Aim : Concisely communicate the essential information and findings of a text while maintaining clarity and coherence.

Objectives:

Text Preprocessing:

- Cleaning
- Tokenization
- Lowercasing
- Removal of Stopwords
- Lemmatization

Extractive Summarization:

- Build a frequency table
- Sentence Scoring
- Normalisation of Sentences
- Extract the sentences with the maximum score

Abstractive Summarization:

- Tokenization
- Generation of input ids and attention masks
- Generation of summary
- Decoding of tokens

Literature review:

The vast amount of information available on the internet can lead to information overload, making it challenging for individuals to efficiently process and extract relevant insights. Text summarization tools play a crucial role in addressing this issue by condensing large volumes of information into concise summaries. As the internet continues to grow, with approximately 328.77 million terabytes of information added daily, the demand for more powerful and efficient text summarization tools becomes increasingly evident.

These tools not only save valuable time for users but also contribute to resource optimization by reducing the need to sift through extensive amounts of data. By extracting key information and presenting it in a condensed form, summarization tools enhance comprehension and decision-making processes.

Developing and enhancing text summarization algorithms will be essential to keep up with the ever-expanding volume of information on the internet. This technological advancement is crucial for individuals, businesses, and researchers who rely on quick access to relevant information without being overwhelmed by the sheer quantity available.

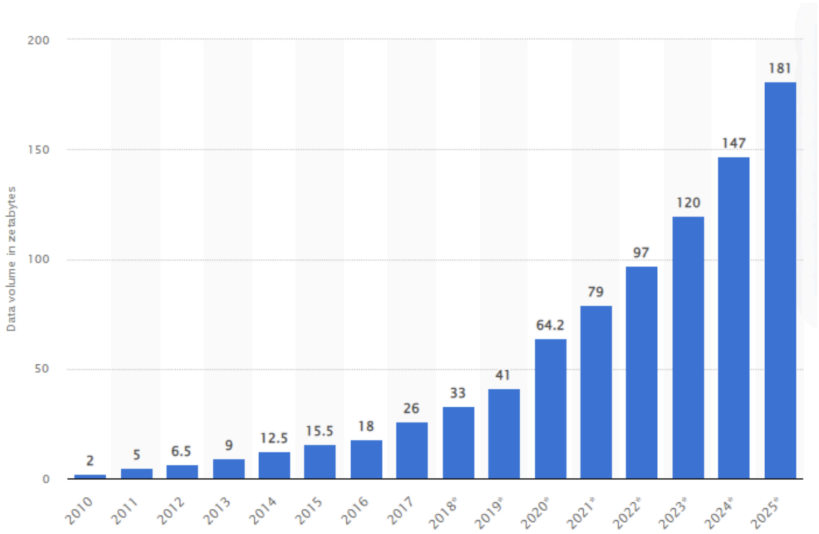

Fig 1: Volume of data/information created, captured, copied, and consumed worldwide from 2010 to 2025 (Arne von See, 2021)

[1]

There are two types of Summarization models:

a) **Extractive summarization:** These models operate by selecting and extracting sentences directly from the original text to create a summary. Here are some advantages and disadvantages of extractive summarization models:

Advantages:

1) **Preservation of Meaning:** Extractive models maintain the meaning of the original text by directly using existing sentences in the summary. This ensures that the key information is retained accurately.
2) **Compatibility with Keyword Extraction:** Extractive summarization can seamlessly be used in keyword extraction algorithms since it relies on selecting and using existing sentences.
3) **Measurable Importance of Sentences:** The importance of sentences within the text can be quantified in extractive models, aiding in understanding which sentences contribute more significantly to the overall meaning.

Disadvantages:

1) **Inability to Generate New Sentences:** Extractive models are limited to using sentences exactly as they appear in the original text. They cannot generate new sentences, potentially limiting the creativity and variety in the summary.
2) **Whole Number of Sentences Only:** Extractive models can only select complete sentences from the original text. This might result in suboptimal summarization, as key information could be spread across partial sentences or require a more concise representation.[2]

b) **Abstractive Summarization:**

- Abstractive summarization models, resembling human summarization, employ paraphrasing and semantic comprehension to condense the original text.
- Instead of selecting and extracting sentences verbatim, they aim to understand the underlying meaning and then generate novel sentences that effectively capture the essence of the content.
- Similar to humans, these models create a semantic representation or "picture" of the information and then use this understanding to craft concise and coherent summaries. This involves not only selecting appropriate words but also formulating new sentences that may not exist in the original text.

Advantages:

1. **Enhanced Clarity and Readability:** Abstractive summarization models have the capability to improve the overall clarity and readability of the summary. By using better words and constructing more refined sentences, they can present the information in a way that is not only concise but also easier for the audience to comprehend.
2. **Adaptability to Different Styles:** These models can adapt to various writing styles and tones, allowing the generated summary to match the desired style or tone, which may be different from the original text.
3. **Handling Redundancy:** Abstractive models can mitigate redundancy by rephrasing repetitive information in a way that maintains the core meaning but avoids unnecessary duplication.
4. **Contextual Understanding:** They demonstrate a level of contextual understanding, enabling them to capture nuances and context-specific details that may be crucial for conveying the complete meaning of the text.

Disadvantages:

1. **Risk of Introducing Inaccuracies:** Abstractive models may occasionally generate sentences that deviate from the intended meaning of the original text, leading to inaccuracies in the summary.
2. **Difficulty in Handling Ambiguity:** Ambiguous phrases or context-dependent information in the original text can pose challenges for abstractive models, as they might struggle to accurately interpret and represent such content.
3. **Computational Intensity:** Generating abstractive summaries often requires more computational resources compared to extractive methods. This can lead to longer processing times and increased computational costs.

4. **Lack of Transparent Logic:** Understanding the decision-making process of abstractive models can be challenging, as the logic behind the generation of specific phrases or sentences may not be easily interpretable. This lack of transparency can be a drawback in certain applications.
5. **Need for Large Training Datasets:** Abstractive models typically benefit from extensive training datasets to effectively learn language nuances. Obtaining and processing such large datasets can be resource-intensive.
6. **Potential for Biases:** If the training data contains biases, abstractive models may inadvertently amplify or propagate those biases in the generated summaries.
7. **Handling of Uncommon Terms:** Abstractive models might struggle with uncommon or domain-specific terms that are not well-represented in their training data, potentially resulting in less accurate summaries for specialised content.

[3]

Methodology:

Text Preprocessing

1. **Cleaning**: Remove any irrelevant or noisy information, correct any typos or errors.
   Tokenization: Split the text into sentences or words.
2. **Lowercasing**: Convert all text to lowercase to ensure consistency.
3. **Removal of Stopwords:** Remove common words that do not add to the meaning of a sentence.
4. **Stemming or Lemmatization:** Conversion of word to its base form.
   Extractive Summarization:
5. **Sentence Scoring:** Assign scores to each sentence based on importance.
6. **Sentence Selection:** Select top-scoring sentences to form the extractive summary.
7. **Tools:** Utilise libraries like Hugging Face and Pytorch for extractive summarization.
   Interactive Summarization:
8. **User Input:** Allow the user to interactively select key sentences or phrases.
9. **Dynamic Summarization:** Update the summary based on user input.
10. **Tools:** Depending on the complexity, you may use a simple interface or more advanced techniques.

Extractive Summarization:

- **Build a Frequency Table:**The first step involves creating a frequency table that keeps track of the occurrence of each word in the document. This table helps identify words that are significant and those that appear frequently. Typically, common words like articles and prepositions are excluded (stop words) to focus on more meaningful terms.

- **Sentence Scoring:**Once the frequency table is created, the next step is to score each sentence in the document. The scoring is based on the importance of the words within each sentence. Sentences with words that have higher frequencies in the frequency table are considered more important. This scoring mechanism helps in identifying key sentences that carry essential information.
- **Normalisation of Sentences:** Normalisation involves adjusting the scores to account for the length of sentences. Longer sentences may naturally have more occurrences of important words, so their scores need to be normalised to avoid bias toward longer sentences. One common normalisation method is dividing the score of each sentence by the total number of words in that sentence.
- **Extract the Sentences with the Maximum Score:** After scoring and normalisation, the sentences are ranked based on their scores. The sentences with the highest scores are then extracted to form the final summary. These sentences are considered the most representative and informative parts of the original text.

Abstractive Summarization:

- **Tokenization:**Tokenization is the process of breaking down a piece of text into individual units, or tokens. Tokens are usually words or subwords. This step involves converting the input text into a sequence of tokens. This is a crucial step as it forms the basis for the model to understand and process the input text.
- **Generation of Input IDs and Attention Masks:**After tokenization, each token is mapped to a unique identifier known as an input ID. Additionally, attention masks are generated to indicate which parts of the input are important for the model to pay attention to during processing. These input IDs and attention masks serve as the model's input for training or generating summaries.
- **Generation of Summary:**Using the input IDs and attention masks, the model generates a summary. The model is trained to understand the relationships between tokens and to generate coherent and contextually appropriate summaries. This step involves the model predicting the next tokens that form the summary based on its understanding of the input.
- **Decoding of Tokens:** The generated summary is initially in the form of token IDs. The decoding step involves converting these token IDs back into human-readable text. This is crucial to obtain the final abstractive summary that can be easily understood.

Conclusion:

The fusion of extractive and abstractive summarization methods is pivotal in navigating the expansive digital terrain. Striking a balance between fidelity and innovation is essential as technology evolves. The user-centric approach remains paramount, ensuring that summarization tools effectively meet the diverse needs amid the sea of information. The future lies in harmonising innovation, ethics, and user-centric design, providing users with powerful tools to extract clarity from the overwhelming digital landscape.

References:

[1]: Yadav, Divakar , et al. "ORCID." *Arxiv*, orcid.org/0000-0001-6051-479X. Accessed 29 Feb. 2024.

[2]: Bhargava, Rupal, and Yashvardhan Sharma. "Deep Extractive Text Summarization." Procedia Computer Science, vol. 167, 2020, pp. 138–146, https://doi.org/10.1016/j.procs.2020.03.191.

[3]: Roy, Abhijit. "Understanding Automatic Text Summarization-2: Abstractive Methods." Medium, 7 Aug. 2020, towardsdatascience.com/understanding-automatic-text-summarization-2-abstractive-methods-7099fa8656fe.