General Analisis

Dari hasil ujicoba, model bigram memiliki perplexity yang lebih tinggi dibandingkan unigram.

```
UNIGRAM
n = 1, Perplexity: 9.5864

BIGRAM
n = 1, Perplexity: 10.9771
```

Ini berbanding terbalik dengan ekspektasi, di mana kita menginginkan perplexity untuk naik ketika n di n-gram naik. Adapula ujicoba menggunakan dataset yang lebih kecil yang menghasilkan perplexity unigram yang lebih tinggi dibandingkan bigram.

```
UNIGRAM
n = 1, Perplexity: 2.7330

BIGRAM
n = 1, Perplexity: 1.8751
```

Dataset yang lebih kecil mengandung lebih banyak kalimat yang berulang. Hal ini kemungkinan mempengaruhi model bigram yang memperhitungkan given word sebelum prediksi.

Beberapa hal yang dapat ditarik dari ini adalah:

- Ukuran data sangat mempengaruhi perplexity dari model baik dalam hal training maupun efektivitas model.
- Perulangan kata di dalam dokumen sangat berpengaruh ketika menggunakan n-gram model. Bila dokumen memiliki sedikit kata yang mengulang, sebaiknya tidak menggunakan n-gram
- Ukuran n dalam n-gram, meskipun membuat prediksi yang lebih bagus, juga membuat model dengan perplexity yang tinggi bila dataset tidak sesuai.

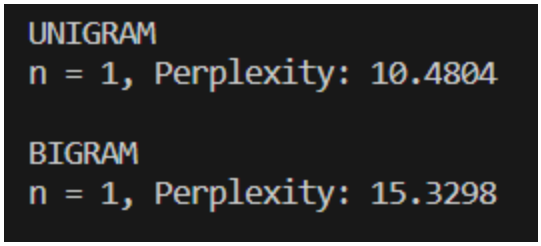
1. **Case dan No-Case**

No-Case

```
UNIGRAM
n = 1, Perplexity: 9.5864

BIGRAM
n = 1, Perplexity: 10.9771
```

Case



Untuk Bigram dan Unigram *no case* bersifat lebih bagus. Ini dapat dilihat dari perplexitynya.
Karena n-gram bersifat case sensitif, sudah seharusnya data dengan *no case* memiliki perplexity yang lebih besar.

2. **Tokenizer**
   Menghilangkan character tanda baca dan hanya menyimpan semua yang merupakan kata dan menyimpannya dalam array

3. **Uji Generasi**

```
UNIGRAM
karena yang dan yang dan yang dan yang dan yang dan
Prob -77.8544926832669
n = 1, Perplexity: 6.4879
Berkas yang dan yang dan yang dan yang dan yang dan
Prob -78.14035152344674
n = 1, Perplexity: 6.5117
oleh yang dan yang dan yang dan yang dan yang dan
Prob -76.14146514375744
n = 1, Perplexity: 6.3451
pergi yang dan yang dan yang dan yang dan yang dan
Prob -82.08788410355261
n = 1, Perplexity: 6.8407
dari yang dan yang dan yang dan yang dan yang dan
Prob -75.20932223100257
n = 1, Perplexity: 6.2674

BIGRAM
karena itu sendiri yaitu di dalam bahasa Indonesia yang lebih dari
Prob -99.93705398529836
n = 1, Perplexity: 8.3281
Berkas jpg jmpl px jmpl px jmpl px jmpl px jmpl
Prob -100.60252286962947
n = 1, Perplexity: 8.3835
oleh orang yang lebih dari bahasa Indonesia yang lebih dari bahasa
Prob -94.54114262098912
n = 1, Perplexity: 7.8784
pergi ke dalam bahasa Indonesia yang lebih dari bahasa Indonesia yang
Prob -96.59855218524088
n = 1, Perplexity: 8.0499
dari bahasa Indonesia yang lebih dari bahasa Indonesia yang lebih dari
Prob -93.50953028801395
n = 1, Perplexity: 7.7925
```

Teks dihasilkan dari mengambil word dengan probabilitas tertinggi. Karena mengambil dari probabilitas tertinggi, unigram menghasilkan generasi yang berulang-ulang dan tidak membentuk kalimat yang benar.

Bisa dilihat bahwa bigram menghasilkan kalimat yang lebih koheren. Ini karena bigram memperhitungkan satu kata yang datang sebelumnya. Meskipun begitu perplexity pada bigram tampak lebih tinggi. Ini mungkin disebabkan oleh besarnya vocab dan document yang menyebabkan kecilnya probabilitas untuk dua kata muncul.

4. **Probabilitas dari Kalimat**

```
UNIGRAM
Kalimat : saya sedang menunggu di peron 5 stasiun tersebut
Prob -124.34350141109913
n = 1, Perplexity: 13.8159

UNIGRAM
Kalimat : para pekerja terlihat lincah saat membersihkan lokomotif tersebut
Prob -123.38683179559904
n = 1, Perplexity: 13.7096

BIGRAM
Kalimat : pak ustad berceramah di atas mimbar masjid
Prob -112.08409527533189
n = 1, Perplexity: 12.4538

BIGRAM
Kalimat : para murid diajarkan budi pekerti di sekolah
Prob -113.87095827864333
n = 1, Perplexity: 12.6523
```

Probabilitas di atas adalah log2 dari angka probabilitas aslinya.

5. **Ukuran Vocab**
   Ukuran vocab adalah 8282 untuk Unigram dan 111493 untuk Bigram

6. **Analisis**

```
UNIGRAM
karena yang dan yang dan yang dan yang dan yang dan
Prob -77.8544926832669
n = 1, Perplexity: 6.4879
Berkas yang dan yang dan yang dan yang dan yang dan
Prob -78.14035152344674
n = 1, Perplexity: 6.5117
oleh yang dan yang dan yang dan yang dan yang dan
Prob -76.14146514375744
n = 1, Perplexity: 6.3451
pergi yang dan yang dan yang dan yang dan yang dan
Prob -82.08788410355261
n = 1, Perplexity: 6.8407
dari yang dan yang dan yang dan yang dan yang dan
Prob -75.20932223100257
n = 1, Perplexity: 6.2674

BIGRAM
karena itu sendiri yaitu di dalam bahasa Indonesia yang lebih dari
Prob -99.93705398529836
n = 1, Perplexity: 8.3281
Berkas jpg jmpl px jmpl px jmpl px jmpl px jmpl
Prob -100.60252286962947
n = 1, Perplexity: 8.3835
oleh orang yang lebih dari bahasa Indonesia yang lebih dari bahasa
Prob -94.54114262098912
n = 1, Perplexity: 7.8784
pergi ke dalam bahasa Indonesia yang lebih dari bahasa Indonesia yang
Prob -96.59855218524088
n = 1, Perplexity: 8.0499
dari bahasa Indonesia yang lebih dari bahasa Indonesia yang lebih dari
Prob -93.50953028801395
n = 1, Perplexity: 7.7925
```

Perplexity pada kalimat unigram lebih rendah dibandingkan bigram. Hal ini disebabkan oleh hasil generasi unigram yang berulang. Hasil berulang disebabkan oleh unigram meletakkan token dengan probabilitas tertinggi. Karena memiliki probabilitas tertinggi, hasil perplexity pun menjadi lebih bagus. Meskipun begitu dapat dilihat bahwa hasil generasi tidak koheren karena hanya mengambil kata dengan probabilitas terbesar.

Di sisi lain, perplexity dari bigram terlihat lebih tinggi. Akan tetapi, hasil generasinya jauh lebih koheren dibandingkan unigram. Ini disebabkan oleh model bigram yang memperhitungkan given word yang ada sebelum kata yang diprediksi. Meskipun

membuat kalimat yang lebih koheren, probabilitas yang kecil menyebabkan perplexity menjadi naik.