Problem Statement

Welcome to the project on **Neural Machine Translation (NMT) model building** as part of your course on **Natural Language Processing**. So far, you have studied classical NLP, such as lexical, syntactic, and semantic processing. Later, this course introduced you to the basics of attention models. NMT is a very powerful method in computer vision, where most recent applications have been around Neural Machine Translation (NMT). The traditional methods that rely on heavy and massive data use labeling techniques to map each word with complex functions.

Using attention mechanisms in NMT is proven to be a much simpler approach.

Let us first go through the background of this assignment. We will then understand the tasks required as part of your submission.

Background:

Due to language barriers, a US-based life insurance company is facing a challenge in communicating with the Spanish-speaking community in Mexico. The locals there need help understanding English, and the company can only provide some translators. To overcome this challenge and provide coverage to the locals, the company needs a machine translation model that can accurately translate the application request letter from Spanish to English.

Problem Statement:

Your task is to build an attention-based sequence-to-sequence model that can effectively understand the context of Spanish sentences and translate them into clear and coherent English sentences. The company aims to use this model to ensure seamless communication and provide coverage to the Spanish-speaking

community in Mexico. In many countries where English is not a very common language for speaking, local traders often need help doing business with merchants from the US or other English-speaking countries.

NOTE: In such a case, a language translation model does wonders, and local traders need not worry about knowing too many languages. In this assignment, you will simply be building a **Spanish-to-English NMT model**. Although, in the real world, you could also build an NMT model for other local languages.

Why is this assignment important?

Building a Spanish-to-English translation model will help you understand the core basics of an attention-based-sequence-to-sequence model. This is slightly similar to the Hindi - English NMT Model you studied in the main content of the modules; however, in this assignment, you will understand some essential data-cleaning tasks specific to the Spanish language. It will help you deal with or understand challenges to languages like Spanish with 'special characters.'

Notebook:

Besides, an important note to remember while doing this assignment is that you are given a **'Stub Code File,'** which means a pre-written code is already shared with you. You have to fill in the place where it is mentioned, **'Write Code Here,'** following the instructions mentioned in that cell or for the task in that particular section.

Remember the following points while solving the assignment:

- All the tasks and details around performing them are detailed in this stub code file.
- You need to go through the instructions mentioned as 'comments' in the stub code file and add a code where it is mentioned as **'**Insert/Write Code Here —'**.
- To understand the complete flow of that particular section and its sub-tasks, scroll through the table of contents section in the left tab of this stub code file.

- In case of any existing part of code or something already written, you may or may not modify it as long as you serve the purpose of the task required in that section.

Evaluation Rubric

Criteria	Meet Expectations	Does Not Meet Expectations
Task 1: Loading and visualizing the data (10%)	The submission file contains the complete code required for Task 1 on data cleaning and visualization and also runs well on execution.	The submission file **DOES NOT** contain the complete code required for Task 1 on data cleaning and visualization and also runs well on execution.
Task 2: Process the data (15%)	The submission file contains the complete code required for Task 2 on cleaning the data using text standardization, text tokenization, creating a train test split, and also creating a tensorflow/tf.dataset.	The submission file DOES NOT contain the complete code required for Task 2 on cleaning the data using text standardization, text tokenization, creating a train test split, and also creating a tensorflow/tf.dataset.
Task 3: Build the NMT model (40%)	The code for Encoder - Bahanadau and Decoder model is completed as per the instructions and runs well on execution.	The Encoder - Bahanadau and Decoder model code is NOT completed or does not run well on execution. Or the code is incomplete in either of these sections.

Task 4: Train the NMT model (10%)	All the preliminary steps in the training part of the NMT model are completed as per the instructions and run well on execution.	All the preliminary steps in the training part of the NMT model are completed as per the instructions and do not run well on execution.
Code readability and conciseness (5%)	The code is concise and syntactically correct. Wherever appropriate, built-in functions and standard libraries are used instead of writing extended code. Custom functions are used to perform repetitive tasks. The code is readable with appropriately named variables and written detailed.	Long, complex code is used instead of shorter built-in functions. Custom functions are not used to perform repetitive tasks, resulting in the same code being repeated multiple times. Code readability is poor because of vaguely named variables or a lack of written comments where necessary.

Submissions required

As part of the submission, you have to submit:

1. **One well-commented Python notebook:** One IPYNB notebook file with completed code around the following tasks with the desired code where instructions are mentioned to 'Insert/Write Code Here':
    1. Load the Data
    2. Process the Data
    3. Build the NMT Model
    4. Train the NMT Model
    5. Evaluate the NMT Model