ANSWERS

1. The fundamental difference between a feedforward neural network and a recurrent neural network (RNN) lies in their architecture and how they handle sequential data. In a feedforward neural network, information flows in one direction, from input nodes through hidden layers to output nodes, without feedback loops. In contrast, a recurrent neural network has connections that form directed cycles, allowing it to maintain information about past inputs and process sequential data. This recurrent structure enables RNNs to model temporal dependencies in sequential data, making them suitable for tasks such as time series prediction, language modeling, and sequential data generation.

2. The vanishing gradient problem in recurrent neural networks (RNNs) refers to the phenomenon where gradients become increasingly small as they propagate backward through time during training, leading to slow or stalled learning. This problem arises because of the multiplicative nature of gradients in RNNs, which can cause them to exponentially decay as they propagate through long sequences. As a result, RNNs struggle to capture long-term dependencies in sequential data. Techniques to mitigate the vanishing gradient problem include using alternative activation functions like ReLU, employing gradient clipping to limit the magnitude of gradients, and using architectural modifications such as gated recurrent units (GRUs) or long short-term memory (LSTM) cells, which are designed to better preserve gradient information over long sequences.

3. Part-of-speech tagging is the process of automatically assigning a grammatical category (such as noun, verb, adjective, etc.) to each word in a sentence, based on its context and syntactic role within the sentence. It is an important task in natural language processing (NLP) because it provides valuable linguistic information about the structure and meaning of text, which is essential for many downstream NLP tasks, including parsing, information extraction, machine translation, and sentiment analysis.

4. Named entity recognition (NER) systems aim to identify various types of named entities in text, including:
   - Person names (e.g., "John Smith", "Barack Obama")
   - Organization names (e.g., "Google", "United Nations")
   - Location names (e.g., "New York", "Mount Everest")
   - Date and time expressions (e.g., "January 1st, 2022", "2:00 PM")
   - Numerical quantities (e.g., "three million dollars", "100 kilograms")
   - Miscellaneous entities (e.g., "product names", "event names")

5. An encoder-decoder architecture is a neural network model composed of two main components: an encoder and a decoder. In the context of sequence-to-sequence tasks, such as machine translation or text summarization, the encoder processes an input sequence (e.g., a source sentence) and encodes it into a fixed-length vector representation called a context vector. The decoder then takes this context vector as input and generates an output sequence (e.g., a target sentence) by predicting one token at a time, conditioning on the previously generated tokens and the context vector. Encoder-decoder architectures, often enhanced with attention mechanisms, enable the model to handle variable-length input and output sequences and capture long-range dependencies in the data, making them effective for tasks requiring sequence-to-sequence mapping.