TEXT SUMMARIZATION WITH HUGGING FACE TRANSFORMERS: A BEGINNER'S GUIDE

RAMCHANDRA PADWAL

WHAT IS TEXT SUMMARIZATION?

Text summarization is the process of condensing a large text document into a shorter version while preserving its key information and meaning. The goal of text summarization is to extract the most important information from a text document and present it in a concise and comprehensible form. This can be done through techniques such as keyword extraction, sentence extraction, or abstractive summarization. Text summarization has various applications including news aggregation, content analysis, and information retrieval.

HOW TEXT SUMMARIZATION PERFORMED USING TRANSFORMERS?

Text summarization using Transformers can be performed in two ways: **extractive summarization** and **abstractive summarization**.

1. **Extractive summarization:** In this approach, the most important sentences or phrases from the original text are selected and combined to form a summary. This can be done using algorithms such as TextRank, which uses graph-based algorithms to rank sentences based on their relevance and importance. Transformers can be used to process the text, extract features, and perform sentence ranking.
2. **Abstractive summarization:** In this approach, a new summary is generated by understanding the context of the original text and generating new phrases and sentences that summarize its content. This can be done using techniques such as encoder-decoder models, where the encoder processes the input text to extract its features and the decoder generates the summary. Transformers can be used as the encoder or decoder in this architecture.

In both extractive and abstractive summarization, Transformers can be trained on large amounts of text data to learn the patterns and relationships between words, sentences, and documents, making them well-suited for text summarization tasks.

USE OF TEXT SUMMARIZATION

Text summarization has various uses, including:

1. **News aggregation:** Summarizing news articles to provide a quick overview of the most important information.
2. **Content analysis:** Reducing large volumes of text data to identify patterns, trends, and insights.
3. **Information retrieval:** Summarizing search results to provide users with a concise and easy-to-understand overview of relevant information.
4. **Document management:** Summarizing long documents to make them easier to manage and search through.
5. **Meeting minutes:** Summarizing the key points discussed in a meeting to provide a concise and organized record.
6. **Customer feedback:** Summarizing customer feedback to identify common themes and issues.
7. **Legal contracts:** Summarizing legal contracts to provide a quick overview of their key terms and conditions.
8. **Customer service:** Summarizing customer support requests to quickly identify the root cause of an issue.

By condensing large amounts of text into a more manageable form, text summarization can help users quickly understand the content of a document and make more informed decisions.

WHY USE TRANSFORMERS FOR TEXT SUMMARIZATION?

Transformers are used for text summarization because they are highly effective in processing and understanding large amounts of text data. There are several reasons why Transformers are a popular choice for text summarization:

1. **Contextual understanding:** Transformers use attention mechanisms to understand the context of words, sentences, and documents, which is crucial for text summarization tasks. This allows Transformers to accurately identify the most important information in a text document.
2. **Large language model:** Transformers have been trained on vast amounts of text data, which enables them to have a deep understanding of language patterns and relationships. This makes them well-suited for text summarization tasks where a comprehensive understanding of language is required.
3. **Scalability:** Transformers can process large amounts of text data in parallel, making them well-suited for summarizing long documents or large volumes of text data.
4. **End-to-end training:** Transformers can be trained end-to-end on text summarization tasks, which allows them to optimize their performance for the specific task at hand.
5. **State-of-the-art results:** Transformers have achieved state-of-the-art results on a wide range of natural language processing tasks, including text summarization.

By using Transformers for text summarization, organizations can benefit from their ability to understand and process large amounts of text data, leading to more accurate and reliable summarization results.

ENOUGH THEORY!!! LET'S CODE...

We will begin by downloading dataset which is the BBC news dataset. This contains nice long articles, there would be nice candidates for summarization task.

```
!wget -nc https://www.dropbox.com/s/7hb8bwbtjmxovlc/bbc_text_cls.csv?dl=0
```

The command will download the file from the URL .

```
# install transformers
!pip install transformers
```

The command **pip install transformers** is used to install the **transformers** package, which provides access to state-of-the-art Transformer-based models for NLP tasks, including Text Summarization.
Once the **transformers** package is installed, you can import and use the Transformer-based models in your own projects.

```
# import required libraries
from transformers import pipeline
import textwrap
import numpy as np
import pandas as pd
from pprint import pprint
```

The code imports the "pipeline" module from the "transformers" library, which is a high-level API for performing natural language processing tasks such as text generation and sentiment analysis. The code also imports the "textwrap" and "numpy" libraries, which are used for text formatting and numerical operations, respectively. Finally, the code imports the "pandas" library and the "pprint" function from the "pprint" library, which are used for data manipulation and pretty-printing, respectively.

Load dataset into dataframe

```
df = pd.read_csv('bbc_text_cls.csv?dl=0')
```

print first few rows of data

```
df.head()
```

	text	labels
0	Ad sales boost Time Warner profit\n\nQuarterly...	business
1	Dollar gains on Greenspan speech\n\nThe dollar...	business
2	Yukos unit buyer faces loan claim\n\nThe owner...	business
3	High fuel prices hit BA's profits\n\nBritish A...	business
4	Pernod takeover talk lifts Domecq\n\nShares in...	business

Choose the random news article

```
doc = df[df.labels == 'business']['text'].sample(random_state=42)
```

```
# text wrapping function
def wrap(x):
  return textwrap.fill(x, replace_whitespace = False, fix_sentence_endings = True)
```

above function returns a copy of x where line breaks are inserted such that each line in the returned string is no longer than a specified number of characters (the default is 70).

The parameters used in the function call are:

- **replace_whitespace** - a boolean that specifies whether to replace consecutive whitespaces in **x** with a single space before wrapping. In this case, it's set to **False**, meaning that whitespaces will be preserved.
- **fix_sentence_endings** - a boolean that specifies whether to ensure that wrapped lines end with a complete sentence. In this case, it's set to True, meaning that the function will try to wrap sentences at a period **(.)** if possible.

Print the news article we have selected

```
print(wrap(doc.iloc[0]))
```

```
Christmas sales worst since 1981

UK retail sales fell in December, failing to meet expectations and making it by some counts the worst
Christmas since 1981.

Retail sales dropped by 1% on the month in December, after a 0.6% rise in November, the Office for National
Statistics (ONS) said.  The ONS revised the annual 2004 rate of growth down from the 5.9% estimated in November to 3.2%. A number of
retailers have already reported poor figures for December.  Clothing retailers and non-specialist stores were the worst hit with only
internet retailers showing any significant growth, according to the ONS.

The last time retailers endured a tougher Christmas was 23 years
previously, when sales plunged 1.7%.

The ONS echoed an earlier caution from Bank of England governor Mervyn King not to read too much into
the poor December figures.  Some analysts put a positive gloss on the figures, pointing out that the non-seasonally-adjusted figures showed
a performance comparable with 2003. The November-December jump last year was roughly comparable with recent averages, although some way
below the serious booms seen in the 1990s.  And figures for retail volume outperformed measures of actual spending, an indication that
consumers are looking for bargains, and retailers are cutting their prices.

However, reports from some High Street retailers highlight the
weakness of the sector.  Morrisons, Woolworths, House of Fraser, Marks & Spencer and Big Food all said that the festive period was
disappointing.
```

Create summarization pipeline object

```
summarizer = pipeline('summarization')
```

The code creates a summarization pipeline from the “transformers” library using the “pipeline” function. The argument “summarization” specifies that the created pipeline will be used for summarizing texts. This pipeline will allow for inputting a text and generating a summarized version of the text as its output. As we haven’t specified any model, by default it uses **distilbart-cnn-12–6 model**. This model uses **Abstractive summarization technique**.

Next pass selected article into the summarizer object

```
summarizer(doc.iloc[0].split('\n',1)[1])
```

```
[{'summary_text': ' Retail sales dropped by 1% on the month in December, after a 0.6% rise in November
. Clothing retailers and non-specialist stores were the worst hit with only internet retailers showing
any significant growth . The last time retailers endured a tougher Christmas was 23 years ago, when
sales plunged 1.7% .'}]
```

summarized text

The code applies the summarization pipeline “summarizer” to the first element of the “doc”.

The first element is extracted from the dataframe using the **iloc** method, which is used for indexing and selecting data from a dataframe based on its position.

The extracted first element is then **split** into two parts using the split method, where the first argument **'\n'** specifies the separator, and the second argument **1** specifies the maximum number of splits. The resulting list only contains two parts, and the second part is selected using indexing **[1]**. Here we are doing this to ignore title of the article.

Finally, the summarized text is generated by passing the selected text as an argument to the summarizer pipeline.

Let’s try on another article. This time we will select entertainment article.

```
doc = df[df.labels == 'entertainment']['text'].sample(random_state=50)
```

Print selected article

```
Ocean's Twelve raids box office

Ocean's Twelve, the crime caper sequel starring George Clooney, Brad Pitt and Julia Roberts, has gone
straight to number one in the US box office chart.

It took $40.8m (£21m) in weekend ticket sales, according to studio estimates.  The
sequel follows the master criminals as they try to pull off three major heists across Europe.  It knocked last week's number one, National
Treasure, into third place.  Wesley Snipes' Blade: Trinity was in second, taking $16.1m (£8.4m). Rounding out the top five was animated
fable The Polar Express, starring Tom Hanks, and festive comedy Christmas with the Kranks.

Ocean's Twelve box office triumph marks the
fourth-biggest opening for a December release in the US, after the three films in the Lord of the Rings trilogy.  The sequel narrowly beat
its 2001 predecessor, Ocean's Eleven which took $38.1m (£19.8m) on its opening weekend and $184m (£95.8m) in total.  A remake of the 1960s
film, starring Frank Sinatra and the Rat Pack, Ocean's Eleven was directed by Oscar-winning director Steven Soderbergh.  Soderbergh returns
to direct the hit sequel which reunites Clooney, Pitt and Roberts with Matt Damon, Andy Garcia and Elliott Gould.  Catherine Zeta-Jones
joins the all-star cast.  "It's just a fun, good holiday movie," said Dan Fellman, president of distribution at Warner Bros.  However, US
critics were less complimentary about the $110m (£57.2m) project, with the Los Angeles Times labelling it a "dispiriting vanity project". A
milder review in the New York Times dubbed the sequel "unabashedly trivial".
```

Summarize selected article

```
summarizer(doc.iloc[0].split('\n',1)[1])
```

```
[{'summary_text': " The crime caper sequel took $40.8m (£21m) in weekend ticket sales . It knocked last week's number one,
National Treasure, into third place . Wesley Snipes' Blade: Trinity was in second, taking $16.1m (£8.4m)"}]
```

This is the summarized texts of 2 lines from given article of 14 lines. This summarization using transformers effectively captures the essence of the original article, providing a comprehensive understanding of its main points and ideas. The use of advanced language processing technology has enabled the creation of a succinct summary that accurately represents the content of the original article. This approach to summarization is a testament to the effectiveness of transformers in transforming complex information into easily digestible formats.