과제 #2

Due 9/29

201904479 임서연 컴퓨터전자시스템

1. 0~9 범위의 10개의 숫자를 인식하는 시스템의 경우의 perplexity를 유도하라.

$$PP(w) = P(w_1, w_2, w_3 \cdots w_N)^{-\frac{1}{N}} = \left(\frac{1}{10}^N\right)^{-\frac{1}{N}} = \frac{1}{10}^{-1} = 10$$

2. 다음 문장에 대해 unigram, 2-gram, 3-gram 방식에 의해 perplexity를 계산하는 수식을 유도하라.
   문장: There is a major problem with the maximum likelihood estimation process.

unigram

$$PP(w) = \sqrt[N]{\prod_{j=1}^{N} \frac{1}{P(w_i | w_1 \cdots w_{i-1})}}$$

$P(w_i | w_1 \cdots w_{j-1})$

= P(there)·P(is)·P(a)·P(major) P(problem) ·····

2-gram

$$PP(w) = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i | w_{i-1})}}$$

$P(w_i | w_{i-1})$

= P(is|there)·P(a|is)·P(major|a)·
P(problem|major)·P(with|problem) ·····

3-gram

$$PP(w) = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P(w_i | w_{i-1}, w_{i-2})}}$$

$P(w_i | w_{i-1}, w_{i-2})$

= P(process| likelihood estimation)·
P(estimation|maximum likeilhood)·
P(likelihood|the maximum)·····

3. 일정한 text data를 이용하여 n-gram 시스템을 훈련시켰을 때의 perplexity를 고려한다.

a. n이 증가할수록 perplexity가 감소하는 이유를 설명하라.

-perplexity is related inversely to the likelihood of the test sequence according to the model)

b. Perplexity를 계산할 때 사용 단어군을 고정시켜야 하는 이유는 무엇인가?

-Any kind of knowledge of the test set can cause the perplexity to be artificially low. The perplexity of two language models is only comparable if they use identical vocabularies

4. 다음 프로그램은 nltk 패키지를 이용하여 n-gram을 구하는 프로그램이다.

```
from nltk import ngrams

sentence = 'this is a foo bar sentences and i want to ngramize it'

n = 6
sixgrams = ngrams(sentence.split(), n)

for grams in sixgrams:
  print(grams)
```

또한 brown corpus 데이터를 읽어오는 프로그램은 다음과 같다.

```
brown = nltk.corpus.brown
corpus = [word.lower() for word in brown.words()]
```

brown corpus 에서 모든 3-gram을 구한 다음, 어떤 문장의 perplexity를 계산하는 프로그램은 어떻게 구현해야 하는지 설명하라. 단, 최종 단계로 구현하지 않아도 된다.

n-gram은 다음에 나올 단어의 예측은 오직 n-1개의 단어에 의존합니다. 'this is a foo bar sentences and i want to ngramize it' 문장에서 다음에 나올 단어를 예측하고 싶을때, n=3라고한 3-gram일 이용한 언어모델을 사용한다. 이경우 ngramize 다음에 올 단어를 예측하는 것은 n-1에 해당되는 앞의 2개의 단어만 고려합니다.

$$P(w \mid \text{to ngramize}) = \frac{\text{Count(to ngramize w)}}{\text{Count (to ngramize)}}$$

만약 가지고 있는 corpus에서 want to 이 100번 등장했다고 한다. 그리고 to ngramize it이 50번 등장했으며, to ngramize them이 20번 등장했다고 가정을 한다. 그렇게 되면 to ngramize 다음에 it이 등장할 확률은 50프로 이며, them이 등장할 확률은 20프로 이다. 확률적인 선택에 따라 it이 더 맞다고 판단하게 된다.

$$P(\text{it} \mid \text{to ngramize}) = 0.50$$

$$P(\text{them} \mid \text{to ngramize}) = 0.20.$$