

گزارش بخش perplexity

	1gram	2gram	3gram
Label1_Train	2203.60	1187.24	1071.25
Label2_Train	3169.87	2284.98	1837.91
Label1_Test	1972.29	1512.50	1253.49
Label2_Test	2075.08	1737.38	1448.35

همانطور که مشاهده میشود چون perplexity برعکس احتمال است پس بنابراین مدلی که perplexity کمتری دارد مدل بهتری (با احتمالات بالاتری) است. بنابراین طبق مشاهدات ما هرچه درجه ngram زیاد شود مدل بهتر عمل کرده و perplexity پایین می آید.

$$PP(1\text{gram}) > PP(2\text{gram}) > PP(3\text{gram})$$

در باره داده های train و test نیز نتایج من بسیار نزدیک به هم شد اما منطقا داده های train نتایج بهتری باید داشته باشند چون روی آنها مدل را ساخته ایم البته این تفاوت به علت تفاوت تعداد unkها در ورودی است اگر آنرا متعادل سازی کنم جواب منطقی تری خواهم گرفت .

* تصویر نتایج در صفحه ی بعد *

```
zahra@zahra-X556UR:~/Desktop/6th Semester/NLP/NLP97982/p2/Perplexity/src$ python3 main.py
What is your language model type?
1)unigram
2)bigram
3)trigram
4)test for my own data automatically
4

*****LABEL1 - TRAIN*****

Perplexity of unigram - train data- label1
PP = 2203.608935087357
Perplexity of bigram - train data- label1
PP = 1187.2456969334955
Perplexity of trigram - train data- label1
PP = 1071.253134639781

*****LABEL2 - TRAIN*****

Perplexity of unigram - train data- label2
PP = 3169.876948629802
Perplexity of unigram - train data- label2
PP = 2284.9843345159934
Perplexity of unigram - train data- label2
PP = 1837.919972154055

*****LABEL1 - TEST*****

Perplexity of unigram - test data- label1
PP = 1972.296162080615
Perplexity of bigram - test data- label1
PP = 1512.5003702217045
Perplexity of trigram - test data- label1
PP = 1253.492594719969

*****LABEL2 - TEST*****

Perplexity of unigram - test data- label2
PP = 2075.088746346847
Perplexity of bigram - test data- label2
PP = 1737.385878898383
Perplexity of trigram - test data- label2
PP = 1448.353011463487
zahra@zahra-X556UR:~/Desktop/6th Semester/NLP/NLP97982/p2/Perplexity/src$
```