# Final Project Report

# Text Summarization Using Deep Learning

## 1. Abstract

This project explores the development of a neural networks-based text summarization model aimed at generating concise, contextually relevant summaries from long customer reviews. The primary objective is to create an abstraction-based summarization tool using a Sequence-to-Sequence (Seq2Seq) model with Long Short-Term Memory (LSTM) layers, incorporating an attention mechanism for enhanced context understanding.

The dataset used consists of Amazon Fine Food reviews, containing over 100,000 samples with text and corresponding summaries. The methodology involves preprocessing text, tokenizing, padding, and building an attention-based Seq2Seq model to generate the summaries. Key findings include the model's ability to summarize positive sentiment reviews effectively, producing accurate and concise outputs.

Results show that the model is highly accurate for simpler, positive reviews but struggles with more complex or nuanced content. The project provides valuable insights into building text summarization systems and proposes potential areas for improvement, such as using Bi-Directional LSTM, beam search strategies, and pointer-generator networks.

---

## 2. Introduction

Text summarization plays a critical role in Natural Language Processing (NLP), especially for processing large volumes of unstructured text, such as product reviews, news articles, and social media content. Businesses and consumers alike benefit from summarization tools that can distill long text into short, insightful summaries.

This project focuses on building an abstractive text summarization model, using a Seq2Seq architecture with LSTM layers. The challenge of summarizing customer reviews, especially for a platform like Amazon, lies in maintaining the essence of the text while making it shorter and more readable. The project proposes an innovative solution using an attention mechanism to dynamically focus on relevant parts of the input sequence, enhancing the decoder's performance in generating summaries.

The key novelty of this approach is the integration of an attention mechanism, which allows the model to focus on important words or phrases from the input during the summarization process.

---

## 3. Related Works

The field of text summarization has been broadly divided into two categories: extractive and abstractive summarization. Extractive summarization works by selecting key sentences directly from the input text. In contrast, abstractive summarization involves generating new sentences that convey the meaning of the original text more concisely.

One of the landmark approaches in abstractive summarization is the Sequence-to-Sequence (Seq2Seq) model, which forms the backbone of this project. Seq2Seq models use an encoder-decoder architecture where the encoder processes the input sequence and compresses it into a context vector, and the decoder uses this context to generate the output sequence.

Bahdanau et al. (2015) introduced the attention mechanism, a crucial enhancement to the Seq2Seq model. The attention mechanism allows the model to weigh different parts of the input sequence differently when generating each word in the output sequence. This focus on specific parts of the input makes the model more effective, especially for longer sequences, and is incorporated in this project.

Recent advancements like the Pointer-Generator Networks (See et al., 2017) combine extractive and abstractive summarization methods. These models can both copy portions of the input sequence and generate new content. In this project, attention is used to enhance the Seq2Seq model's ability to focus on relevant information dynamically.

---

## 4. Preliminary/Background

To implement the text summarization model, it's essential to understand the following core concepts:

1. **Seq2Seq Model**
   A Seq2Seq model maps an input sequence to an output sequence. It consists of:
   - **Encoder:** Converts the input sequence into a fixed-length context vector.
   - **Decoder:** Uses the context vector to generate the output sequence, token by token.
2. **Attention Mechanism**
   The attention mechanism allows the decoder to focus on relevant parts of the input sequence by dynamically calculating weights for encoder outputs. This enables the model to pay attention to different parts of the input for each output token.
3. **LSTM (Long Short-Term Memory)**
   LSTM is a type of RNN effective in learning long-term dependencies. It is used in both the encoder and decoder layers to capture contextual information for generating accurate summaries.
4. **Tokenization and Padding**
   Tokenization breaks the text into smaller units (tokens), which are mapped to integers. Padding ensures uniform sequence length, crucial for batch processing during model training.
5. **Contraction Mapping for Data Preprocessing**
   Many reviews contained contractions (e.g., "don't," "can't"), which could introduce inconsistencies. By expanding these contractions using a custom mapping dictionary, text consistency is improved, leading to better tokenization and model performance.

## 5. Methodology

This project aims to generate abstractive summaries of Amazon Fine Food reviews using a Seq2Seq model with LSTM layers and an attention mechanism. Below is a concise overview of the methodology.

### 5.1 Data Preprocessing

The preprocessing steps ensure clean, consistent input data:

- **Text Cleaning**: HTML tags, punctuation, stopwords, and short words were removed to ensure that only meaningful content is processed.
- **Tokenization**: Reviews and summaries were tokenized using Keras' Tokenizer to convert words into integer sequences.
- **Padding**: Sequences were padded to a uniform length (30 words for text, 8 for summaries) for efficient batch processing and uniform input size.
- **Rare Word Filtering**: Words appearing less frequently than a set threshold were discarded to reduce vocabulary size and focus on more relevant terms.

### 5.2 Model Architecture

The core model is a Sequence-to-Sequence (Seq2Seq) architecture:

- **Encoder**: The input review is processed through an embedding layer (size: 100) and multiple stacked LSTM layers (300 hidden units each), producing a context vector representing the entire review. Dropout (0.4) and recurrent dropout (0.4) were applied to prevent overfitting.
- **Decoder**: A single LSTM layer (300 hidden units) generates the summary, with the context vector from the encoder as its initial state, predicting one word at a time.

### 5.3 Attention Mechanism

The attention mechanism enables the model to focus on different parts of the input sequence during each decoding step:

- **Score Calculation**: At each decoding step, attention scores are calculated for each encoder output, based on the current state of the decoder.
- **Weighted Sum**: The attention scores are normalized using the softmax function to compute a weighted sum of the encoder's outputs, which serves as the context vector for the decoder, allowing it to focus on relevant parts of the input sequence.

### 5.4 Training Details

- **Optimizer**: RMSProp was chosen to adjust the learning rate dynamically, helping the model converge more quickly without overshooting.
- **Loss Function**: Sparse categorical cross-entropy was used, appropriate for tasks with a vocabulary distribution in sequence prediction.
- **Metrics**: Accuracy and loss were monitored during training to ensure the model was learning effectively.
- **Batch Size**: A batch size of 128 was used, allowing efficient training by processing multiple sequences simultaneously.

- **Early Stopping**: To avoid overfitting, training was stopped after 19 epochs, based on the validation loss not improving for 2 consecutive epochs.

**5.5 Evaluation**
The model's performance was evaluated based on accuracy and loss metrics, with attention to loss curves during training. Early stopping was used to prevent overfitting.

**5.6 Model Summary**
The model consists of 4,823,389 trainable parameters, utilizing stacked LSTM layers in the encoder and an LSTM with an attention mechanism in the decoder. This architecture effectively generates relevant, context-aware summaries of the reviews.

## 6. Numerical Experiments

### Key Metrics

- **Training and Validation Loss**: The training loss decreased steadily, indicating that the model was learning effectively. However, the validation loss plateaued after epoch 17, suggesting the model may have started to overfit.
- **Results**
- The training progress is summarized in the table below:

| Epoch | Training Loss | Validation Loss | Observation |
|---|---|---|---|
| 1 | 2.8152 | 2.5780 | Initial learning phase. |
| 5 | 2.1604 | 2.1862 | Significant improvement. |
| 10 | 1.9476 | 2.0636 | Stable convergence begins. |
| 17 | 1.7518 | 2.0374 | Validation loss plateaus. |
| 19 | 1.7070 | 2.0398 | Early stopping triggered. |

### Evaluation

**Based on the given reviews and their summaries, here's an evaluation of the model's predicted summaries:**

Best Predictions: Review: "really like product super easy order online delivered much cheaper buying gas station stocking good long drives"

Predicted Summary: "great product"

Evaluation: This prediction is a correct and relevant summary of the user's positive experience with the product.

Review: "husband gluten free food several years tried several different bread mixes first actually enjoys buying amazon saves loaf"

Predicted Summary: "great gluten free bread"

Evaluation: The predicted summary is accurate as it summarizes the positive experience with the gluten-free bread.

Review: "absolutely loves apple chicken happy hips looks forward one morning one night gets soooo excited would eat allowed"

Predicted Summary: "great for training"

Evaluation: The prediction works well, as it highlights the dog's excitement for the treats, which can be an indicator of their suitability for training.

Review: "great toy dogs chew everything else little literally eats toys one toys yet destroy loves carries around everywhere got rex cutest thing"

Predicted Summary: "dogs love it"

Evaluation: A good summary capturing the dog's positive interaction with the toy.

## 7. Conclusion

This project demonstrates the power of **Recurrent Neural Networks (RNNs)**, particularly **LSTM-based Seq2Seq models** with an integrated **attention mechanism**, for generating precise and meaningful summaries from customer reviews. The model leverages **stacked LSTM layers** in both the encoder and decoder to effectively process sequential data, capturing temporal patterns and dependencies. Attention mechanisms dynamically focus on relevant input segments, improving the model's contextual understanding and output accuracy. Regularization techniques and early stopping ensured robust training while preventing overfitting. Future enhancements, such as incorporating **Transformer models** or integrating additional linguistic features, could further refine performance for complex or nuanced reviews.

### Limitations:

- The model tends to oversimplify longer or more detailed reviews, missing some nuanced information.
- The focus on positive sentiment reviews may limit the model's ability to generalize to other types of reviews.

### Future Directions:

- **Bi-Directional LSTM**: Implementing Bi-Directional LSTM could enhance context understanding by processing sequences in both forward and backward directions.
- **Pointer-Generator Networks**: Incorporating pointer-generator networks could allow the model to copy words directly from the input, improving the generation of more accurate and contextually relevant summaries.
- **Beam Search**: Implementing beam search decoding could improve summary quality by exploring multiple candidate sequences and selecting the best one.

.