

The background is a dark navy blue. On the left, there is a large, semi-transparent circular graphic containing a detailed image of a printed circuit board (PCB). Overlaid on the top-left of this circle are two overlapping triangles: a blue one in front and a light green one behind it. In the top-right corner, there is a faint, high-contrast image of a circuit board's surface, showing intricate patterns of copper traces and components.

# NEWS SUMMARIZATION

Capstone project  
Kotapati Naga Sai Ruthvik

# Contents

Problem Statement

Introduction

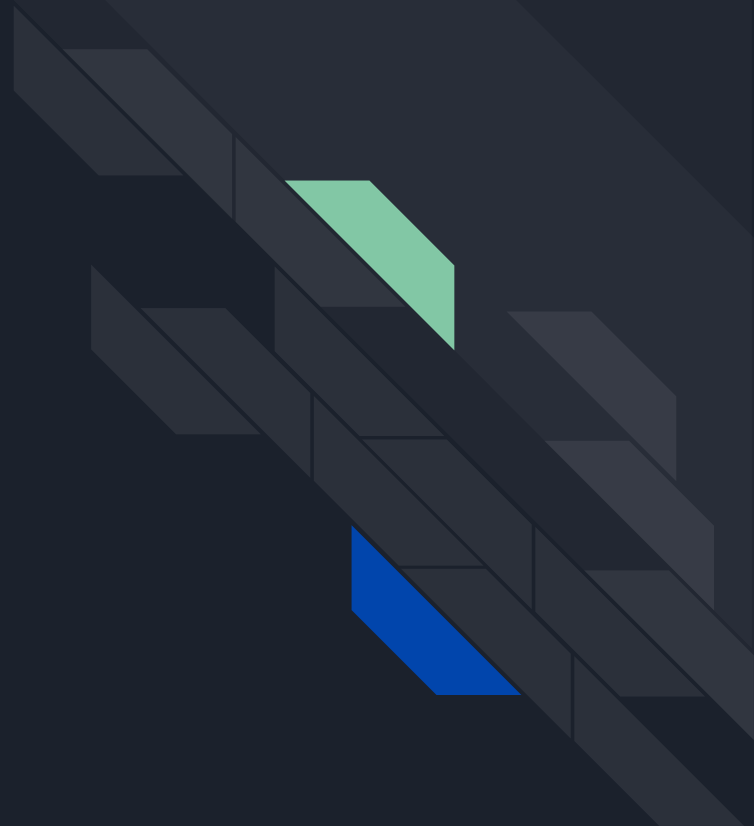
Use Cases

Types of Text summarization

Dataset

Training and Serving

Conclusion





# Problem Statement

1. Create an end-to-end solution that performs extractive text summarization of newspapers.
2. Create an end-to-end solution that performs abstractive text summarization of newspapers.
3. Test out all solutions using newspaper articles from recent times.
4. Serve both models using Tensorflow serving (or any other form of serving)



# Introduction


Text summarization is a technique to shorten long texts to such that the summary has all the important information. The problem statements states to build an end to end solution for text summarization on news papers. The project requires a text to text sequence generators and corresponding models are used.

The outline of the project is as follows :

1. Choose a text to text extraction model and fine tune on news paper articles.
2. Test the model on recent articles.
3. Serve the model using TF serving



# Use Cases

- 
1. Used by several websites and applications to create news feed and article summaries.
  2. People prefer reading short summaries compared to a lengthy report.
  3. Making notes from reports or books



# Types of Text Summarization

Extraction based summarization	Abstraction based summarization
It involves picking the most phrases and lines from document.	It involves summarizing based on deep learning.
It then combines all the important lines and sentences to make up the summary.	It generates text based on learning and is similar to how we summarize, keeping the point same.
Every line and word of summary is from the actual text.	It uses new phrases and terms, different from the actual text.
Used Bert based model	Used T5-small model



# Dataset

1. For both the models I used CNN Daily news dataset from datasets
2. It consists of three fields articles, highlights and ID.
  1. Article - The actual text of news article
  2. Highlights - The summary of the news article
  3. ID - A unique identifier corresponding to each article.
3. Total number of articles is around 300k

```
DatasetDict({
  train: Dataset({
    features: ['article', 'highlights', 'id'],
    num_rows: 287113
  })
  validation: Dataset({
    features: ['article', 'highlights', 'id'],
    num_rows: 13368
  })
  test: Dataset({
    features: ['article', 'highlights', 'id'],
    num_rows: 11490
  })
})
```



# Training and serving the model

Here is the link to the notebook used for generating summary based on extraction approach

[https://drive.google.com/file/d/16JO0pc6\\_pduvU3ClkOuQv9T\\_v3LNPjhP/view?usp=sharing](https://drive.google.com/file/d/16JO0pc6_pduvU3ClkOuQv9T_v3LNPjhP/view?usp=sharing)

Here is the link to the notebook used for generating summary based on abstraction approach

<https://drive.google.com/file/d/1118y6IOVShoeYMB-2Yt2qwAO5x6ko1j-/view?usp=sharing>

The notebooks are well documented and explanations are provided wherever necessary.





# Conclusion

1. Built an end-to-end solution for both extraction and abstraction based text summarization on news datasets.
2. Used pre-trained models and fine tuned on the CNN dataset.
3. Served the model using TF serving and the results were similar to usual predictions.
4. Generated summaries on abstraction based approach were satisfactory but extraction based approach can be improved.



Thank you!