Text generation serves various purposes like machine translation, chatbots, virtual assistants, and AIGC. To understand the text generation process with a language model, consider it as an iterative approach. Initially, we predict the first word from the input sequence and subsequently use that predicted word in the input to generate the second word. Repeat the process until *done*.

If we recall, we focused on *letter*generationin class. Now, **using the review data in HW1**, let's build upon the in-class examples to create a simulated review using a **GRU or LSTM**model. For simplicity's sake, let's concentrate on coding the training process without delving into hyperparameter tuning or model evaluations. We'll generate both a negative review (combining 1-star, 2-star, and 3-star comments) and a positive review (utilizing 4-star and 5-star comments). Essentially, our goal is to generate tokens instead of individual letters.

**Problem 2: BLEU score.**

Create a function that implements the BLEU score. See the sample output below for an example. For more details, see bottom of the [seq2seq](https://gastate.view.usg.edu/d2l/le/content/2885881/viewContent/58526503/View) example as shown in class.

A screen shot of a computer

Description automatically generated

**Problem 3: Training Data Generator (If the HW feels too hard, this one is ok to skip)**

In real-world Natural Language Processing (NLP) applications in Deep Learning, Python [iterators](https://realpython.com/python-iterators-iterables/) are frequently used to create training data and formalize [model training pipelines](https://www.geeksforgeeks.org/natural-language-processing-nlp-pipeline/). I'd like the class to explore a basic example to grasp this concept.

As a quick recap, when generating training data for an RNN model, we typically sample data with a specified sequence length (referred to as 'num\_steps' in [RNN sample code](https://gastate.view.usg.edu/d2l/le/content/2885881/viewContent/58369575/View)).

For detailed instructions, please refer to the section below. Keep in mind that you only need to provide the completed function along with a sample result for submission.

In the attached ipynb, The function in **Cell 8** `seq\_data\_iter\_sequential` is the sequential iteration. As shown in **Cell 9** below, the first X is `[0, 1, 2, 3, 4]` while the second X is `[5, 6, 7, 8, 9]`.

**A screen shot of a computer code

Description automatically generated**

I have hidden the function body in **Cell 6**, please complete it to generate results similar to those of **Cell 7** as shown below (where `[8, 9, 10, 11, 12]` is followed by `[28, 29, 30, 31, 32]`).

A screen shot of a computer

Description automatically generated

Due on Oct 21, 2023 11:59 PM

Attachments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | [lm.ipynb](https://gastate.view.usg.edu/d2l/common/viewFile.d2lfile/Database/MjA0NDE5Mjg1/lm.ipynb?ou=2885881) (125.24 KB) |  |  |  | |
| Download All Files |