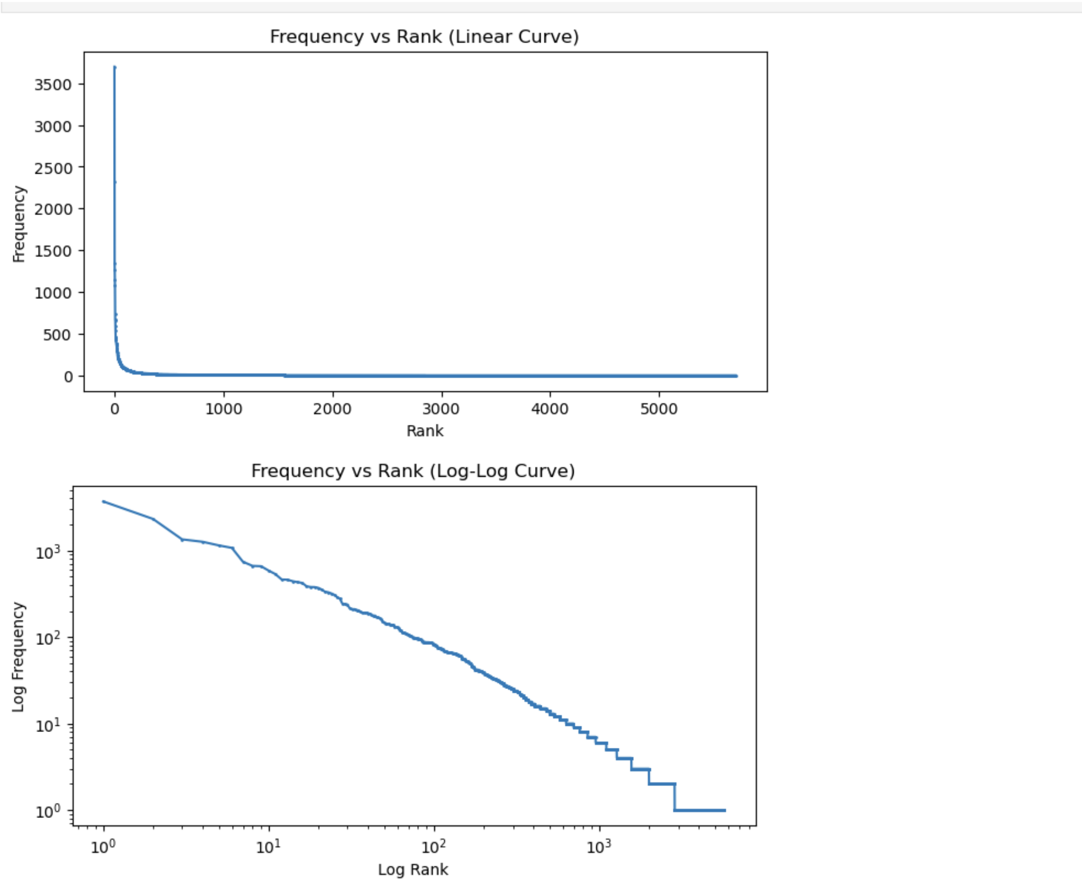
Assignment 1 Kotomi Fukushima

Task 1

The graph:

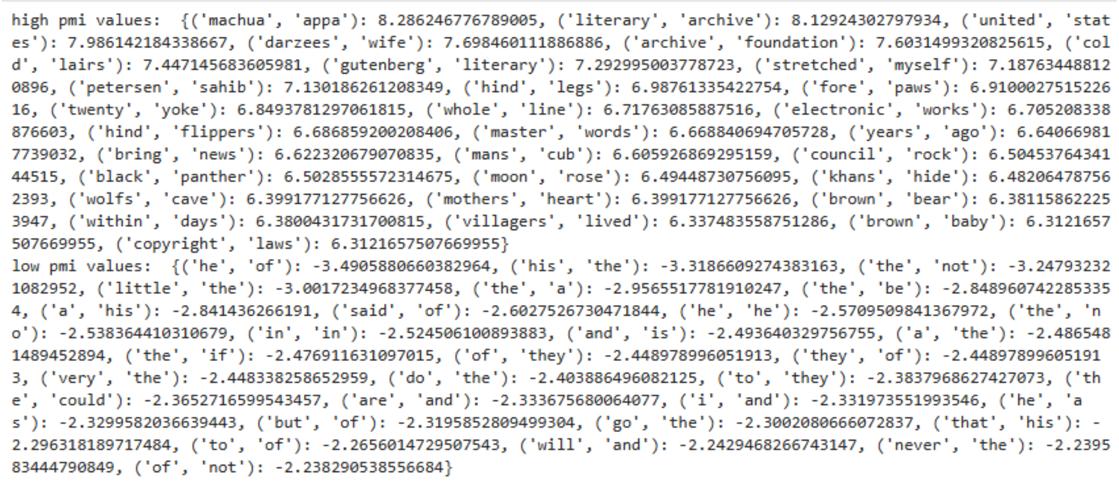


Findings:

- The log-log plot looks like a linear line, which supports Zipf's Law: the frequency of a word is inversely proportional to its rank.
- Many words have very low frequency.
- The most frequent word appears much more than low-frequency words.

Task 2

High and low PMI values:



```
high pmi values:  {('machua', 'appa'): 8.286246776789005, ('literary', 'archive'): 8.12924302797934, ('united', 'stat
es'): 7.986142184338667, ('darzees', 'wife'): 7.698460111886886, ('archive', 'foundation'): 7.6031499320825615, ('col
d', 'lairs'): 7.447145683605981, ('gutenberg', 'literary'): 7.292995003778723, ('stretched', 'myself'): 7.18763448812
0896, ('petersen', 'sahib'): 7.130186261208349, ('hind', 'legs'): 6.98761335422754, ('fore', 'paws'): 6.9100027515226
16, ('twenty', 'yoke'): 6.8493781297061815, ('whole', 'line'): 6.71763085887516, ('electronic', 'works'): 6.705208338
876603, ('hind', 'flippers'): 6.686859200208406, ('master', 'words'): 6.668840694705728, ('years', 'ago'): 6.64066981
7739032, ('bring', 'news'): 6.622320679070835, ('mans', 'cub'): 6.605926869295159, ('council', 'rock'): 6.50453764341
44515, ('black', 'panther'): 6.5028555572314675, ('moon', 'rose'): 6.49448730756095, ('khans', 'hide'): 6.48206478756
2393, ('wolfs', 'cave'): 6.399177127756626, ('mothers', 'heart'): 6.399177127756626, ('brown', 'bear'): 6.38115862225
3947, ('within', 'days'): 6.3800431731700815, ('villagers', 'lived'): 6.337483558751286, ('brown', 'baby'): 6.3121657
507669955, ('copyright', 'laws'): 6.3121657507669955}
low pmi values:  {('he', 'of'): -3.4905880660382964, ('his', 'the'): -3.3186609274383163, ('the', 'not'): -3.24793232
1082952, ('little', 'the'): -3.0017234968377458, ('the', 'a'): -2.9565517781910247, ('the', 'be'): -2.848960742285335
4, ('a', 'his'): -2.841436266191, ('said', 'of'): -2.6027526730471844, ('he', 'he'): -2.5709509841367972, ('the', 'n
o'): -2.538364410310679, ('in', 'in'): -2.524506100893883, ('and', 'is'): -2.493640329756755, ('a', 'the'): -2.486548
1489452894, ('the', 'if'): -2.476911631097015, ('of', 'they'): -2.448978996051913, ('they', 'of'): -2.44897899605191
3, ('very', 'the'): -2.448338258652959, ('do', 'the'): -2.403886496082125, ('to', 'they'): -2.3837968627427073, ('th
e', 'could'): -2.3652716599543457, ('are', 'and'): -2.333675680064077, ('i', 'and'): -2.331973551993546, ('he', 'a
s'): -2.329958203663944З, ('but', 'of'): -2.3195852809499304, ('go', 'the'): -2.3002080666072837, ('that', 'his'): -
2.296318189717484, ('to', 'of'): -2.2656014729507543, ('will', 'and'): -2.2429468266743147, ('never', 'the'): -2.2395
83444790849, ('of', 'not'): -2.238290538556684}
```

Observations:

- High PMI pairs seem unique to this corpus. These are word combinations that rarely appear outside this particular context, leading to a significantly higher PMI value. This validates PMI's strength in capturing strong, specific associations rather than just frequency.
- Low PMI pairs seem random combinations of common words. They're usually pretty common words that just happen to be next to each other sometimes, but not in any meaningful or consistent way. It's like they appear together just by chance, not because they're part of a fixed phrase or idea.

Assumption:

High PMI pairs show that certain words are strongly tied together, and low PMI values highlight that frequent words are often incompatible with one another in direct sequence. Thus, unigram model is limited in capturing patterns, so n-gram model is better suited for language modeling.

Task 3

Data preprocessing:

- Convert all text to lowercase
- Remove punctuation and special characters
- Tokenize the cleaned text into individual words

Method:

- The next word is predicted based on the most likely word to follow the current word.
- Evaluation is done by checking if the predicted word matches the actual next word

Experiment design:

- The model is trained on wiki.train.raw, validated on wiki.valid.raw, and tested on wiki.test.raw from the Wikitext-2 dataset.
- Evaluation is performed on 1000 randomly selected word positions in the validation and test sets.

Hyper-parameter:

- max_word = 1000

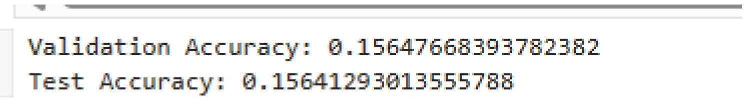
  This hyper-parameter reduces computation time.

Evaluation metric:

- Accuracy

  Calculate the percentage of correctly predicted next words among 1000 samples.

Accuracy:



```
Validation Accuracy: 0.15647668393782382
Test Accuracy: 0.15641293013555788
```

Findings:

- Both validation and test accuracies are around 15.64%, indicating this model can capture some patterns, but not good.

Drawbacks:

- This model cannot predict anything if a word was not in the training data.

Improvements:

- Use smoothing techniques
- Use more training data
- Use higher-grams