CAPSTONE PROJECT PROPOSAL: TEXT SUMMARIZATION

Problem Statement

I propose here to do my capstone project on text summarization (for legal documents) where the objective is to read in a piece of text (potentially containing many paragraphs) and output a summarized version of it. A good summarizer will output all the important details from the input text while being succinct.

There are many places where a good document summarizer will be valuable. For example, in the legal industry it can be used to summarize long legal documents, in the healthcare industry it can be used to summarize important aspects of a medication, in the news industry it can be used to summarize news articles, and many other examples.

In terms of approaches, there are two different types to text summarization approaches:

1. Extractive text summarization identifies important sections of the original article and then copies it to form the summary. It can be thought of as a highlighter.
2. Abstractive text summarization reproduces important information in the article by first understanding the entire article and then succinctly generating new text based upon it.

Out of the two, abstractive summarization is more like what humans do; and thus, it has greater potential. But the downside is that it is much more complicated to implement because it requires a language model to generate new text based upon some prior (i.e. the input article). For this project, I plan on developing an abstractive text summarizer using deep learning.

Example

Here is an example showcasing the performance of a few different text summarization architectures [1].

Original Text (truncated): lagos, nigeria (cnn) a day after winning nigeria's presidency, muhammadu buhari told cnn's christiane amanpour that he plans to aggressively fight corruption that has long plagued nigeria and go after the root of the nation's unrest. buhari said he'll "rapidly give attention" to curbing violence in the northeast part of nigeria, where the terrorist group boko haram operates. by cooperating with neighboring nations chad, cameroon and niger, he said his administration is confident it will be able to thwart criminals and others contributing to nigeria's instability. for the first time in nigeria's history, the opposition defeated the ruling party in democratic elections. buhari defeated incumbent goodluck jonathan by

about 2 million votes, according to nigeria's independent national electoral commission. the win comes after a long history of military rule, coups and botched attempts at democracy in africa's most populous nation.
Baseline Seq2Seq + Attention: UNK UNK says his administration is confident it will be able to destabilize nigeria's economy. UNK says his administration is confident it will be able to thwart criminals and other nigerians. he says the country has long nigeria and nigeria's economy.
Pointer-Gen: muhammadu buhari says he plans to aggressively fight corruption in the northeast part of nigeria. he says he'll "rapidly give attention" to curbing violence in the northeast part of nigeria. he says his administration is confident it will be able to thwart criminals.
Pointer-Gen + Coverage: muhammadu buhari says he plans to aggressively fight corruption that has long plagued nigeria. he says his administration is confident it will be able to thwart criminals. the win comes after a long history of military rule, coups and botched attempts at democracy in africa's most populous nation.

Example [1]: Comparison of output of 3 abstractive summarization models on a news article. The baseline model makes factual errors, a nonsensical sentence and struggles with OOV words muhammadu buhari. The pointer-generator model is accurate but repeats itself. Coverage eliminates repetition. The final summary is composed from several fragments.

Dataset

There are many different datasets available for text summarization. Here are a few examples.

1. CNN/Daily Mail dataset: this dataset is sourced from CNN/Daily Mail news articles that also have a human written summary. This dataset has over 300k training examples taking about 1.2GB of space.
2. Gigaword dataset: this is based upon the Gigaword corpus which is one of the largest static corpus of English news documents. This dataset has about 4M examples taking almost 1GB of space.
3. Arxiv and Pubmed scientific papers: this dataset is sourced from all the papers published in these two places where the text of the paper is the input article and the abstract is the target summary. This dataset has about 320k training examples taking about 21GB of space.
4. BigPatent dataset: this dataset is sourced from all the patents filed with the USPTO since 1971 where the article is the patent description and target summary is the patent's abstract. It has about 1.2M training examples (25GB).

Amongst these, BigPatent is a better dataset because of (a) its large size, (b) it contains longer articles, (c) it has important information uniformly distributed throughout the article and so the model can't easily cheat, (d) its summaries contain richer discourse

structure with more recurring entities, and (e) its summaries contain fewer and shorter extractive fragments [2]. To that end, I plan on using the BigPatent dataset.

In terms of data collection, I have written a script to collect all these datasets, the details of which can be found on my GutHub at [3].

Model

This is a classification type of supervised learning problem because we have discrete target outputs (i.e. sentences/words) that the model needs to learn. The output predictor's dimension is the size of the vocabulary -- i.e. very high dimensional.

I plan on using deep learning for this as traditional machine learning algorithms have struggled with this task. The basic idea is to use a sequence-to-sequence model (e.g. encoder-decoder) that reads in the input article, converts it into some latent space representation, which is then used by the decoder to generate natural language text that resembles the target summary. One such approach is shown in [1], as has already been discussed above.

My final deliverable will be a web based API where a user inputs the text they'd like to summarize and the API will display the summary generated by the trained model.

Resource Usage

A GPU is required for this project due to the inherently parallel nature of training deep neural networks. In particular, I will likely use Nvidia's V100 (or a better) GPU to get reasonable training speed. Moreover, given the large dataset size and a potentially large model, V100's memory of at least 16GB will allow for a reasonable training batch size. V100 GPU is available at all the three major cloud service providers (i.e. AWS, Azure, & GCP); but owing to my past experience, I plan on using GCP for this project. In terms of framework to build and train the model, I plan on using Pytorch due to my prior experience using it.

References:

[1] - https://arxiv.org/pdf/1704.04368.pdf
[2] - https://arxiv.org/pdf/1906.03741.pdf
[3] - https://github.com/gtg162y/Text_Summarization_UCSD/tree/main/Data_Collection_3-5-2