UNIT 7 ASSIGNMENT

Use ML for Text Analysis

## 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 7 Written Portion

# Choosing Your Model

Answer the questions below about using text as data and word embedding.

## Questions:

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

|  |
| --- |
| Natural Language Processing (NLP) is a branch of artificial intelligence that focuses on enabling computers to understand, interpret, and generate human language. Its applications are wide-ranging and have practical uses in various fields. Some key NLP applications include machine translation, sentiment analysis, chatbots and virtual assistants, information extraction, text summarization, spam filtering, language understanding, speech recognition, language generation, medical text analysis, and social media analysis. NLP plays a crucial role in making computers interact with humans more naturally and effectively, and it has numerous real-world applications that impact our daily lives. |

1. Why and how do we have to transform features for NLP tasks? Provide some examples of commonly used techniques.

|  |
| --- |
| Feature transformation in NLP tasks is the process of converting raw text data into numerical representations suitable for machine learning algorithms. This transformation is necessary because machine learning models work with numerical data, while natural language data is in the form of text. The main reasons for feature transformation are to provide numerical representation for text data and to extract meaningful information from the text.  Commonly used techniques for feature transformation in NLP include tokenization, text lowercasing, stopword removal, stemming, lemmatization, bag of words (BoW), term frequency-inverse document frequency (TF-IDF), word embeddings, and sequence padding. These techniques help capture the relevant information in the text and create informative features that can be used for various NLP tasks, such as sentiment analysis, text classification, named entity recognition, and machine translation. |

1. Explain lemmatization and provide an example.

|  |
| --- |
| Lemmatization is a text normalization technique used in natural language processing (NLP) to convert words into their base or root form, known as the lemma. The lemma represents the canonical form of a word, and lemmatization helps reduce inflected words to their base form, making it easier to analyze text by considering different inflections as the same word.  Example:  Consider the words "running," "ran," and "runs." Applying lemmatization to these words would result in the following lemmatized forms:  "running" → "run"  "ran" → "run"  "runs" → "run"  In this example, lemmatization reduces different inflected forms of the word "run" to its base form "run," making it easier to process and analyze text data. |

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

|  |
| --- |
| TF-IDF (Frequency-Inverse Document Frequency) is a numerical representation used in natural language processing (NLP) and information retrieval to measure the importance of a word in a document relative to a collection of documents. It is commonly used to convert text data into a numerical format that can be used for machine learning algorithms.  TF-IDF = TF \* IDF where  Term Frequency (TF): The number of times a word appears in a document divided by the total number of words in that document.  TF = (Number of occurrences of word in the document) / (Total number of words in the document)  Inverse Document Frequency (IDF): The logarithm of the total number of documents divided by the number of documents containing the word.  IDF = log((Total number of documents) / (Number of documents containing the word)) |

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

|  |
| --- |
| Vectorizers and word embeddings are techniques used in NLP to convert text data into numerical representations. Vectorizers like CountVectorizer and TF-IDF Vectorizer create fixed-size representations based on word frequencies, treating words in isolation. They have high dimensionality and do not capture semantic meaning or context. On the other hand, word embeddings like Word2Vec and GloVe are dense, low-dimensional representations learned from data using neural networks. They capture semantic relationships and context, mapping words into a continuous vector space. Word embeddings are preferred for NLP tasks as they handle large vocabularies, preserve word meanings, and improve performance in tasks like sentiment analysis and machine translation. |

1. What is the difference between a Neural Network and the other Supervised Learning models that you have implemented? When should neural networks be used?

|  |
| --- |
| * Neural Networks are machine learning models inspired by the human brain, consisting of interconnected nodes organized into layers with adjustable weights that learn from data. * Other Supervised Learning models (e.g., Decision Trees, Random Forests, Linear Regression) have different algorithms and structures compared to Neural Networks. * Neural Networks are used in tasks involving complex data and large amounts of training data, such as image recognition, speech recognition, NLP, predictive modeling, and reinforcement learning. * Neural Networks are suitable for handling complex patterns in data, but they require substantial computational resources and can overfit with limited training data. * For smaller datasets and simpler tasks, traditional supervised learning models may be more appropriate due to faster training and inference times. |

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