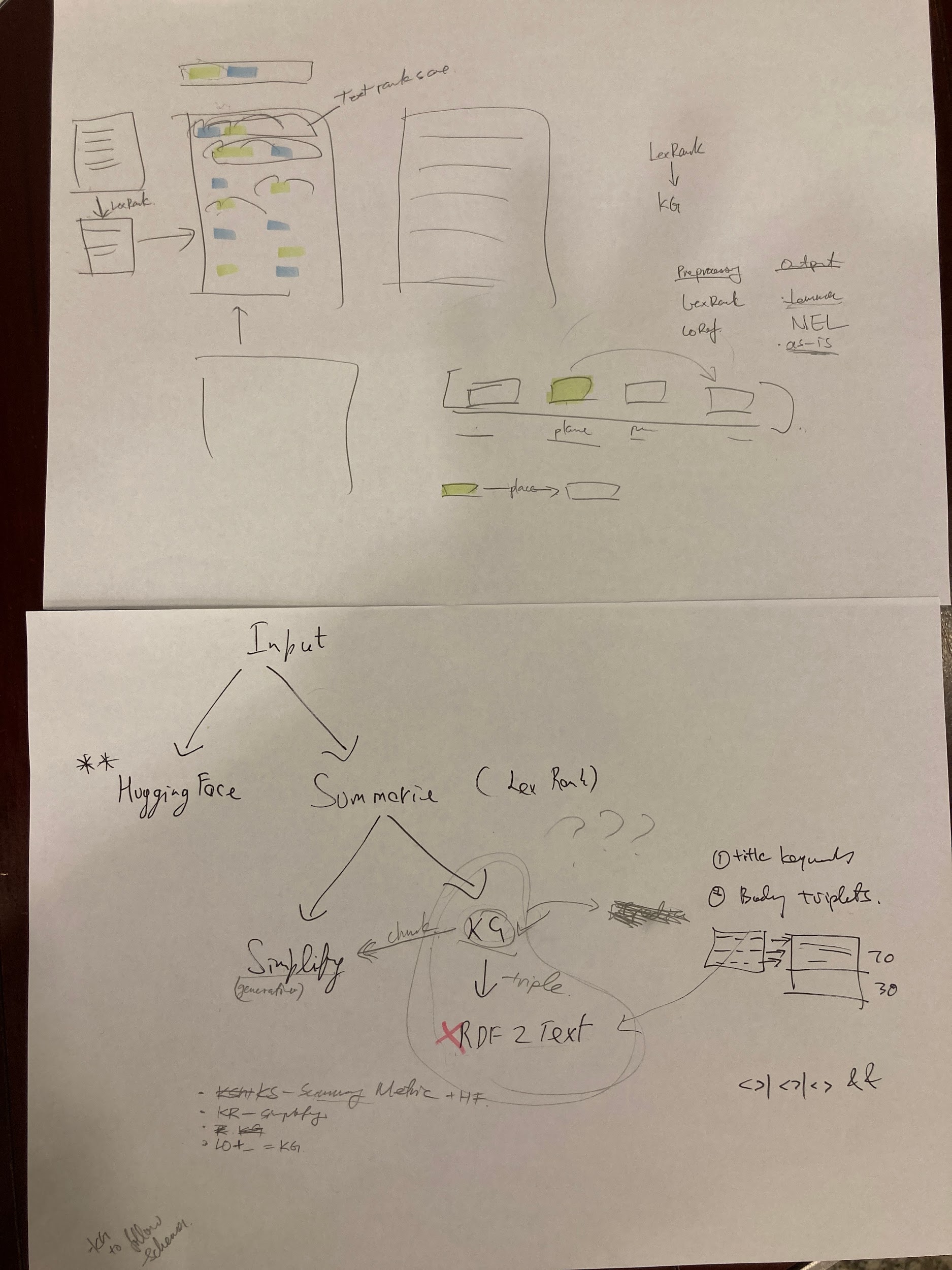


GPT2-Med

TrainOutput(global\_step=5112, training\_loss=1.1265956075725048, metrics={'train\_runtime': 1283.9414, 'train\_samples\_per\_second': 11.932, 'train\_steps\_per\_second': 3.981, 'total\_flos': 3556923660042240.0, 'train\_loss': 1.1265956075725048, 'epoch': 8.0})



# For Late Night Session 1



**Project Pieces**

* **Text-Gen** ← KS
  + ~~No API/Library on HuggingFace that performs rdf2text~~
    - Rdf2text is the formal name for triple to text generation
  + T5 base transformer : <https://ai.googleblog.com/2020/02/exploring-transfer-learning-with-t5.html>
  + WebNLG dataset for triple-to-text training: <https://paperswithcode.com/dataset/webnlg>
  + Notebook used for training: <https://towardsdatascience.com/data-to-text-generation-with-t5-building-a-simple-yet-advanced-nlg-model-b5cce5a6df45>
  + DistilBART 101
    - https://jalammar.github.io/a-visual-guide-to-using-bert-for-the-first-time/
* **Simplification** (hugging face) ← KR
  + <https://colab.research.google.com/drive/1xL5uZibXC-8vqOxwR4XXp_OTVm5B64VU>
* **Summarization** (hugging face) ← KS
  + Use HuggingFace pipelines
    - Alpha philschmid/bart-large-cnn-samsum
    - Meta facebook/bart-large-cnn
* **Knowledge Graph Generator** (text to triplets) ← LO + DX
* **~~Dataset Cleanup~~** ~~(min 50 docs) ← LO~~
* Research
  + Dataset: [NEWS SUMMARY | Kaggle](https://www.kaggle.com/datasets/sunnysai12345/news-summary)
  + [BBC News Summary | Kaggle](https://www.kaggle.com/datasets/pariza/bbc-news-summary)
  + [CNN-DailyMail News Text Summarization | Kaggle](https://www.kaggle.com/datasets/gowrishankarp/newspaper-text-summarization-cnn-dailymail)
  + (sources cited by prof are all paid sources)
* Decision:
  + went with BBC News because 1) text seems more complicated 2) format is simple.
* Outcome:
  + All Files in google drive > data > BBC
  + For DX to continue with KG generation, all articles in “business” category have been processed w coref resolution and reoutput. (folder naming is self evident)
* **~~Evaluation -~~** ~~Simplification Evaluation ← LO~~
* Built index classifier
* **Evaluation** - Summarization Evaluation ← KS
  + Summarization evaluation methods: <https://direct.mit.edu/tacl/article/doi/10.1162/tacl_a_00373/100686/SummEval-Re-evaluating-Summarization-Evaluation>
    - ROUGE (Used Google’s implementation)
      * [~~https://pypi.org/project/rouge-score/~~](https://pypi.org/project/rouge-score/)
    - BertScore
    - BLEU
    - SummaQA

<https://www.overleaf.com/project/6419879881d2a93ca999b774>

Goal: Simplify an article

Process:

Article -> Summary [Krishna’s notebook)

Summary pipelines for benchmark evaluation

John E Blaha, born in San Antonio on 1942-08-26, worked as a fighter pilot.

(John\_E\_Blaha birthDate 1942\_08\_26) (John\_E\_Blaha birthPlace San\_Antonio) (John\_E\_Blaha occupation Fighter\_pilot)

Abner W. Sibal, served in the United States Army and fought in World War II, one of the commanders of which, was Chiang Kai-shek.

Ad sales | boost | Time Warner profit && Quarterly profits | jumped | $ Time 639 && fourth quarter sales | profit | $ 10.9bn && one US media which | buoyed by | less Warner AOL && US Securities Exchange which | has | US Securities Exchange Commission && full Time Warner | projecting | higher earnings revenue && AOL it | book | advertising revenue

Article → KG → Text-gen

Article → Simplify → KG → Text-gen

Article → Summary → KG → Text-gen

KG->Text-gen is our implemented version which is run on 3 different versions of text - original/simplified version/summarized version.

Stretch goal:

Article -> KG+ -> Text-gen

Where KG+ is its own module of KG + Simplification on KG

Text-gen in this scenario refers to triple-to-text generation.

Possible with prototype triplets as referenced here: https://www.ijcai.org/proceedings/2020/0523.pdf

| Datasets + Evaluation | Text Generators | KG Generator |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# 20230223

## Knowledge Graphs

Progress

* Completed proof of concept for knowledge graph generation using tweaked tokenization, NER and POS tagging
* Implemented interactive graph using pyVis . It’s lots of fun!

Problems

* Tokenization
  + Need to implement **co-reference resolution**. The hugging face package is for python 3.7. Alts:
    - <https://github.com/richardpaulhudson/coreferee> ,
    - <https://explosion.ai/blog/coref> , <https://github.com/explosion/spaCy/discussions/11585>
    - [~~https://spacy.io/universe/project/neuralcoref~~/](https://spacy.io/universe/project/neuralcoref/)
    - <https://github.com/explosion/spaCy/discussions/12302>
    - <https://gist.github.com/thomashacker/b5dd6042c092e0a22c2b9243a64a2466>
  + Extend **objects-in the-world tokenization** by using NEL (named entity linking), also sometimes Called “wikification” and fix tokenization
    - [EntityLinker · spaCy API Documentation](https://spacy.io/api/entitylinker/)
    - [[2106.01623] Few-shot Knowledge Graph-to-Text Generation with Pretrained Language Models](https://arxiv.org/abs/2106.01623),
    - [ConceptNet](https://conceptnet.io),
    - [Making Sense of News, the Knowledge Graph Way](https://neo4j.com/developer-blog/making-sense-of-news-the-knowledge-graph-way/)
    - <https://stackoverflow.com/questions/54640715/tokenizing-named-entities-in-spacy#58402708>
    - <https://spacy.io/universe/project/spacy-dbpedia-spotlight>
    - <https://neo4j.com/blog/text-to-knowledge-graph-information-extraction-pipeline/>
    - [facebookresearch/BLINK: Entity Linker solution (github.com)](https://github.com/facebookresearch/BLINK)
    - [Entity Linking (devopedia.org)](https://devopedia.org/entity-linking)
    - Check NLP / NEL.
    - Goal: add span attributes
* Graph
  + Clustering - group semantically similar words together (USA, United States)
  + Graph depth - consider adding hierarchical ordering, maybe by paragraph or topic?
* Overall
  + Review <https://neo4j.com/blog/text-to-knowledge-graph-information-extraction-pipeline/>

Plans

1. Break notebook into explicit parts. It’s currently one big function
2. Implement co-reference resolution
3. Implement wikificstion

Maybe next time

* **Consider text generation from knowledge graphs ?** 
  + [2211.10511.pdf (arxiv.org)](https://arxiv.org/pdf/2211.10511.pdf) (text to graph)
  + [Text Generation from Knowledge Graphs with Graph Transformers | by Alec Robinson | Towards Data Science](https://towardsdatascience.com/text-generation-from-knowledge-graphs-with-graph-transformers-c84156ddd446)
  + <https://paperswithcode.com/paper/investigating-pretrained-language-models-for>
  + [[2107.09556] WikiGraphs: A Wikipedia Text - Knowledge Graph Paired Dataset (arxiv.org)](https://arxiv.org/abs/2107.09556)
  + GNN: [A Gentle Introduction to Graph Neural Networks (distill.pub)](https://distill.pub/2021/gnn-intro/), [What Can You Do With GNNs. Manipulation, Utility and Advantages of… | by Anuradha Wickramarachchi | Towards Data Science](https://towardsdatascience.com/what-can-you-do-with-gnns-5dbec638b525)
* Hugging face summarization walkthrough: <https://huggingface.co/tasks/summarization>
* Consider front end using dash or similar
* For phrase summarization ,TextRank : <https://spacy.io/universe/project/spacy-pytextrank>
* Graph Academic
  + [Stanford Graph Learning Workshop 2022](https://snap.stanford.edu/graphlearning-workshop-2022/)
* Clause segmentation
  + <https://stackoverflow.com/questions/26070245/clause-extraction-using-stanford-parser>
  + <https://stackoverflow.com/questions/9595983/tools-for-text-simplification-java/9606606#9606606>
  + <https://subscription.packtpub.com/book/data/9781838987312/2/ch02lvl1sec13/splitting-sentences-into-clauses>
  + <https://github.com/mmxgn/spacy-clausie>
  + Remove stop words <https://www.dataquest.io/blog/tutorial-text-classification-in-python-using-spacy/>
* Paragraph topics
  + <https://medium.com/@soorajsubrahmannian/extracting-hidden-topics-in-a-corpus-55b2214fc17d>
  + <https://towardsdatascience.com/nlp-with-lda-latent-dirichlet-allocation-and-text-clustering-to-improve-classification-97688c23d98>
  + <https://www.kaggle.com/code/thebrownviking20/topic-modelling-with-spacy-and-scikit-learn>
  + https://www.machinelearningplus.com/nlp/topic-modeling-visualization-how-to-present-results-lda-models/
* KB
  + <https://spacy.io/universe/project/spacy-dbpedia-spotlight>
* Generation
  + <https://github.com/gagan3012/keytotext>
  + <https://github.com/minimaxir/aitextgen>
* Summarization alt
  + <https://medium.com/analytics-vidhya/text-summarization-using-spacy-ca4867c6b744>

Question

* ~~How to do dependency parsing with custom tokens (e.g. using entities)~~

## Summarization

Implementation

* Huggingface course - <https://huggingface.co/tasks/summarization>
* T5 models - <https://huggingface.co/docs/transformers/model_doc/t5>
  + Arxiv paper - <https://arxiv.org/abs/1910.10683> [PDF - <https://arxiv.org/pdf/1910.10683.pdf>]
* Pegasus model - <https://huggingface.co/google/pegasus-large>
* BART model - <https://huggingface.co/facebook/bart-base>
* GPT-2 model - <https://huggingface.co/gpt2-xl>

## Question Answering

Implementation

* HuggingFace pipeline <https://huggingface.co/docs/transformers/main_classes/pipelines#transformers.QuestionAnsweringPipeline>
* End-to-end implementation from scratch: <https://huggingface.co/course/chapter7/7>
  + Using a checkpoint of BERT
  + Should not be required in our case; most of the effort goes in cleanup and tokenization, training process and model architecture is mostly standard

Possible Datasets:

* Natural Questions [Natural Questions](https://ai.google.com/research/NaturalQuestions)
* NewsQA [NewsQA Dataset - Microsoft Research](https://www.microsoft.com/en-us/research/project/newsqa-dataset/)
* Squad <https://rajpurkar.github.io/SQuAD-explorer/> (<https://rajpurkar.github.io/SQuAD-explorer/explore/1.1/dev/>)
* <https://github.com/cooelf/AwesomeMRC> (<https://arxiv.org/pdf/2001.09694v4.pdf>)

Generic Evaluation Metric

* Paper by NUS NLP team [2016, 396 citations] <https://aclanthology.org/D16-1193.pdf>
* Neural Essay Assessor (requires Keras for setup)
* Code provided on Github <https://github.com/nusnlp/nea>
* Can compare our output against this as a benchmark

# Week 06 Discussion

Updates from Lenn

1. Conducted review of academic papers. See my summaries here - [Notes-Lenn](https://docs.google.com/presentation/d/1EBk6Esu80eJax_kHwtZe7VMA1HpSNuiGHQBzRuZz-HI/edit?usp=sharing) and here for papers [📁 references](https://drive.google.com/drive/folders/18dt-upLAv7OniMkhNtVNdl8Vr9_U0bhX?usp=share_link)
2. Implemented a vectorized TextRank summarization in Jupyter. Insightful exercise: [roughwork\_lenn.ipynb](https://drive.google.com/file/d/1d4Ulim8WXLkgVUsHaBj7nMPFbGadbQSm/view?usp=share_link)

Notes

<https://https//catalog.ldc.upenn.edu/topten>

<https://huggingface.co/course/chapter7/5?fw=pt>

# Week05 - 31 Jan

| **“What Topic should we do?** | |
| --- | --- |
| DX | * Interested in knowledge graphs construction |
| Kshitij | * Interested in some form of visualization of the text |
| Krishna | * Not interested in sentiment analysis. |
| Lenn | * Interested in 1 or 2. Pros of 2 is its short text, so we can be more detailed. Pros of 1 is its more interesting * Would love to get really familiar with spacy. * Interested in using an industry benchmark for evaluation (ref: hemingway editor) |
| Misc | * Agreement that 3 and 4 are out for various reasons: complexity of data gathering   From Prof   * To think about: what do we mean by simplifying? Is it length of text? Complexity of words? * To set up tiered goals: basic, ok, stretch. Feels that knowledge graph creation is complex and that should be a stretch goal |
| Decisions | * To focus on Problem 1 |
| Decision for next week | * Dataset - to decide: what dataset do we think is best to use? Relevant only if we intend to used supervised training. (see <https://metatext.io/datasets-list/summarization-task>) * Methodology - to work on and share progress on approaches to problem. |

Example project from Krishna - <https://symbiotic.fyi/>

Involves:

* Knowledge Graph (internal understanding/ visualization of core points)
* Word Sense Disambiguation (knowledge graph requires appropriate PoS tags for effective replies)
* Question Answering (main user interface/ evaluation component for their project)

More from Stanford:

* <https://rajpurkar.github.io/SQuAD-explorer/>

For evaluating the Question Answering system, gives a benchmark plus related models that are SOTA on this metric.

# Brief

## Overview

The purpose of the final project is for you to show how you perform text mining tasks in a practical setting. Given a task (optional with an existing dataset), you need to collect suitable data, select appropriate techniques to solve the task, justify design and implementation issues, as well as interpret your results and assess any limitations of your approach. To provide you with some flexibility for the final project, each team can choose between two options:

**Option A: Pick-your-Project** – you select 1 of the suggested projects.

**Option B: Open Project** – you explore and area of interest and propose your own task to solve using text mining.

In the following, we detail on both options. If you have any questions, please do not hesitate to post your question on the Canvas or send me an email (chris@comp.nus.edu.sg)

### Option A: Pick-your-Project

We have formulated the following 4 specific tasks, from which your team can pick one for your project work. Note that each task is formulated rather broadly, and we list a couple of bullet points how it can be approached – but not all have to be covered! Thus, even if 2 teams pick the same task, their approaches and results can be quite significant. That is on purpose. Also, solutions don't have to be production-ready, but any proof-of-concept should be convincing that an approach might indeed work in practice.

#### "Simplify News"

While Singapore has a high literacy overall, [more than one in four adults](https://www.oecd.org/skills/piaac/Skills-Matter-Singapore.pdf) (26.1%) in Singapore are proficient at or below Level 1 in literacy. This can make it difficult to stay well-informed when reading online news, particularly for Singaporeans and foreigners with poor English skills. Wouldn't it be good if we could:

* Simplify the language or vocabulary
* Simplify and/or shorten sentences
* Summarize articles by focusing on the core message
* Provide support (e.g., provide definitions, synonyms, translations for words)
* News4Kids: visualize news (e.g., convert quotes to images with speech bubbles)

There are various approaches for simplifying text – and we already covered basic rule-based ones in Lecture 3. Of course, there are also [datasets available](https://metatext.io/datasets-list/summarization-task) to treat summarization as a supervised machine learning task.

#### "No Regrets!"

[Many social media users have regretted at least some of their posts](https://dl.acm.org/doi/abs/10.1145/2078827.2078841) . The most common reasons are embarrassment and the reveal of sensitive information. In the fast-paced online world, deleting a post even after a couple of minutes is usually already too late. Let's try to warn users before they click "Send" about potential negative consequences:

* What might classify as "regrettable" content? Phone numbers? Offensive remarks?
* How can we identify, e.g., sensitive content?
* How can we explain to users our analysis result?
* Can we help users to mitigate problems (e.g, remove sensitive snippets)?

Approaches to address these questions range from pattern matching (e.g., identifying phone numbers), Named Entity Recognition (e.g., identifying persons and places), text classification (e.g., is a statement offensive or not), to sentence simplification/rewriting.

#### "Storms 'n' Stocks"

Sudden natural disasters (e.g., earthquakes, floods, storms) and man-made disasters (e.g., terrorist attacks, major accidents) can have a significant [effect on the stock market](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6407467) and the global economy in general: crucial supply chains might be interrupted, travel restrictions, relief efforts, etc. Quickly detecting and assessing such events can therefore be very useful when dealing with stocks. On the Web, information about such events can be available very early:

* Monitor news of social media sources (e.g., Twitter accounts of governmental agencies)
* Extract time, location, and type of event (e.g., earthquake)
* Estimate/assess severity of event (e.g., strength of earthquake)
* Assess potential impacts (e.g., affected industries near the event)

A core component to address this task is certainly Named Entity Recognition as well as keyword extractions. Clustering might help to avoid overcounting multiple reports about the same event. Another related task is entity linked, i.e., to connect information (e.g., the name of a city or region) to a knowledge base to extract useful information.

#### "Wellbeing Watch"

In recent years, mental wellbeing has become a major health-related topic in Singapore. Even if it's just to raise awareness, NUS introduced the Wellbeing Day a year ago. Mental health problems come in all shapes and forms, with affected people seeking help often not early enough, if at all. However, people share their thoughts, fears, worries, etc. across social media and/or online forums. Can we use this data to predict and monitor the mental wellbeing of individuals or of a society? For this we need:

* Appropriate source to collect relevant information (e.g., subreddits?)
* A good understanding how mental health problems may manifest in writing
* Classification models to predict mental health problems with a high accuracy
* Interpretable/explainable models to allow for effective interventions

This is first and foremost a classification task. However, it relies on solid data to ensure meaningful results. So data collection as well as data preprocessing / preparation will be very important. Classification models will arguably face the trade-off between complexity (high accuracy) and simplicity (good explainability).

### Option B: Open Project

The open project allows you to explore and area of your interest in the form a text mining task. Thanks to the Web, text data containing interesting information is omnipresent and there are many ideas to build useful applications using text mining techniques. Alternatively, if you are currently working in a particular industry area, or doing research within a particular academic area, you can also pursue a topic in those areas. This could take the form of performing some data analysis on a dataset from your domain of interest, or proposing a new method relevant to text mining on a particular type of data.

#### Project Proposal

If your choose the open project, you will have to submit a **1-page proposal** briefly motivating the task, describing data, outline important research questions expected outcomes;

we will provide a basic [template](https://canvas.nus.edu.sg/courses/38784/files/796197?wrap=1)for that. This proposal is not graded, and is purely to allow the teaching team to ensure your project topic has reasonable scope and feasibility, and possibly offer some helpful advice. If needed, you can still change your project topic after the deadline, but please let the teaching team know first (so they can help to ensure that the new topic and new time frame is reasonable). Also feel free to approach us if you would like feedback on your project topic before the deadline.

#### Things to Consider

If you are looking into Option B, here are a couple of points you should consider before making your final decision.

* **Scope.** The project should be in line with the course aims of CS5246. This includes the emphasis on understanding the data, the pros and cons of different text mining methods for a given task, and the meaningful interpretation of the results. This is generally not trivial for unstructured data such as text. Just having a good deep learning model for some text classification task does not guarantee a good grade.
* **Dataset availability.** Make sure that appropriate data for the task available – e.g. a public dataset available for download – or can be (easily) collected. If data needs to be collected, using public API is the preferred alternative to Web scraping. We can provide some help in terms of code samples for collecting data from the Web.
* **Computing power.** Depending on your task of choice, the evaluation of your solution might requires large amounts of processing power. You should make sure that processing time does not become a bottleneck for your project. Note: not all text mining task require much computing power.

#### Useful Resources

**Repositories of Public Datasets.** Below you can find links to a selected set of repositories of publicly available dataset. There are many more such repositories, particularly for very

domain-specific datasets, to can find with your Web search engine of choice.

* [Google Dataset Search](https://datasetsearch.research.google.com/)
* [Kaggle public datasets](https://www.kaggle.com/datasets)
* [UCI Machine Learning Repository](http://archive.ics.uci.edu/ml/index.php)
* [Github page with links to public datasets](https://github.com/awesomedata/awesome-public-datasets)
* [Subreddit of answers](https://www.reddit.com/r/datasets/)
* [Singapore open datasets](https://data.gov.sg/)

**Data collection using public APIs.** In principle, you can also collect your own dataset using public APIs. Here some resources where you can browse for public APIs:

* [ProgrammableWeb](https://www.programmableweb.com/)
* [RapidApi](https://rapidapi.com/collection/list-of-free-apis)
* [APIs.guru](https://apis.guru/browse-apis/)
* [AnyAPI](https://any-api.com/)
* [GitHub page with links to public APIs](https://github.com/public-apis/public-apis)

If you are considering to collect your own data, here are some things to consider:

* Data collection might take take, particularly if an API provides only live but not historical data, and you need a sufficient amount of data for a meaningful analysis (e.g., several weeks and months)
* Make sure that the API is indeed public and free to use; this is not always very obvious until you are indeed trying to access the API.
* Most open APIs have rate limits, i.e., you can only submit a limited number of requests within a time interval. Make sure that such rate limits will not cause a bottleneck for your data collection. In short, relying on self-collected data will involve some risks when it comes to achieving your project goals in the allocated time frame. So please be aware of that.

## Deliverables

### Progress Report

The progress report will be a simple slide deck as a PDF document of approx. 10-15 slides. The purpose of the progress report is two-fold: (a) to give us a chance to check if your project goes into the right direction, and (b) to provide you with a little incentive to start early. There is no official layout or structure. As the name suggests, it should outline your progress with your project work (e.g., goals and questions, understanding of task and the data, first design decisions and/or results, but also issues/challenges/obstacles that you are facing). The last 1-2 slides should outline the next steps until the end of the project.

* Deadline: TBD soon

**Note:** You are welcome to submit your progress report earlier. Ideally, this will in turn give you earlier feedback, but also allow us to better balance the workload. The progress report will not be explicitly graded but not submitting any report will negatively affect your final grade.

### Final Report

The final report will be a PDF document in the format of a scientific paper of at most **10 pages** including tables, plots and figures, but excluding references and the appendix. The appendix may contain supplementary content but should be used sparingly. As a rule of thumb, the report should be readable and completely comprehensible without the appendix. The appendix typically may include plots or tables that elaborate on the results of your EDA or your evaluation. For the layout and presentation of the report, we provide [templates](https://canvas.nus.edu.sg/courses/38784/files/796197?wrap=1)for Word and LaTeX.

#### Structure & Content

Your report should include the name and student IDs of all team members as well as your team name – pick something cool :). Please also include a breakdown of your workload, i.e., some overview what team member was (mainly) responsible for each part of the project. This can be a table, Gantt chart, etc. to be added to the appendix. While the overall structure of the report is up to you, it should cover the following aspects – although this might differ from your exact project task:

* **Motivation.** Motivate and outline the goals and questions you address. Why is this work important, what are the challenges, will benefits from the results. In short, what problem are you trying to solve?
* **Data Collection & Data Understanding.** If your project requires the collection of data, discuss why this data is suitable for your task; technical details may include how you collected the data. You should show a good understanding of your data, i.e., appreciate the characteristics of the data (e.g., language/vocabulary, writing style, etc.)
* **Data Preprocessing & Preparation.** Motivate and justify which preprocessing steps you might have performed to transform the raw text dataset into a valid for the subsequent text mining techniques.
* **Text Mining Methods.** Describe how you chose and applied appropriate text mining techniques (e.g., clustering, classification, information extraction) to solve your task, or at least brought you closer to solve your task. This description should include the choice of important parameters. Note that you do not need to explain the techniques themselves. However, in case of more advanced methods or models, you should add relevant references. As a rule of thumb: Everything covered in the lectures does not need to be cited.
* **Evaluation & Interpretation.** Evaluate the performance of your approach, where performance mainly refers to effectiveness and less on efficiency. If your project allows for the consideration of different models (e.g., different types of classifiers) compare the performance of different methods. Discuss/interpret your results, and understand in what cases your methods perform bad, and discuss principle limitations and potential future steps for improvement.

#### Submission

The final submission contains both the report as a PDF document as well as your source code, uploaded to Canvas in a zipped folder. Instead of the source code, you can also add a link to a GitHub repository. Note that the reproducibility of your approach is part of the grading (see below) which includes the organization, documentation, and readability of your code.

* Deadline: TBD soon

## Grading

### Marking Guide

The individual projects are likely to different greatly. To keep it fair, all projects will be based on the same fundamental criteria. In a nutshell, a good grade requires that your approach is methodologically sound, and that the outcome – mainly the report but also your source code – is of a high quality. In more detail, we weigh the core criteria for the grading as follows

**Methodological Quality (60%).** While the exact distribution may depend on your chosen project and approach, methodological quality generally covers the following aspects:

* Data preparation: appropriate preprocessing methods are chosen (informed by the results of the EDA) and correctly implemented; missing values, categorical attributes, etc. are handled correctly.
* Visualization: appropriate plots, figures and tables are used to visualize results, architectures and work flows.
* Methods: applied methods are well motivated and correctly implemented; alternatives are discussed and design decision are justified.
* Evaluation: depending the proposed approach, the correct experiment setup and evaluation metrics areconsidered; common errors and principle limitations are evaluated and discussed.

**Quality of Report (30%).** The report describes your methodology and explains your results in a clear, concise and comprehensible manner. Related work should be appropriately referenced; the limit of 10 pages should not be exceeded (excluding references and appendix).

**Reproducibility (10%).** The code you submit is complete, well-organized ad readable. Simply put, it should be easy for an outsider to use and understand your code to retrace your steps and reproduce results.

### Team Assessment

By default, all members of a team will receive the same marks. However, we reserve ourselves the possibility to deviate from this approach in case different marks for different members are indeed warranted. To quantify this, but also to incentivize each team member to equally contribute, we will be using the the peer review system [TEAMMATES](https://teammatesv4.appspot.com/web/front/home) . For a smooth collaboration we recommend:

* Start the project early (at least to get a sense of the expected workload)
* Meet and discuss regularly (organize at least 1 or 2 weekly sessions where the whole team meets, if possible)
* Include some buffer time for any exams or unexpected and unplanned idle times.
* Have fun and make it a good experience!

If your team faces some internal conflicts, best let the teaching team know as soon as possible.