Project Report
Text Summarization

1. Abstract

In today's world, with the abundance of written information, it's essential to be able to summarize the text in a concise and meaningful way. Our solution to this challenge is an Abstractive Text Summarizer that uses the CNN/Daily Mail dataset. A text summarizer creates a shorter version of a longer text while retaining its key information and meaning. The goal of text summarization is to condense a large body of text into a shorter, more manageable form, without losing the essence of the original content. Our system relies on a seq2seq encoder-decoder model with LSTM and attention mechanism. We also utilize pre-trained GloVe embeddings from the 42B model to generate embedding vectors and enhance the summarization process. To evaluate the effectiveness of our summarizer, we use the BLEU score, a well-known metric for assessing the quality of machine-generated summaries. Our ultimate goal is to create concise and informative summaries, making it easier for users to extract key insights from large amounts of textual information.

2. Introduction

With the advent of digital connectivity, people are constantly bombarded with an enormous amount of information daily. From news articles and research papers to social media updates and online content, there is an overwhelming volume of written material to sift through. As a result, it can be challenging to consume and understand all the information available effectively. Text summarization has emerged as a critical technique to tackle this challenge. The goal of text summarization is to distill the essential meaning and key points of a document or piece of text into a concise summary. By condensing the information, text summarization allows individuals to grasp the main ideas and extract relevant insights without having to read through the entire document.

Abstractive summarization holds significant importance as it enables the generation of concise and coherent summaries that go beyond the mere extraction of sentences. By understanding the context, semantic meaning, and relationships within the text, abstractive summarization can create human-like summaries that provide a comprehensive understanding of the original content. This makes abstractive summarization an invaluable tool for information retrieval, document organization, and aiding individuals in quickly grasping the key points from large volumes of text. In this project, we aim to leverage the power of abstractive summarization to provide users with accurate and concise summaries that facilitate efficient information consumption.

3. Types of Text Summarizers

Text summarizers can be classified into two broad categories: extractive and abstractive summarization.

Extractive summarization involves selecting the most relevant sentences or phrases from the original text and presenting them in a condensed form as the summary. The extracted sentences are chosen based on their relevance to the main ideas of the text and their ability to convey the overall message of the text. This method requires minimal language generation and relies on existing language within the original text.

Abstractive summarization, on the other hand, involves generating a new summary that may contain words and phrases that do not appear in the original text. This method requires more sophisticated natural language processing techniques and a deeper understanding of the meaning and context of the original text. Abstractive summarization is more difficult to implement than extractive summarization, but it has the potential to generate more accurate and informative summaries.

Other Classification Types:

- Based on the summarization approach:
    - Extractive
    - Abstractive
    - Hybrid
- Based on input:
    - Single-document
    - Multi-document
- Based on purpose:
    - Generic
    - Domain-specific
- Based on the length of the summary:
    - Sentence-level
    - Paragraph-level
    - Document-level
- Based on the type of ML algorithm:
    - Neural network-based
    - Decision tree-based
    - Support vector machine-based
- Based on output format:

- Text-based
- Graphical

4. Dataset and its Preprocessing

The CNN/Daily Mail dataset is a widely used benchmark dataset for text summarization tasks. It comprises news articles from the CNN and Daily Mail news websites, along with corresponding human-written summaries. The dataset includes over 300,000 articles and summaries and is annotated with additional information such as the article's heading, publication date, and article ID.

As part of the preprocessing steps for our abstractive text summarizer, we performed the following tasks:

- Sentence Segmentation: We used the NLTK library to split the raw text into individual sentences. This step allows us to process and analyze the text at a sentence level.
- Tokenization: Each sentence was tokenized, breaking it down into individual words or subword units. Tokenization enables the model to process and understand the text at a more granular level.
- Removing Punctuation + Converting to Lowercase: We removed punctuation marks from the text, such as periods, commas, and quotation marks. Additionally, we converted all words to lowercase. This ensures consistency in word representations and reduces vocabulary size.
- Stopword Removal: Stopwords are common words that do not carry significant meaning, such as "the," "and," or "a." We removed these stopwords from the text as they do not contribute to the overall meaning of the sentences.
- Stemming: We applied the Snowball algorithm for stemming, which involves removing suffixes from words. This step helps to reduce words to their base form by removing variations such as "ed," "er," "tion," and so on.
- Lemmatization: For lemmatization, we utilized the WordNetLemmatizer algorithm. Lemmatization transforms words into their base form, such as converting "went" to "go." This step ensures that different inflections of words are mapped to their common base form.
- Removing Special Characters and Numbers: We used regular expressions (regex) to remove special characters and numbers from the text. Special characters, such as hashtags or asterisks, and numerical digits are removed to focus on the textual content.

By applying these preprocessing steps, we cleaned and standardized the text data, making it more suitable for training our abstractive text summarizer. These steps help to improve the

quality of the generated summaries by reducing noise, standardizing text representations, and focusing on the most meaningful content.

5. Methodology

Integration of the pre-trained GloVe embeddings (42B model): GloVe embeddings provide pre-trained word vectors that capture semantic and syntactic information from large text corpora. In our approach, we utilize the 42B version of the GloVe model, which has been trained on a massive corpus containing billions of words. By integrating these pre-trained word embeddings, we can enhance the quality of our model's representation of words and improve its understanding of the input text.

During the preprocessing step, each word in the input sequence is mapped to its corresponding pre-trained GloVe embedding vector. These embeddings serve as the initial input for the encoder and decoder. By leveraging the semantic information captured by GloVe embeddings, our model can better capture the meaning and context of words in the input sequence, leading to more accurate and coherent summaries.

A detailed explanation of the seq2seq encoder-decoder model architecture:

The seq2seq encoder-decoder model is a popular framework for sequence-to-sequence tasks like text summarization. It consists of two main components: an encoder and a decoder. The encoder processes the input sequence (source text) and encodes it into a fixed-length representation called the context vector. The decoder takes the context vector as input and generates the output sequence (summary). The encoder typically uses recurrent neural networks (RNNs), such as Long Short-Term Memory (LSTM) or Gated Recurrent Unit (GRU), to capture the sequential dependencies in the input text. Each word in the input sequence is embedded into a continuous vector representation and fed into the encoder RNN. The final hidden state of the encoder RNN serves as the context vector, summarizing the input text. The decoder, also an RNN, takes the context vector as its initial hidden state and generates the output sequence word by word. At each time step, the decoder predicts the next word based on the context vector and the previously generated words. This process continues until an end-of-sequence token is generated or a predefined maximum length is reached.

Incorporation of LSTM and attention mechanism:

To enhance the capabilities of the seq2seq model, we incorporate LSTM units within the encoder and decoder. LSTM networks are a variant of RNNs that address the vanishing gradient problem by introducing memory cells and gating mechanisms. The LSTM units allow the model to capture long-range dependencies and maintain a more robust representation of the input text.

Additionally, we employ an attention mechanism in our model. The attention mechanism enables the decoder to focus on different parts of the input sequence during the decoding process. It assigns weights to the hidden states of the encoder based on their relevance to the current decoding step. By attending to different parts of the input sequence, the attention mechanism helps the model generate more accurate and context-aware summaries.

6. Results

The blue score is a commonly used metric for evaluating the quality of machine-generated text summaries. It measures the similarity between the generated summary and a set of reference summaries, and ranges from 0 to 1, with higher scores indicating better quality. Our text summarizer model achieved a blue score of 0.3, which indicates that there is room for improvement in terms of summary quality.

Below is the comparison of the original summary and the output of the predicted summary. The predicted summary is somewhat similar to the original summary.

Original summary: mary whitaker 61 found murdered gunshot wound chest wednesday garage home westfield new york jonathan conklin 43 30 year old charles sanford charged murder police said violinist spent year new york city had forced give men information killed home authorities believed ms whitaker targeted suspects isolated home

Predicted summary: charles woodburn shot dead scene home friday afternoon charles hosting bed shot dead home sunday afternoon 57 year old victim separate incidents burglary recovered

7. Applications and Future Directions

The abstractive text summarizer developed in this project has potential applications in various domains, including news aggregators, content summarization platforms, and document management systems. Future research may involve exploring reinforcement learning techniques for training the model and incorporating domain-specific knowledge to further enhance summarization capabilities.

Use Cases -

- News articles and press releases
- Business reports
- Academic papers
- Social media posts

- Customer reviews for businesses
- Long emails or messages

The abstractive text summarizer developed in this project has significant potential in various applications. It can be integrated into news aggregators, content summarization platforms, and document management systems, enabling users to quickly extract key information from large volumes of text. Future directions for research may include exploring reinforcement learning techniques for training the model and incorporating domain-specific knowledge to further enhance its summarization capabilities.

The potential applications of our abstractive text summarizer are diverse and impactful. It can be integrated into news aggregators, content curation platforms, and document management systems, enabling users to quickly grasp essential information from large volumes of text. Furthermore, future research directions may include exploring reinforcement learning techniques for model training and incorporating domain-specific knowledge to enhance summarization capabilities in specialized domains.