CSE 4392 Special Topic: Natural Language Processing

Homework 2 - Spring 2024

Due Date: Feb 3, 2024, 11:59 p.m. Central Time

As we delve into the fascinating world of Natural Language Processing (NLP), one of the fundamental concepts we encounter is the N-gram language model. This model is a crucial tool in understanding how sequences of words are structured in a language, and it forms the basis for many applications in NLP, such as text prediction, machine translation, and speech recognition.

In this assignment, you will work with some small corpora that are deliberately designed to be simple yet challenging. You'll train your n-gram model on this corpus and then test its effectiveness on a new sentence. This process will involve handling scenarios where your model encounters unseen data, thereby introducing you to the concept of smoothing - a critical aspect in the real-world application of language models.

Problem 1 - 40%

The corpus provided for training the bi-gram model consists of the following four sentences:

- Cats chase after playful mice
- Birds sing at morning dawn
- Fish swim in the pond
- Dogs bark at passing cars

The corpus provided for testing the bi-gram model consists of the following two sentences:

- Cats sing and dogs swim
- Playful dogs chase birds at dawn

Question 1 - 15%

Please write down the raw counts in the training data for the following bigrams found in **Test Sentence 1**

	cats	sing	and	dog	swim
cats					
sing					
and					
dogs					
swim					

Question 2 - 15%

Upon reviewing the table, you'll notice several zeros indicating absent bigrams from our training data. To overcome this, please apply Add-one Laplace Smoothing to filling the following table the reconstituted counts. This adjustment will help us better understand our model's performance.

	cats	sing	and	dog	swim
cats					
sing					
and					
dogs					
swim					

Question 3 - 10%

Please calculate the PPL of the **Whole Test Set** after Laplace Smoothing .

Problem 2 - 60%

The corpus provided for training the trigram model consists of the following sentences:

- the cat watched children play in the park.
- their laughter echoed near the fragrant garden.
- the breeze spread the garden's scent into the city.
- it wafted past the cafe, famous for apple pie.
- the cafe's aroma reminded people of the nearby library.
- the library held tales of the ancient clock tower.
- the tower tolled, echoing in the quiet morning streets.

- these streets, bustling by day, were peaceful at dawn.
- at night, they lay under a star-filled sky.
- the moonlight shone on the lake where a fisherman waited.

The corpus provided for testing the trigram model consists of the following three sentences:

- the sunset painted the city sky with colors.
- an old train's whistle echoed past the lake.
- soft music played in the cozy cafe.

Note that punctuation and 's are treated as separate tokens. In the last problem, we discussed the concept of bigrams. Now, let's shift our focus to trigrams and delve into their practical applications in programming.

Question 1 - 40%

Create a trigram model using the given corpus. Calculate the probabilities of each trigram in the test set based on your model. If a trigram from the test sentence does not exist in your training data, handle this scenario using **Linear Interpolation** smoothing technique. You can set the λ_1 as 0.5, λ_2 as 0.4, λ_3 as 0.1 for initialization. (You don't need to fine-tune the parameter.) Please take a screenshot of the experimental results and attach it in the PDF.

Question 2 - 20%

Calculate the perplexity of the **whole test set**. Please take a screenshot of the experimental results and attach it in the PDF.

Please finish this homework in Python3.
Please add a readme.md to introduce how to run the code.
Please upload a zip file contains the pdf and .py file.