Standup Comedian Transcript Analysis and Generation

An NLP Project By Group 4

Group Members

Group 4

Afzal Mukhtar

PES2201800675

Hritika Rahul Mehta

PES2201800024

Farheen Zehra

PES2201800651

Problem Statement

What we want to solve

Generating a similar text for each standup artist, using Deep Learning Models

Approach

Using Deep Learning Models and testing different parameters to get the best generative model.

Overview of Research Methods

The research was done on various aspects of the generative models for Natural Language Processing.
Researching helped us understanding which model will be useful for our use-case.
The papers referred were:

- **Customizable text generation via conditional text generative adversarial network [Elsevier 2018]:** Jinyin Chen, Yangyang Wu, Chengyu Jia, Haibin Zheng, Guohan Huang
- **Smart Reply: Automated Response Suggestion for Email [ACM 2016]:** Anjuli Kannan, Tobias Kaufmann, Karol Kurach, Andrew Tomkins, László Lukács, Vivek Ramavajjala, Sujith Ravi, Balint Miklos, Marina Ganea, Greg Corrado, Peter Young
- **Diversity regularised auto encoders [ACM 2020]:** Hyeseon Ko, Junhyuk Lee, Jinhong Kim, Jongwuk Lee, Hyunjung Shim
- **Improving Variational Encoder-Decoders in Dialogue Generation (Reference Paper) [AAAI]:** Xiaoyu Shen, Hui Su, Shuzi Niu, Vera Demberg
- **Automatic Generation of Restaurant Reviews with LSTM-RNN [IEE/WIC/ACM 2016]:** Alberto Bartoli, Andrea De Lorenzo, Eric Medvet, Dennis Morello, Fabiano Tarlao

Data Collection Methods

Scraping and Cleaning

Scraping

The data for the transcripts for each artist was scraped from different websites using web scraping methods and APIs

Cleaning

The Data was cleaned to remove unnecessary words, emojis, emoticons, and expansion of contractions. They were further lemmatized ofr text normalization.

Data Preperation

The data was changed into a word to vector representation for word level text generation.

Project Synopsis

Model and Results

Model Used

An LSTM model was used with input sequence of 100 Words, where a one hot encoded Embedding was used. The softmax function was used as the output layer and a cross-entropy loss function was used for errors.

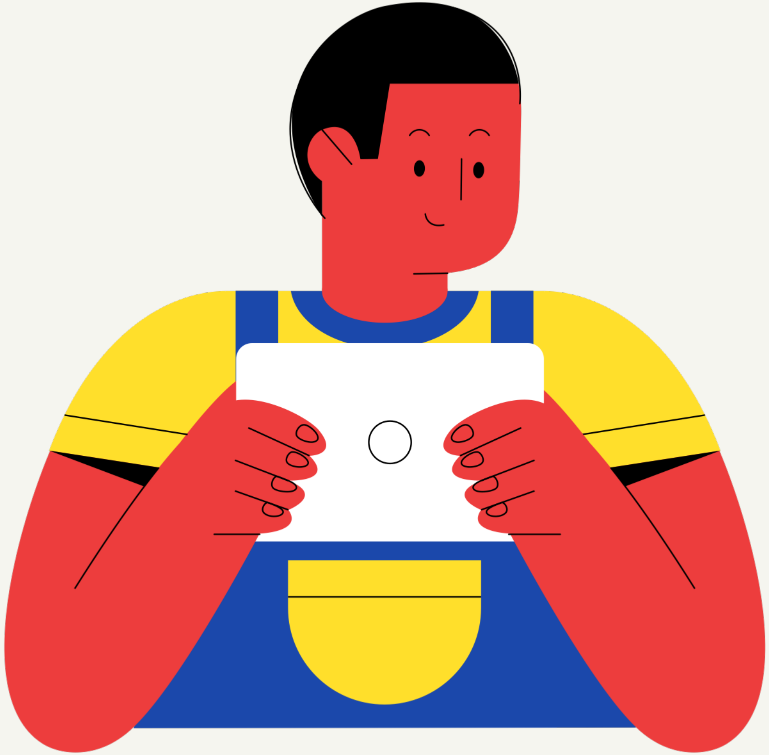
Training Data

The entire transcript of each artist was combined into one and passed to the model for training.

Results

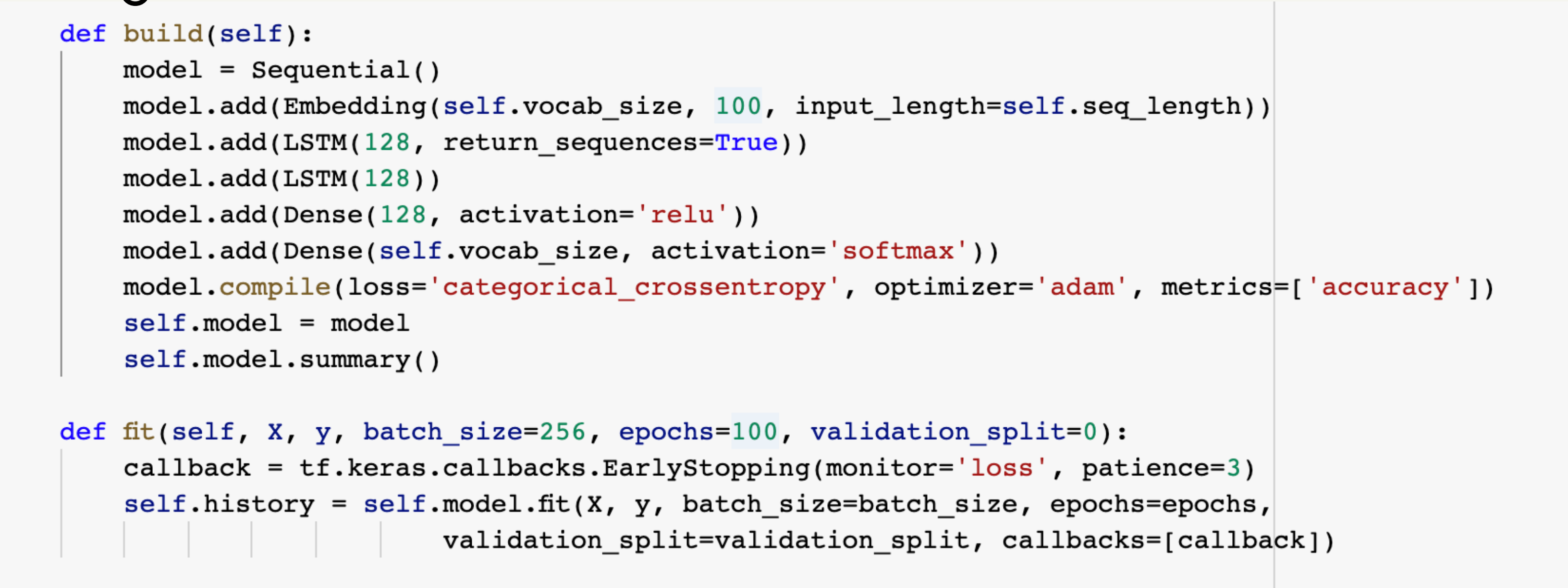
Three different models were built. One with stop words, Second without stop words and Third with stop words and drop-out layers.
In all three the one with drop-out generalised better and had an overall lowest perplexity value.

Project Demo



LSTM Model

Model Architecture and Training Parameters

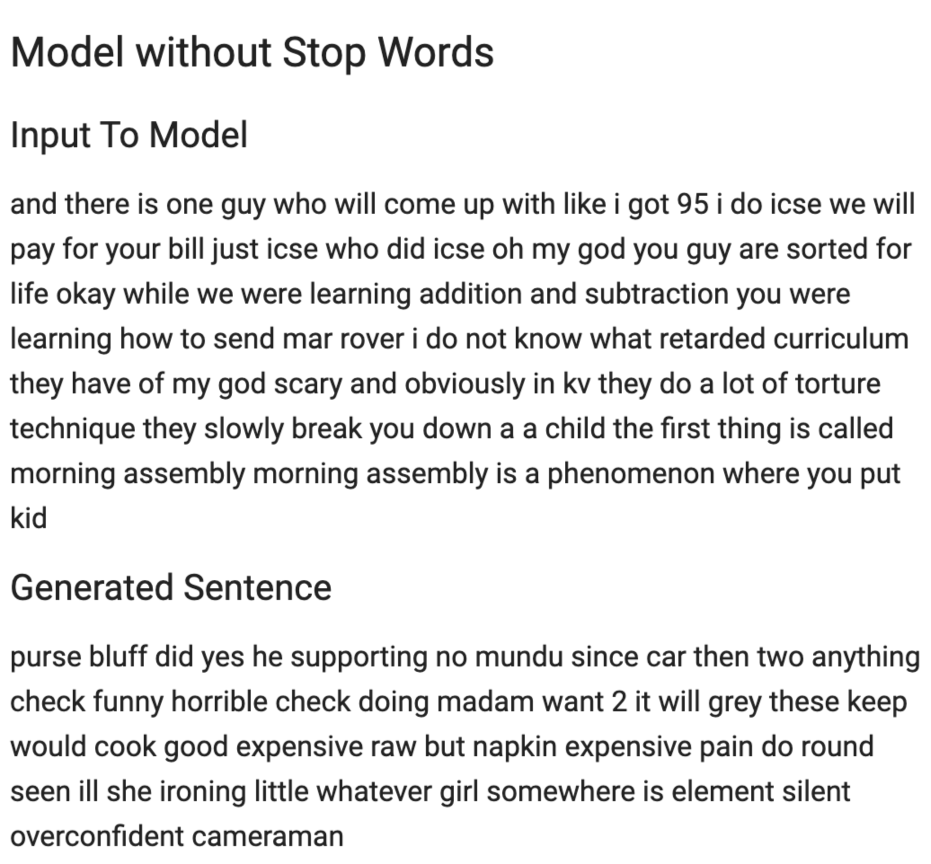


```
def build(self):
    model = Sequential()
    model.add(Embedding(self.vocab_size, 100, input_length=self.seq_length))
    model.add(LSTM(128, return_sequences=True))
    model.add(LSTM(128))
    model.add(Dense(128, activation='relu'))
    model.add(Dense(self.vocab_size, activation='softmax'))
    model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
    self.model = model
    self.model.summary()

def fit(self, X, y, batch_size=256, epochs=100, validation_split=0):
    callback = tf.keras.callbacks.EarlyStopping(monitor='loss', patience=3)
    self.history = self.model.fit(X, y, batch_size=batch_size, epochs=epochs,
            validation_split=validation_split, callbacks=[callback])
```

Model Results

Model without Stop Words



Model without Stop Words

Input To Model

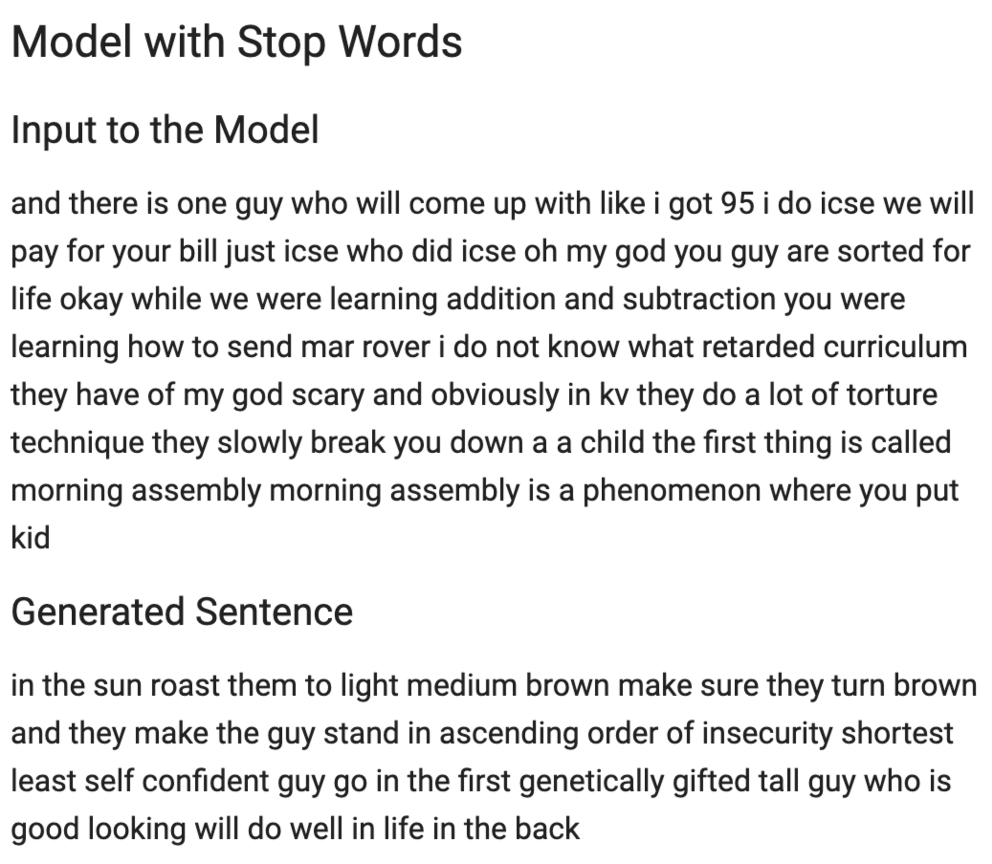
and there is one guy who will come up with like i got 95 i do icse we will pay for your bill just icse who did icse oh my god you guy are sorted for life okay while we were learning addition and subtraction you were learning how to send mar rover i do not know what retarded curriculum they have of my god scary and obviously in kv they do a lot of torture technique they slowly break you down a a child the first thing is called morning assembly morning assembly is a phenomenon where you put kid

Generated Sentence

purse bluff did yes he supporting no mundu since car then two anything check funny horrible check doing madam want 2 it will grey these keep would cook good expensive raw but napkin expensive pain do round seen ill she ironing little whatever girl somewhere is element silent overconfident cameraman

Model Results

Model with Stop Words



Model with Stop Words

Input to the Model

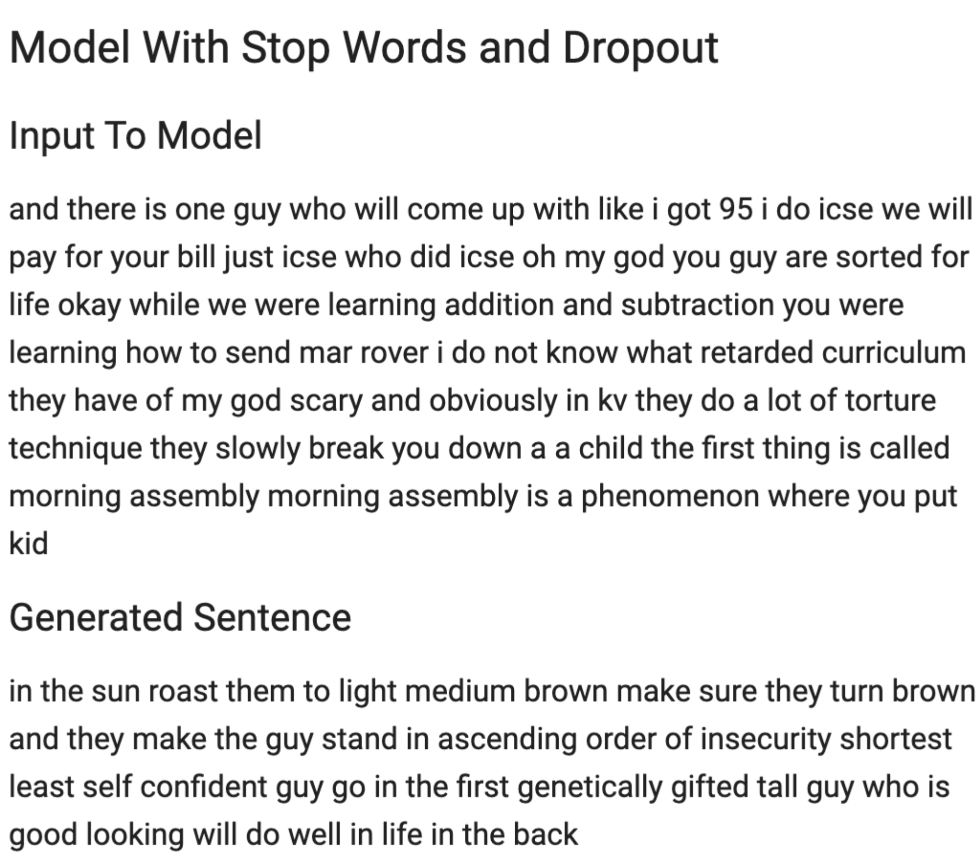
and there is one guy who will come up with like i got 95 i do icse we will pay for your bill just icse who did icse oh my god you guy are sorted for life okay while we were learning addition and subtraction you were learning how to send mar rover i do not know what retarded curriculum they have of my god scary and obviously in kv they do a lot of torture technique they slowly break you down a a child the first thing is called morning assembly morning assembly is a phenomenon where you put kid

Generated Sentence

in the sun roast them to light medium brown make sure they turn brown and they make the guy stand in ascending order of insecurity shortest least self confident guy go in the first genetically gifted tall guy who is good looking will do well in life in the back

Model Results

Model with Stop Words and Dropout



Model With Stop Words and Dropout

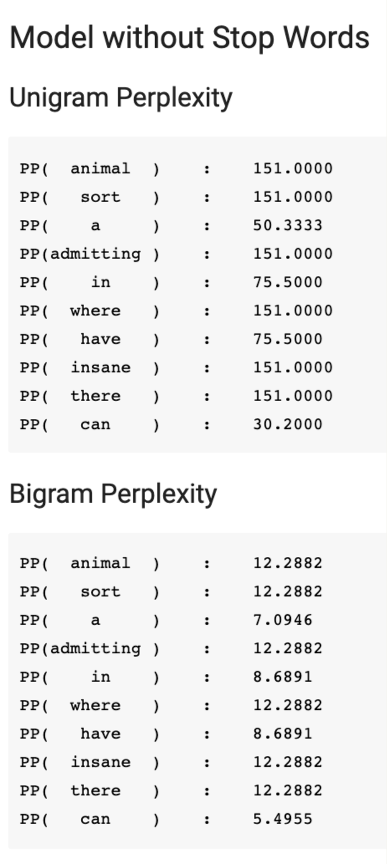
Input To Model

and there is one guy who will come up with like i got 95 i do icse we will pay for your bill just icse who did icse oh my god you guy are sorted for life okay while we were learning addition and subtraction you were learning how to send mar rover i do not know what retarded curriculum they have of my god scary and obviously in kv they do a lot of torture technique they slowly break you down a a child the first thing is called morning assembly morning assembly is a phenomenon where you put kid

Generated Sentence

in the sun roast them to light medium brown make sure they turn brown and they make the guy stand in ascending order of insecurity shortest least self confident guy go in the first genetically gifted tall guy who is good looking will do well in life in the back

Model Evaluation



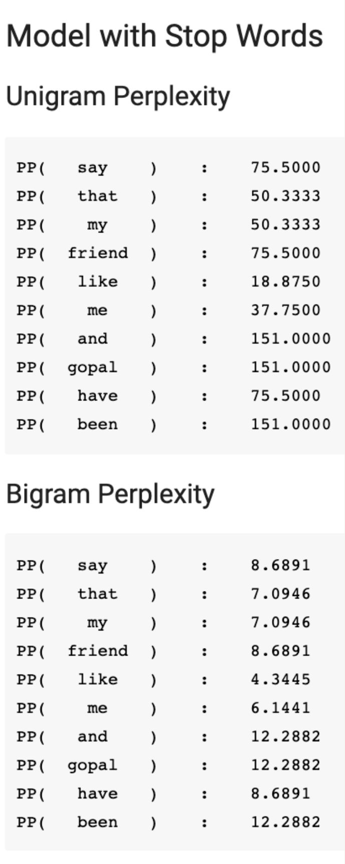
Model without Stop Words

Unigram Perplexity

```
PP(  animal  )    :    151.0000
PP(   sort   )    :    151.0000
PP(    a     )    :    50.3333
PP(admitting )    :    151.0000
PP(    in    )    :    75.5000
PP(  where   )    :    151.0000
PP(   have   )    :    75.5000
PP(  insane  )    :    151.0000
PP(  there   )    :    151.0000
PP(   can    )    :    30.2000
```

Bigram Perplexity

```
PP(  animal  )    :    12.2882
PP(   sort   )    :    12.2882
PP(    a     )    :    7.0946
PP(admitting )    :    12.2882
PP(    in    )    :    8.6891
PP(  where   )    :    12.2882
PP(   have   )    :    8.6891
PP(  insane  )    :    12.2882
PP(  there   )    :    12.2882
PP(   can    )    :    5.4955
```



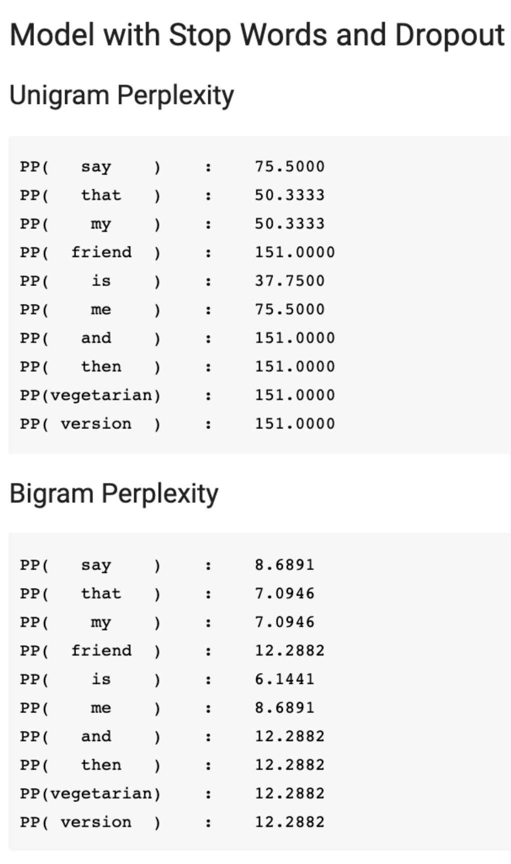
Model with Stop Words

Unigram Perplexity

```
PP(   say    )    :    75.5000
PP(   that   )    :    50.3333
PP(    my    )    :    50.3333
PP(  friend  )    :    75.5000
PP(   like   )    :    18.8750
PP(    me    )    :    37.7500
PP(   and    )    :    151.0000
PP(  gopal   )    :    151.0000
PP(   have   )    :    75.5000
PP(   been   )    :    151.0000
```

Bigram Perplexity

```
PP(   say    )    :    8.6891
PP(   that   )    :    7.0946
PP(    my    )    :    7.0946
PP(  friend  )    :    8.6891
PP(   like   )    :    4.3445
PP(    me    )    :    6.1441
PP(   and    )    :    12.2882
PP(  gopal   )    :    12.2882
PP(   have   )    :    8.6891
PP(   been   )    :    12.2882
```



Model with Stop Words and Dropout

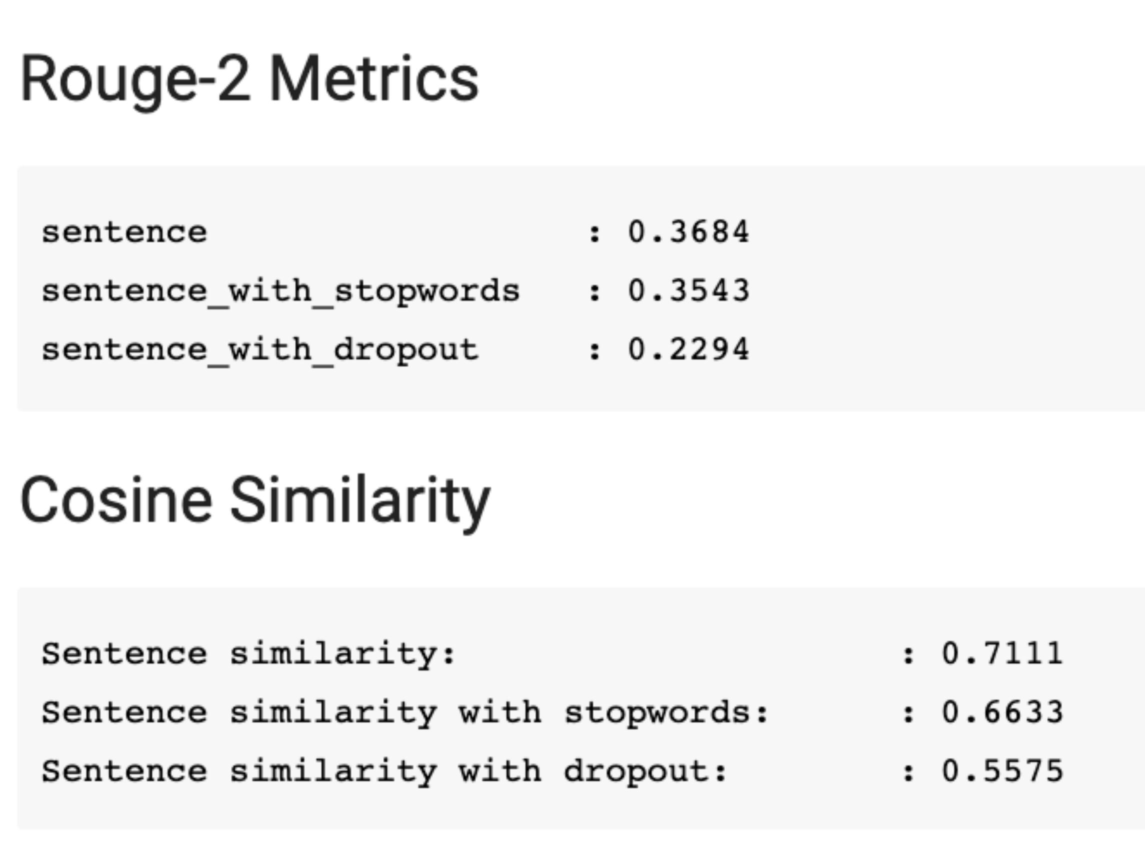
Unigram Perplexity

```
PP(   say    )    :    75.5000
PP(   that   )    :    50.3333
PP(    my    )    :    50.3333
PP(  friend  )    :    151.0000
PP(    is    )    :    37.7500
PP(    me    )    :    75.5000
PP(   and    )    :    151.0000
PP(   then   )    :    151.0000
PP(vegetarian)    :    151.0000
PP( version  )    :    151.0000
```

Bigram Perplexity

```
PP(   say    )    :    8.6891
PP(   that   )    :    7.0946
PP(    my    )    :    7.0946
PP(  friend  )    :    12.2882
PP(    is    )    :    6.1441
PP(    me    )    :    8.6891
PP(   and    )    :    12.2882
PP(   then   )    :    12.2882
PP(vegetarian)    :    12.2882
PP( version  )    :    12.2882
```

Model Evaluation



Rouge-2 Metrics

```
sentence                   : 0.3684
sentence_with_stopwords    : 0.3543
sentence_with_dropout      : 0.2294
```

Cosine Similarity

```
Sentence similarity:                        : 0.7111
Sentence similarity with stopwords:         : 0.6633
Sentence similarity with dropout:           : 0.5575
```

Thankyou

And Future works

Trying different Embeddings

Trying different models

Training on a Larger Dataset