UNIT 8 ASSIGNMENT

Natural Language Processing

## Instructions

The questions below will prepare you for future interviews as they relate to concepts discussed throughout the week. You’ve practiced these concepts in the coding activities, exercises and coding portion of the assignment. Now, let’s formulate your programming into well-thought responses.

Except as indicated, use this document to record all your assignment work and responses to any questions. At a minimum, you will need to turn in a digital copy of this document to your facilitator as part of your assignment completion. You may also have additional supporting documents that you will need to submit. Your facilitator will provide feedback to help you work through your findings.

**Note:** Though your work will only be seen by those grading the course and will not be used or shared outside the course, you should take care to obscure any information you feel might be of a sensitive or confidential nature.

*Begin your assignment by completing the questions below. Directions to submit your work can be found on the assignment page. Information about the grading rubric is available on any of the course assignment pages online. Do not hesitate to contact your facilitator if you have any questions about the assignment.*

Unit 8 Written Portion

# Implementing NLP Tasks

Answer the questions below about natural language processing.

## Questions:

1. What is NLP? What are real-world applications of NLP?

|  |
| --- |
| NLP stands for the branch of AI called natural language processing, where computers understand text and verbal words and phrases (language). Real-world applications of NLP are email spam filters, chat/ voice bots, autocorrect, and translations like Google Translate. |

1. Why and how do we have to transform raw text data for NLP tasks? Provide some examples of commonly used techniques in the NLP pipeline.

|  |
| --- |
| We have to transform raw text data for NLP tasks because raw data is too big and contains unnecessary words that provide no predictive value. Commonly used techniques in the NLP pipeline for this data preprocessing techniques are lemmatization, n-gram creation, and stop word removal. Stop word removal reduces the data set size by removing stop words such as “and”, “is” and “the”. Lemmatization is converting a word to its canonical form. For example, “cats” becomes cats, but not “cats” becoming “tiger”. N-gram creation is another process which reduces our vocabulary size by breaking down something like a sentence into smaller and more important parts. |

1. What is TF-IDF? And how is it calculated?

|  |
| --- |
| TF-IDF stands for term frequency-inverse document frequency. It is a type of vectorization counts word frequencies to determine which word/token is more important to understanding a document, for example. It is calculated by counting the number of times a token appears in the document and divide this count by the frequency of the token as it appears across all documents. |

1. What is the difference between vectorizers and word embeddings?

|  |
| --- |
| Word embeddings are similar to vectorizers because both convert text into numerical representations. The difference is that word embeddings capture the meanings of words within a body of text, while vectorizers convert words into numbers based on some predetermined rules, just in with TF-IDF. |

1. What is the difference between a traditional neural network and a sequence-to sequence model? Why should sequence-to-sequence models be used in NLP? Explain the components of a sequence-to-sequence model.

|  |
| --- |
| The difference between a traditional neural network and a sequence-to sequence model is that one is better than the other model to handle text data, and that model is a sequence-to-sequence model. A sequence-to-sequence model is used in NLP because it can determine the meaning of words, given context and sequence. There are two main components to the said model – an encoder and decoder. The encoder is given a sequence of words and outputs a vector or code which gives a summary of the input. A decoder takes in the encoder’s output and outputs a scalar or sequence of outputs, that can be represented by things such as word embeddings. |

1. Compare and contrast a deep averaging network to a recurrent neural network.

|  |
| --- |
| Both deep averaging networks and recurrent neural networks are types of sequence-to-sequence models and are trained with SGD. A deep averaging network is the simplest form of sequence-to-sequence models, but they ignore the sequence of words and ultimately change the meanings of sentences. Recurrent neural networks (RNNs) on the other hand, use a hidden state variable to encode sequential information. Now, the components of deep averaging networks are a traditional neural network (decoder) and a word embedding (encoder). |



*To submit this assignment, please refer to the instructions in the course*.