1. Too Long Don’t Read TLDR: Text summarization from basic to advanced approaches

In this project, we explored several neural-based approaches to do abstractive text summarization on news articles. The methods we use are vanilla sequence-to-sequence models without attention [1], BART transformer models with two different sizes [2], and T5 transformer models with three different sizes [3]. As these models have different architectures and complexity, we would like to evaluate and compare their performance using ROUGE metrics [4]. Throughout our experiments, we conclude that a LSTM sequence-to-sequence model lacks the ability to generalize with unseen data compared with fine-tuned and pre-trained transformers.

**2.** -So what is summarization? Text summarization is the task of producing a concise and fluent summary of texts or documents while preserving key information.

-There are two main strategies for text summarization.

The first is summarization by extraction, which consists of concatenating source sentences into a summary. (count the frequency of words in documents. Words that occur often are likely to be the main topic of the document. However, this approach does not account for words in different contexts.)

And second is summarization by abstraction, which involves generating novel sentences for the summary

-The goal of this project is to implement different abstractive summarization techniques on the same dataset and compare them based on the same evaluation metrics such as ROUGE.

**3.** -We use raw data without data anonymization of the CNN-Daily Mail dataset, which consists of online news articles (or stories) paired with multiple summaries (or highlights). This data is shown in the table.

-The first feature is a text of news articles which is used as the documents to be summarized while the second feature is the joined text of highlights which is the target text summarization.

-The articles have an average of 781 tokens while the summaries have an average of 56 tokens.

- For pre-processing, we remove punctuation, number, and CNN and Daily Mail name tags at the beginning of every line. We also lower every letter in all words and remove noisy words that are not related to news articles such as advertisements.

-And to save computation time, we use only the first two sentence of stories together with 2 summaries.

**4.** For our approaches,

-First, our baseline model is a vanilla LSTM sequence-to-sequence model which is a general Deep learning-based architecture used in NLP tasks. Generally, we use LSTM with an encoder-decoder architecture inspired by Neural Machine Translation. Summaries are generated from the decoder, using target vocabulary. This model provides us a baseline of the training time and accuracy

-For our second model, we finetune the BART-base model on our pre-processed dataset. BART-base is a pre-trained transformer model trained on XSum news dataset.

-BART uses a standard sequence-to-sequence architecture with a Bidirectional encoder similar to BERT and a left-to-right decoder similar to GPT.

It is a denoising autoencoder built with a sequence-to-sequence model that is applicable to a very wide range of end tasks. Text is corrupted with an arbitrary noising function.

The second stage is that a sequence-to-sequence model is learned to reconstruct the original text. it is useful for text summarization tasks because it has an autoregressive decoder as it can be directly finetuned

A key advantage of this setup is the noising flexibility as arbitrary transformations can be applied to the original text such as arbitrary length adjustment.

-For BART-large, we directly use the model without finetuning.

-For T5, it is an encoder-decoder model pre-trained on a multi-task mixture of unsupervised and supervised tasks. It is trained on C4 dataset which is a cleaned version of Common Crawl’s web crawl corpus.

we present a large-scale empirical survey to determine which transfer learning techniques work best and apply these insights at scale to create a new model that we call the Text-To-Text Transfer Transformer (T5).

Text-2-Text - According to the graphic taken from the T5 paper. All NLP tasks are converted to a text-to-text problem. Tasks such as translation, classification, summarization and question answering, all of them are treated as a text-to-text conversion problem, rather than seen as separate unique problem statements.

Since the task is reflected purely in the text input and output, you can use the same model, objective, training procedure, and decoding process to ANY task. Above framework can be used for any task

-The goal of this model is to reframe all NLP tasks into a unified text-to-text-format where the input and output are always text strings. T5 uses the same Masked Language Model as BERT but it is different from BERT-based models that can only output either a class label or a span of the input. The advantage of this model is that it allows us to use the same model, loss function, and hyperparameters on any NLP task

**5.**

-To evaluate the generated summaries, we use F1 from the ROUGE metric which is commonly used in summarization tasks.

-We select only three specific ROUGE metrics namely: ROUGE-1 and ROUGE-2 which are similar to the BLEU metric by measuring unigram and bigram similarity of reference summary and model-generated one. On the other hand, ROUGE-L measures the longest common subsequence between the two summaries.

-In this example, we can see that there are total 7 words in the system summary, and 6 of them are overlapping words,

so the precision score should be 6 divided by 7 and recall score is 6 divided by 6. And this two number will be used for calculating F1 score for ROUGE-1

**6.**

-From the table, we can see that vanilla LSTM model that was trained from scratch has a relatively disappointing performance in all ROUGE metrics compared with other models.

-The performance of Transformer-based models like BART-large model and all three T5 models almost doubles the LSTM model. Though, they have very similar performance to each other in all metrics regardless of size of models and underlying datasets.

-On the other hand, our finetuned Transformer-based BART-base model outperforms LSTM but still cannot compete with other Transformer models. This may be because of how we perform finetuning or data processing.

-Overall, the accuracy of the best model in our experiment is not high compared with the current state-of-the-art models.

-Nonetheless, in other experiments from different researchers using this same dataset, their best models hardly have an accuracy larger than 0.4 (ROUGE-1, ROUGE-L) and 0.3(ROUGE-2).

-So, there is still a long way to go for text summarization.

**7.**

-Here are the examples of our model outputs compared with a reference summary.

-We can see that the summary output from LSTM sequence to sequence model is incorrect as seen in the word like Manchester United repeating itself in both subject and object.

-In contrast, the outputs of other models are correct and understandable in terms of meaning and grammar.

-There are two key points in the reference summary.

- The first is Manchester united have made Schweinsteiger as their top target. BART base model and all T5 models identified this point successfully.

-The other point is Van Gall eyes Scheweinsteiger reunion at Old Trafford. And for this point, only BART large model can capture this detail.

-To summarize, almost all the models successfully summarize one key point from the text except the LSTM model.

-And the sentences they generate are readable and concise.

-Although none of them point out both of two key points, we think that their performance are acceptable and helpful.

**8.**

From our analysis of news summarization, we conclude that

The accuracy of text summarization models can be increased by using transformers.

Our current models can generate a correct and readable summary sentence in some cases.

Text summarization is still a challenging topic.

The overall accuracy of our models is still relatively low; and in other experiments, researchers can hardly generate a model with the accuracy higher than 0.4.

For Future work, we can incorporate Named-entity recognition to our model since large portion of summaries involve named entity