# Using Word2Vec and N-Grams to Create Shakespearean Texts

Hexin Liu
Advisor: Michael Glass

## Introduction

Can AI generate phony Shakespearean texts? Many AI-models specialize in creating artworks based on certain styles of a well-known artist. However, having an AI model that writes literature and mimics an author is less common.

To address this problem, we are focusing on creating a model that will generate sentences in a style of Shakespeare.

We first trained an n-gram model on Shakespeare's collection of works. The n-gram model is usually employed to select among probable word combinations, for example, to pick the best of several possible translations. We are using the model to statistically produce Shakespearean text. We then used the Word2Vec and Glove word embedding models to substitute words with similar semantic meanings and parts of speech.

Many generated texts are grammatically nonsensical. Hence, we are aiming to create code that will screen out these defective lines.

## Materials and Software Methods

- Complete works of Shakespeare from Project Gutenberg.
  - Train n-gram model.
  - Train Word2Vec embedding model.
- Programming language: Python.
- Google Colab platform.
- Part of speech tagging: NLTK and spaCy libraries
- Gensim's Word2Vec modelling and similarities
- Grammar Checking: GingerIT and the Language Tool Python libraries
- Glove pretrained word embedding model (100K words)

## Generating Texts through Trigrams

An N-gram model possesses probabilities of a final word given the preceding n-1 words. For instance, from a Shakespearean trigram model…

"A good __" → {"man": 0.03, "wit": 0.022, "wench": 0.022, "turn": 0.018, "play": 0.018 …"answer":0.004}

We calculated the bigram and trigram models using Shakespeare's collected works. "<s>" and "</s>" were special words to mark the beginning and ending of sentences.

How to Generate Text with Trigrams

- Start with a small number of words.
  "I love"
- Randomly choose the succeeding word according to trigram probability.
  "I love" → "to"
- Choose another word.
  "I love to" → "read"
- Choose the next word.
  "I love to read" → {"sometimes", "Shakespeare", "poems", "many"…}

Results

- Some results are plausible utterances.
- Some are nearly plagiarized.
- Some are nonsensical.

Examples of the texts generated by the trigrams.

**<s> as i am today i th'vein of chivalry </s>**
<s> if you be safer </s>
**<s> then to thee </s>**
**<s> how now a wood near athens </s>**
<s> lear </s>
**<s> therefore to our rose of youth </s>**
<s> exeunt </s>
**<s> a dangerous law against it </s>**
**<s> in any case not that their first of manhood stand upright </s>**
<s> subdu'd me </s>
<s> great timon noble worthy royal timon </s>
<s> ever true in me else </s>
**<s> what would come against us like the heaven's glorious sun </s>**
<s> warwick </s>

**<s> what would come against us like the heaven's glorious sun </s>**

"What would come against us" -- Shakespearean and is not quoted from Shakespeare (good!).

"Study is like the heaven's glorious sun" – Berowne, *Love's Labour's Lost,* Act 1, Scene 1 (plagiarized).

## Transforming Shakespearean Texts

"Like the heaven's glorious sun"

Tagging each word with part of speech

*like/IN the/DT heaven/NNP 's/POS glorious/JJ sun/NN*

- IN – preposition
- DT – determiner
- NNP – proper noun
- POS – possessive
- JJ – adjective
- NN – Noun

Let's use a word embedding model to find noun words suggested by "heaven".

{"god", "eternity", "Satan", "holy"}

Now we are focusing on the adjective "glorious".

{"wonderful", "dreadful", "remarkable", "unforgettable"}

Finally, the noun "sun" has the substitutes:

{"earth", "sunlight"}

Possible final result:

"Like the eternity's dreadful sunlight"

## Future and Current Work

We have produced some adequate sentences; most generated texts have defects. For example, some sentences are simply the name of a character. Other texts, such as "ever true in me else", are nonsensical.

We plan to eliminate the sentences with syntactical errors. We are, by using software toolkits, attempting to parse the sentences and eliminate the ones that cannot successfully produce a syntactically-correct sentence.

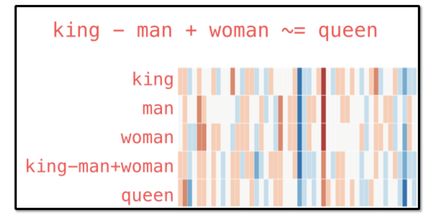
The part of speech tagging can misidentify a word's part of speech. This is especially true of potential replacement words taken out of the context. We are experimenting with substituting most probable words and checking if it alters the syntax of the sentence.

Experimenting with language embedding models trained with only Shakespearean vocabulary vs. modern terminology.

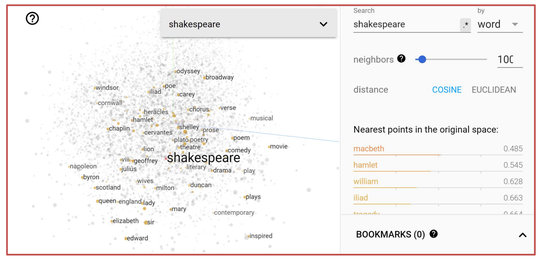
Experimenting with more modern neural network language generation models.

## Word Embeddings

- Each vocabulary word is represented with a vector of floating-point values.
- Words with similar meanings should have similar vector components.
- The cosine similarity between two vectors gives us a way to find words with similar meaning or usage.




A screenshot of the visual form of the word embeddings. Notice the similarity between vectors "queen" and "king – man + woman." (Courtesy of Jay Alammar)

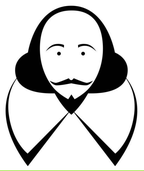



A picture that shows words similar to "Shakespeare" using embeddings trained on various online sources (Courtesy of TensorFlow).

## Acknowledgements

We would like to thank Dr. Michael Glass for assisting in the research process.

We would also like express gratitude for the Department of Computing and Information Sciences.



We would also like to thank this guy.