ASSIGNMENT - 6

1. What are Vanilla autoencoders

ANs: These are the basic form of autoencoders. They consist of an encoder that compresses the input data into a latent representation (often called the code) and a decoder that reconstructs the original data from the code. The goal is for the decoder's output to be as close to the original input as possible. This process helps the autoencoder learn efficient data encodings.

2. What are Sparse autoencoders

Ans: These autoencoders introduce sparsity constraints on the code, meaning they encourage the code activations to be mostly zeros with only a few non-zero values. This promotes learning more distributed and compact data representations, potentially leading to better generalization.

3. What are Denoising autoencoders

Ans: Designed to improve robustness to noise and irrelevant information. They are trained with corrupted versions of the input data. The encoder learns to extract the essential features from the corrupted input, and the decoder aims to reconstruct a clean version of the original data. This process helps the autoencoder learn to denoise and identify important patterns in the data.

4. What are Convolutional autoencoders

Ans: When dealing with data like images or time series that have spatial or sequential structure, convolutional autoencoders are a good choice. They utilize convolutional layers in the encoder to capture these spatial or sequential features effectively, leading to more efficient representations for such data types.

5. What are Stacked autoencoders

Ans: These involve stacking multiple autoencoders one after another. Each layer acts as an encoder for the previous layer's code. This allows for a hierarchical learning process, where the autoencoders progressively extract higher-level features from the data, potentially leading to more complex and informative representations.

6. Explain how to generate sentences using LSTM autoencoders

Ans: Long Short-Term Memory (LSTM) networks are a type of recurrent neural network (RNN) well-suited for sequential data like sentences. Here's a possible approach:

* Train an LSTM autoencoder on a large corpus of text data.
* During generation, provide a seed sequence (a short starting phrase) to the encoder.
* The encoder generates a code representation of the seed sequence.
* Feed the code through the decoder's LSTM layers, one step at a time, predicting the next word in the sequence at each step. This iterative process continues until a complete sentence is generated.

7. Explain Extractive summarization

Ans: This technique aims to identify and extract the most important sentences from a document to create a shorter summary. It often relies on sentence scoring methods that consider factors like sentence position, word frequency, or keywords to determine sentence importance.

8. Explain Abstractive summarization

ANs; This approach goes beyond simply extracting existing sentences. It involves understanding the document's meaning and then rephrasing it to create a concise summary that captures the main points using different words and sentence structures. Abstractive summarization often employs deep learning models like LSTMs or transformers.

9. Explain Beam search

Ans: This is a search algorithm commonly used in machine translation and text summarization. It maintains a set of top-scoring partial sequences (hypotheses) during decoding. At each step, it expands the most promising sequences based on a scoring function that considers both the current sequence's relevance and the likelihood of generating a good continuation.

10. Explain Length normalization

Ans: In text summarization evaluation metrics like ROUGE (see below), length normalization is often applied. This adjusts the scores to account for differences in summary length between the generated summary and a reference summary. It aims for a fairer comparison by penalizing summaries that are significantly shorter or longer than the reference.

11. Explain Coverage normalization

ANs: This is another technique used in summarization evaluation metrics. It focuses on how well the generated summary covers the important aspects of the source document. It might reward summaries that mention key entities or facts more frequently, even if the wording differs from the reference.

12. Explain ROUGE metric evaluation

Ans: ROUGE is a popular set of metrics used to evaluate the quality of automatic summaries. It compares a generated summary to one or more reference summaries. Different ROUGE variants exist, each focusing on different aspects of similarity, such as:

* ROUGE-N: Measures N-gram (sequence of N words) overlap between the summaries.
* ROUGE-L: Considers the longest matching sequence of words.
* ROUGE-W: Focuses on word-level overlap between the summaries.