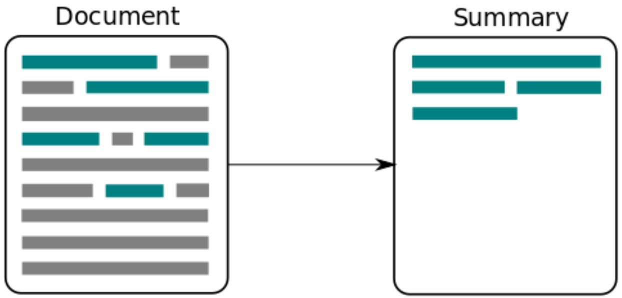
NLP Summarization Techniques

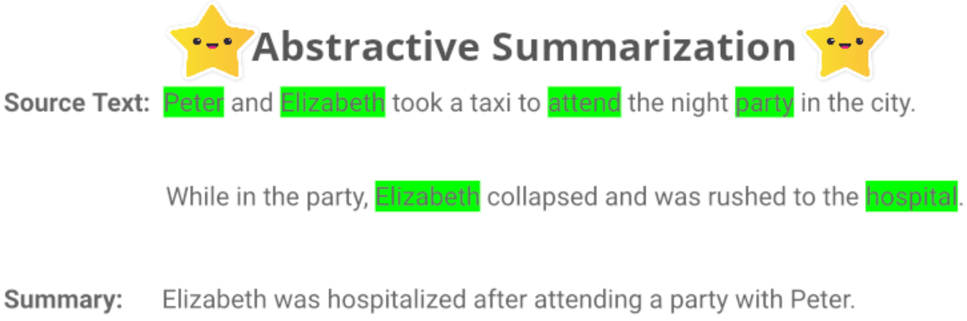
Abstract Text Summarization: Let Transformers Summarize for You!



DATA606: Capstone in Data Science
Final Project, Fall 2022
Lee Whieldon

What is the value of abstract text summarization?

- **Time Saving** – Summarization of Long Meeting Transcripts
- **Consistency** – Summarize complex corpus of text like contracts
- **Customer Satisfaction** – Distilling collection of dialogue into an actionable summary



Data - SAMsum Corpus

Features:

- Dialogues
- Summaries
- Ids – unique identifier

Data Splits:

- Train: 14,732 records
- Validation: 818 records
- Test: 819 records

16,369 dialogues in total

Dialogue:

Hannah: Hey, do you have Betty's number?
Amanda: Lemme check
Hannah: <file_gif>
Amanda: Sorry, can't find it.
Amanda: Ask Larry
Amanda: He called her last time we were at the park together
Hannah: I don't know him well
Hannah: <file_gif>
Amanda: Don't be shy, he's very nice
Hannah: If you say so..
Hannah: I'd rather you texted him
Amanda: Just text him 🙂
Hannah: Urgh.. Alright
Hannah: Bye
Amanda: Bye bye

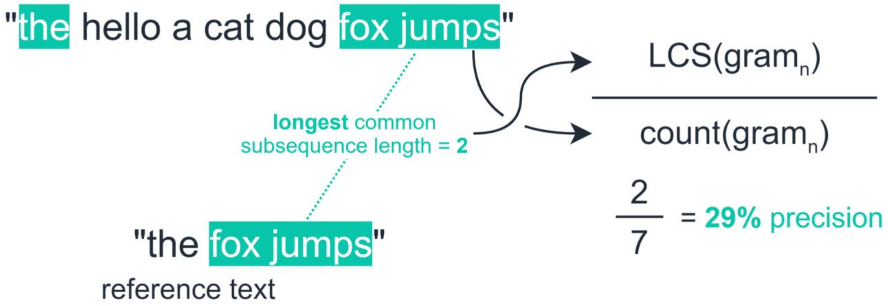
Summary:

Hannah needs Betty's number but Amanda doesn't have it. She needs to contact Larry.

Scoring Metrics

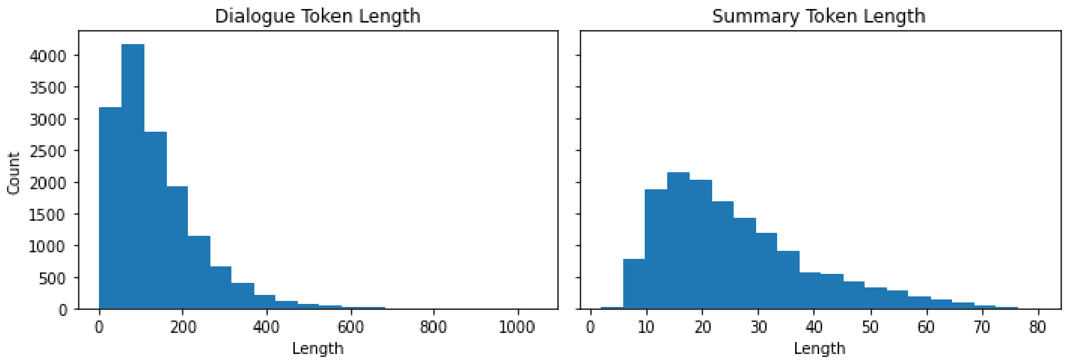
Recall-**O**riented **U**nderstudy for **G**isting **E**valuation – ROUGE!

- Developed for automatic summarization and machine translation software
- High recall is more important than precision scoring alone
- Longest common subsequence (LCS): Help calculate the score per sentence and averages it for the summaries
- We also calculate the score over the whole summary (ROUGE L-Sum)



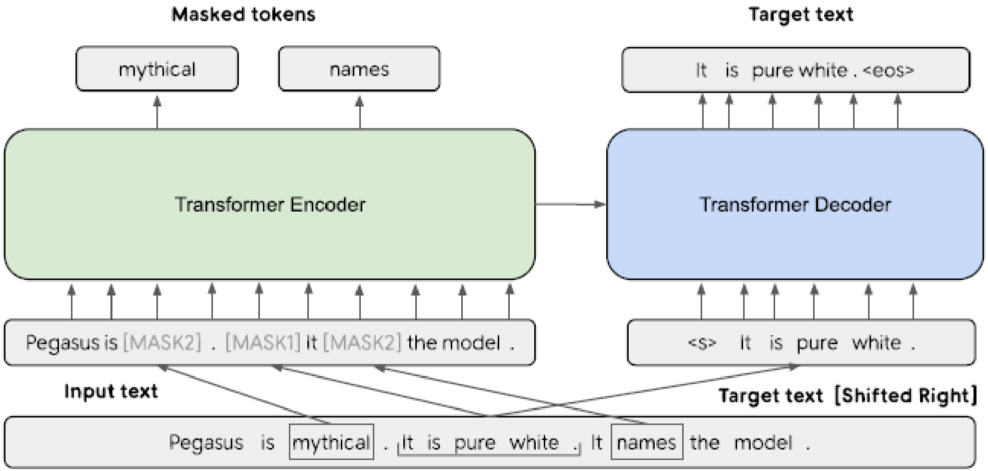
Pretraining Analysis

- Length of distribution:
    - Each Dialogue records contain 100-200 tokens (a.k.a. words)
    - Summaries are shorter, with around 20-40 tokens



Model - PEGASUS

- Encoder-decoder transformer
- Pretraining objective is to predict masked sentences in multisentence texts
- Applying Gap Sentence Generation (GSG)
- Applying Masked Language Modelling (MLM)



UMBC

Training & Evaluation

Google's Pegasus-CNN_DailyMail

- Transfer learning from google/pegasus-cnn_dailymail .

Training hyperparameters

- Learning rate: 5e-05
- Train batch size: 1
- Eval batch size: 1
- seed: 42
- Gradient accumulation steps: 16
- Total train batch size: 16
- optimizer: Adam with betas=(0.9,0.999) and epsilon=1e-08
- Lr scheduler type: linear
- Lr scheduler warmup steps: 500
- Number of epochs: 1

Training results

- Training Loss: 1.6776
- Epoch: 0.54
- Step: 500
- Validation Loss: 1.4919

rouge1		rouge2		rougeL		rougeLsum	
pre-training score	after training score	pre-training score	after training score	pre-training score	after training score	pre-training score	after training score
0.29614	0.431695	0.087609	0.201628	0.229381	0.346877	0.229379	0.347153

Conclusions & Limitations

- Not applicable for long corpus
  - Most models today can handle up to 1,000 characters
  - Research is actively being done today to account for this limitation
- Expedited processing with greater GPU clusters

Demo!

Thank you!

Looking forward to seeing everyone's projects!