Assignment3: Language Modeling

CSE256: Statistical NLP: Spring 2022
University of California, San Diego

Zhaoxing, Lyu
A59011303
zhlyu@ucsd.edu

1 Introduction

This project is focusing on discovering language model. Unigram language model will be analyzed by performing on provided datasets and another language model which considers the context will be constructed and described. Perplexity will be an important evaluation towards these language models and sampled sentences will be generated to show what new language model represents for each domain.

2 Unigram Language Model Analysis

In this part, unigram language model which completely ignores context will be analyzed on In-Domain Text and Out-of-Domain Text with different evaluations.

2.1 Analysis on In-Domain Text

The trends of the perplexity with the increase of training data are tested in this section and the result of three different corpus are illustrated in 1, 2 and 3.

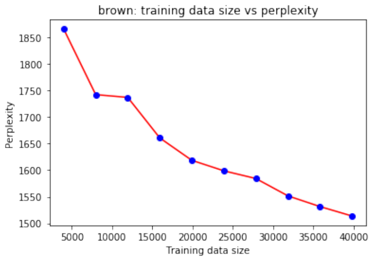

Figure 1: Empirical Analysis of the Unigram LM on Brown Corpus

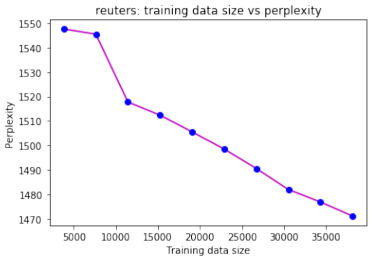

Figure 2: Empirical Analysis of the Unigram LM on Reuters Corpus

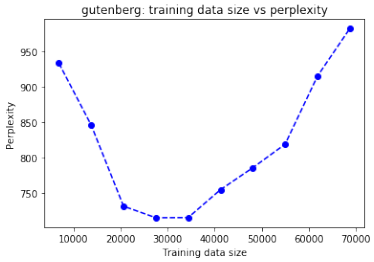

Figure 3: Empirical Analysis of the Unigram LM on Gutenberg Corpus

There are obvious downtrends when implementing unigram language model on brown corpus and reuters corpus, indicating that with the increase of the size of the training dataset, the perplexity will decrease as well. However, when implementing the language model on gutenberg corpus, it can be illustrated that when the size of training data was 3000 to 4000, model had the best outcome on perplexity. After this point, the perplexity increased with the increase of the training data size.

From the describing result, it can be found that on most corpus, the increasing of the training data size will decrease the perplexity of the language model.

2.2 Analysis on Out-of-Domain Text

In this part, unigram models will be implemented on a domain different from the one it was trained on and the result is recorded in tables.

train	brown	reuters	gutenberg
brown	1513.8	6780.82	1758.06
reuters	3806.39	1471.21	4882.8
gutenberg	2616.57	12420.1	982.572

test	brown	reuters	gutenberg
brown	1604.2	6736.6	1762.01
reuters	3865.16	1500.69	4887.47
gutenberg	2626.05	12392.5	1005.79

dev	brown	reuters	gutenberg
brown	1589.39	6675.63	1739.41
reuters	3808.87	1479.09	4833.88
gutenberg	2604.28	12256.3	991.5

Table 1: Perplexity of all Three of the Models on all Three Domains of unigram model

backoff	Perplexity		
	train	dev	test
0.000001	351320.35	483749.45	481541.86
0.0001	8524.70	8155.26	8134.28
0.001	206.84	137.48	137.40

* the alpha in smoothing is 1 and the experiment is on brown corpus

Table 2: Perplexity when the Changing Backoff on the Trigram Model

From the result in 2.2, it can be found that the unigram model perform better in in-domain text than out-domain text in three datasets. Additionally, the model have similar performances on brown and gutenberg, indicating that these two corpus have similar grammars and trained models have related performance. The model trained in reuters performed terribly in brown and gutenberg datasets, which further shows the difference between reuters corpus and the other two.

3 Implement a Context-aware Language Model

In this section a trigram language model will be implemented and explained with performing the model on three different corpus.

3.1 Implementation

3.1.1 Trigram language model

The trigram language model built in this project counting the bigram and trigram in the datasets and calculating the probability by using The Maximum Likelihood Estimate (MLE) in1.

$$
\begin{aligned}
q\left(w_i \mid w_{i-2}, w_{i-1}\right) &= \frac{\operatorname{count}\left(w_{i-2}, w_{i-1}, w_i\right)}{\operatorname{count}\left(w_{i-2}, w_{i-1}\right)} \\
q(\text{ laughs—the, dog }) &= \frac{\text{count(the , dog, laughs)}}{\text{count(the , dog)}}
\end{aligned}
\tag{1}
$$

When inputting sentence having only a word, ['*','*'] will be added to make it become a trigram or bigram. Additionally, in this model, backoff was implemented as the unigram model: when meeting the unseen bigram or trigram, backoff will be used as the probability. The backoff value was modified to find the best outcome and the experiment result is showed in Table 3.1.1.

From the result, we can find out that when backoff is bigger, the model has better outcome and when backoff is 0.001, the perplexity is as good as 206.84 in train set on brown corpus.

3.1.2 Smoothing

The term smoothing refers to the adjustment of the maximum likelihood estimator of a language model so that it will be more accurate. In this project, Laplace smoothing was added to the model 2 to improve the result of the model.

$$
q\left(w_i \mid w_{i-2}, w_{i-1}\right) = \frac{\operatorname{count}\left(w_{i-2}, w_{i-1}, w_i\right) + \alpha}{\operatorname{count}\left(w_{i-2}, w_{i-1}\right) + |\mathcal{V}|}
\tag{2}
$$

alpha	Perplexity		
	train	dev	test
0.000001	222.99	141.18	141.02
0.0001	223.22	141.23	141.07
1	206.84	137.48	137.40

[*] the backoff is 0.001 and the experiment is on brown corpus

Table 3: Perplexity when the Changing Alpha on the Trigram Model

During the experiment, α was adjusted to have a better result. From the test result in 3.1.2, we can find that when α is 1, the model has the best outcome, but the change of the α does not bring too much difference.

3.2 Evaluation

Trigram constructed in this project was implemented with three-domain text and the results from train, dev and test were collected and illustrated in Table 3.2.3.

3.2.1 Comparison with the unigram model

By comparing results in Table 2.2 and Table 3.2.3, it can be found that the perplexity obtained from the built trigram language model is much better than the result from unigram in all domains. This indicates that with considering the context, the model has a better outcome.

3.2.2 Examples of sampled sentences

As for the examples of sampled sentences, I used 'i', 'am' as the prefix input of sample sentence function in generator.py and the result is:

1. brown
   (a) i am wohaws glamour currencies manmade resealed sopranos undergo mesenteric print Hang replaces wind psychical Blauberman strips Lander obtain Russians 642 Styles Annie
   (b) i am small growers wheedled response tryin infidel neither lightning modulations speeded chuck fern rocks accounting isolate cigarette lasting Deaf translating loopholes weapons
2. reuters
   (a) i am legally Judicial WILKINSON 285 aircraft suprised predominant ours retracement charging 837 SHIPBUILDER fastest industrialsed Ordonez PITTSBURGH UM PROCESSING posting quartger PARANAVAI
   (b) i am judging resurge VAL indentify stops acquisition idle absorbed Intex delay committees MCDP Uclaf PENSACOLA Ankara regional acrimony Amax CHELSEA unsettle milk
3. gutenberg
   (a) i am Serious Honest unemployables Edmund delighful definably kaleidoscope Zithri STRONG Poverties prejudice Shop impregns waketh encountering punished mimic Nathanael Authoriz Enlightener Confused
   (b) i am vale aptitude dismays reiteration teasing SICK evenings flagging exercises champing Departed liquours piped unflinching woodpile greasy attributed Forests lanyard foregoing UP

This is the samples showed by my model in different corpus.

train	brown	reuters	gutenberg
brown	206.85	111.247	123.154
reuters	117.307	166.314	112.346
gutenberg	130.91	106.727	202.135

test	brown	reuters	gutenberg
brown	137.406	111.007	123.099
reuters	117.837	132.797	112.478
gutenberg	131.168	107.073	158.756

dev	brown	reuters	gutenberg
brown	137.485	111.02	123.303
reuters	117.239	133.173	112.329
gutenberg	130.061	106.607	159.13

Table 4: Perplexity of all Three of the Models on all Three Domains of trigram model

3.2.3 Out-of-Domain Text Analysis (Empirical)

The interesting part of my building model is that out-of-domain usually have better outcome than its own dataset as being illustrating in the table 3.2.3. The reason towards it is that the building model has good generalization ability, which is also illustrated as the dev and test have less perplexity.

3.2.4 Out-of-Domain Text Analysis (Qualitative)

From the results of the experiments, it can be found that the trigram model built in this project has good generalization ability as it performed better in out domain corpus and dev, test datasets, shown in table 3.2.3. It also improved the outcome compared to the unigram by comparing table 2.2 and table 3.2.3. The reason towards this one is that the trigram model considering the neighboring context instead of considering nothing. From the results of two models, it can be found that brown and gutenberg has related grammar as they got similar perplexity.

4 Conclusion

This report implemented language model including unigram and trigram language models. During implementing the unigram, if can be found that with the increase of the training data, the perplexity become lower. By comparing the results, trigram which considering neighboring context performed better than unigram. Backoff value and smoothing value were adjusted to get good output in trigram. With comparing the results in different domains and different datasets, the trigram has good generalization ability and the brown corpus shares similar grammar with gutenberg.