**Approaches towards text summarization in NLP**

2 approaches: Abstraction vs extraction(still famous)

1. **Extraction based summarization**

**Key phrases are pulled out of a text in the source document and combined to form a summary of the document! The extraction is made according to the defined metric without changes to the original text. E.g as below! This leads to grammatically strange sentences**

**Source text:** **Joseph and Mary** rode on a donkey to **attend** the annual **event**in**Jerusalem**. In the city, **Mary** gave **birth** to a child named **Jesus**.**Extractive summary:** Joseph and Mary attend event Jerusalem. Mary birth Jesus.

**Extractive summary:** Joseph and Mary attend event Jerusalem. Mary birth Jesus

1. **Abstraction based summarization**

The abstraction technique entails paraphrasing and shortening parts of the source document. When abstraction is applied for text summarization in deep learning problems, it can overcome the grammar inconsistencies of the extractive method.

The abstractive text summarization algorithms create new phrases and sentences that relay the most useful information from the original text — just like humans do.

Therefore, abstraction performs better than extraction. However, the text summarization algorithms required to do abstraction are more difficult to develop; that’s why the use of extraction is still popular

Here is an example:

**Abstractive summary:**Joseph and Mary came to Jerusalem where Jesus was born.

Link:<https://towardsdatascience.com/a-quick-introduction-to-text-summarization-in-machine-learning-3d27ccf18a9f#:~:text=Text%20summarization%20refers%20to%20the,natural%20language%20processing%20(NLP)>.

**How decoding algorithms work:**

Different types of decoding algorithms have different methods of determining which word to return next during auto-text summarization

Link:<https://medium.com/voice-tech-podcast/visualising-beam-search-and-other-decoding-algorithms-for-natural-language-generation-fbba7cba2c5b>

Deploying API for text summarization

Link:<https://medium.com/analytics-vidhya/build-save-and-deploy-your-first-web-app-using-flask-and-pythonanywhere-110ddd691026>

Linkflask simplified:<https://dev.to/paymon123/create-a-text-summarization-api-with-flask-sumy-and-trafilatura-12b6>

Link Kaggle : <https://www.kaggle.com/sidsharma97/create-your-own-text-summarization-app> (all functions)

**Article from Gulci**

Link:https://www.machinecurve.com/index.php/2020/12/21/easy-text-summarization-with-huggingface-transformers-and-machine-learning/

**Hugging face transformers**

Link: <https://huggingface.co/transformers/>

App idea: <https://youtu.be/uGYJuOyIvzs>

**Tesnor flow and cuda:** <https://yann-leguilly.gitlab.io/post/2019-10-08-tensorflow-and-cuda/>

**Tensorflow 2 and 2.1:** <https://github.com/tensorflow/tensorflow/issues/34759>

Use this: https://towardsdatascience.com/abstractive-summarization-using-pytorch-f5063e67510