

Bad Poets Society Logbook

Table of Contents

Table of Contents

1. [Poem Fetching](#)
 - [1.1 Stats](#)
2. [Poem Generation With Markov Chains](#)
 - [2.1 Markov Chains](#)
 - [2.1.1 First-Order Markov Chains](#)
 - [2.1.2 Higher-Order Markov Chains](#)
 - [2.2 Implementation](#)
 - [2.3 Generating Poems](#)
 - [2.3.1 Generation with First-Order Chains](#)
 - [2.3.2 Generation with Second-Order Chains](#)
 - [2.3.3 Generation with Second and Third-Order Chains](#)
 - [2.3.4 New Creativity Assesment](#)
 - [2.4 Mixing Poets](#)
 - [2.5 Self Assessment](#)
 - [2.5.1 Strengths](#)
 - [2.5.2 Limitations](#)
 - [2.5.3 Improvements](#)

1. Poem Fetching

We wanted to have an automted system to fetch and store the poems for training the poem generator system. poetrydb.org had a web api where we can fetch all poems of a author by sending an get request to https://poetrydb.org/author/{poet_name}.

We went with the following authors

- Emily Dickinson: 362 poems
 - Known for concise language, frequent use of dashes, and metaphysical themes.
- Oscar Wilde: 10 poems
 - Wilde's poems often feature aesthetic beauty, classical allusions, and wit.
- William Blake: 52 poems
 - His poetry is characterized by vivid imagery, strong moral undertones and symbolic language.
- Percy Bysshe Shelley: 312 poems

1.1 Stats

We also record some stats per author

```
self.stats = {  
    'number_of_poems': num_poems,  
    'mean_length_of_poems': mean_length,  
    'number_of_unique_words': num_unique_words  
}
```

to analyze the creativity of the system based on the characteristics of the reference data

2. Poem Generation With Markov Chains

We first decided to go with fine tuning a LLM to generate poems as it is the most state-of-the-art approach. However, because of its simple nature and ease of implementation, we decided to use Markov chains.

2.1 Markov chains

2.1.1 First-Order Markov Chains

A first-order Markov chain treats each word in the reference database as a node in a directed graph. Edges between nodes indicate the probability of transitioning from one state to another. These probabilities can be calculated from the reference text by noting each consecutive word tuple and normalizing the values to correspond to probabilities.

2.1.2 Higher-Order Markov Chains

A higher-order Markov chain operates similarly to a first-order chain but considers previous transitions to the current node. For example, a second-order Markov chain takes into account the two preceding words. In this case, the probability of a transition is represented as:

$$P(w_3 \mid (w_1, w_2))$$

Where (n) is the number of unique words in the database, and the chain has (n^2) possible states or nodes. Each node represents the likelihood of transitioning from a word pair (w_1, w_2) to a subsequent word (w_3) .

This enables the modelling of the more complex relationships between words and hopefully comes up with meaningful sentences.

2.2 Implementation

We first tried to do our own implementation in Python since the algorithm is fairly simple where we record the probabilities in a $n \times n$ sparse matrix in the training stage and then traverse the chain by "flipping a coin" in each node and following the transition based on the transition probabilities. However, we quickly decided to go with an existing python implementation called **markovify** since its much more efficient and supports higher-order chains out of the box. Also, some of its features came in handy in the later stages where we tried to avoid the model just copying existing poems.

With **markovify** implementing a Markov chain is as simple as

```
model = markovify.Text(text, state_size)
```

2.3 Generating Poems

At first, we tried to generate a poem with a single order Markov chains with Emily Dickinson poems. The majority of the experimentation is done with Emily Dickinson's poems. We then tested the system with the rest of the poets with the optimum parameters obtained from the experimentation.

2.3.1 Generation with first order chains

We can get a sentence from the Markov model like

```
line= self.model.make_sentence()
```

We first generated 3 sentences. Markovify internally handles the termination by treating terminal punctuations as a separate termination node. A sentence is finished once we reach the termination node.

Generated Poem 1.1: *The Moon or of Thee -- By innocent Surprise Cool us -- how Nature -- No more slowly The other, as yet no more -- Its unobtrusive Mass. No Trace -- Intact -- Tell Him -- And so bolder every step For Whom I stood -- the site of a pilgrim be Adjusted -- Prosperity -- And yet it -- They said the Wall In Nature feels. The Forest of thee to scan -- Which solemnizes me. Loose the backward leaves Us homesick, who ne'er so I stepped upon my strong trust -- With God.*

The length of sentences were more than the average sentence line in the database. This is either a result of the randomness of the first-order model or how the termination is treated with termination punctuation marks. Since the use of punctuation in the poems doesn't follow conventional use. So we tried treating end of lines as termination by

```
model = markovify.NewlineText(self.text, state_size=state_size)
```

This resulted in better generations that matched the reference poems.

Generated Poem 1.2:

Alloys our names -- adjust the fragrant Cocks --

Propitiation's claw -- guessed it --

And so, and shake your inference appalling

Generated Poem 1.3:

Thou art not -- like a Hell can see --

It pleased my Knitting --

And whatsoever is Theirs unto the House of Itself -- a cheek --

Generated Poem 1.4:

To hold my Life, My own Grave -- the fire will give him as Hours slid fast --

If they waded -- Her Lips of their loveliness

Is but One Claw --

These poems better matched our expectations. At this point, the way we assessed the creativity of these systems was solely based on our opinions. We **desperately** searched for meaning in the generated lines. The word choice and the style of these poems resembled Emily Dickinson's but as if she didn't speak English.

In the search for getting better meaning out of the system, we experimented with higher-order Markov chains.

2.3.2 Generation with second order chains

Generated poem 2.1 *That like the Sunrise -- left the door of God!*

Tell Him -- When He makes the Fences smile --

From flasks -- so utter --

Generated poem 2.2

The foliage of the night --

The Death I never doubted him -- too

This pattern -- of mine

Generated poem 2.3

As we -- pray -- to take the honorable Work

Peace is a sun --

Just the Day will be

These poems started to make much more sense and started to make us feel something. As a result, we were eager to try higher-order Markov chains.

2.3.3 Generation with second and third-order chains

As we increased the state size the model struggled to generate sentences. We guess this is a result of the model being too complex for the amount of data. This resulted in the model skipping some lines and giving some one-liners. Which we liked! We also experimented with other termination strategies and lines per poem.

Generated poem 2.4

Presuming Me to be a Century --

Till He be justified by Death -- but that was easy --

Generated poem 2.5

I know I heard the Buckle snap --

Generated poem 2.6

So soon to be a Century --

Generated poem 2.7

And Twilight -- in the Sea --

Generated poem 2.8 (4rd Order chain) Pray gather me -- Anemone -- Thy flower -- be gay -- Her Lord -- away!

Generated poem 2.9 (4rd Order chain)

*Angels, when the sun is hottest May be seen the Dews among, Stooping -- plucking -- sighing -- flying --
Parched the flowers they bear along*

These poems especially 2.8 and 2.9 almost brought tears to our eyes. But we also started to have suspicions on the authenticity of these generated texts. To investigate the issue we implemented a plagiarism checker that compares the similarity of n-grams of the generated text to the reference database in a sliding window fashion.

Here is the output of the system for a poem generated with a 4. order Markov chain

Poem 2.10: As should sound -- to me -- so signal -- The hue -- of it -- by Architect

Potential plagiarism detected:

Generated Section: 'As should sound -- to'

Original Section: '-- As should sound --'

Title of Original Poem: 'Teach Him -- When He makes the names'

Similarity: 0.86

Generated Section: 'should sound -- to me'

Original Section: 'As should sound -- to'

Title of Original Poem: 'Teach Him -- When He makes the names'

Similarity: 0.86

Generated Section: 'sound -- to me --'

Original Section: 'sound -- to me --'

Title of Original Poem: 'Teach Him -- When He makes the names'

Similarity: 1.00

Generated Section: '-- to me -- so'
Original Section: '-- to me -- If'
Title of Original Poem: 'I prayed, at first, a little Girl,'
Similarity: 0.86

Generated Section: 'to me -- so signal'
Original Section: 'to me -- so signal'
Title of Original Poem: 'One Life of so much Consequence!'
Similarity: 1.00

Generated Section: 'me -- so signal --'
Original Section: 'to me -- so signal'
Title of Original Poem: 'One Life of so much Consequence!'
Similarity: 0.83

Generated Section: '-- so signal -- The'
Original Section: 'me -- so signal --'
Title of Original Poem: 'One Life of so much Consequence!'
Similarity: 0.81

Generated Section: 'so signal -- The hue'
Original Section: 'so signal -- That I'
Title of Original Poem: 'One Life of so much Consequence!'
Similarity: 0.82

Generated Section: '-- The hue -- of'
Original Section: '-- The hue -- of'
Title of Original Poem: 'The name -- of it -- is "Autumn" --'
Similarity: 1.00

Generated Section: 'The hue -- of it'
Original Section: 'The name -- of it'
Title of Original Poem: 'The name -- of it -- is "Autumn" --'
Similarity: 0.85

Generated Section: 'hue -- of it --'
Original Section: 'name -- of it --'
Title of Original Poem: 'The name -- of it -- is "Autumn" --'
Similarity: 0.84

Generated Section: '-- of it -- by'
Original Section: '-- of it -- is'
Title of Original Poem: 'The name -- of it -- is "Autumn" --'
Similarity: 0.86

Generated Section: 'of it -- by Architect'
Original Section: 'of it -- by Architect'
Title of Original Poem: 'Heaven is so far of the Mind'
Similarity: 1.00

As can be seen, the system is mostly cheating by stitching sections of poems together. In the example above it is getting sections from

- Teach Him -- When He makes the names
- I prayed, at first, a little Girl
- One Life of so much Consequence!
- The name -- of it -- is "Autumn
- Heaven is so far of the Mind

However, to be fair it is stitching these sections in a rather meaningful way and the outputs are very pleasing.

Here is the output of the system for a poem generated with a 3. order Markov chain

Poem 2.11:

This -- to Heaven --

Is enough for me -- today --

All this -- and more -- if I troubled them --

Generated Section: 'enough for me -- today'

Original Section: 'enough for me -- Fade'

Title of Original Poem: "'Twould ease -- a Butterfly --'

Similarity: 0.84

Generated Section: 'for me -- today --'

Original Section: 'for me -- today --'

Title of Original Poem: 'You see I cannot see -- your lifetime'

Similarity: 1.00

Generated Section: '-- All this -- and'

Original Section: '-- All this -- and'

Title of Original Poem: 'A Murmur in the Trees -- to note'

Similarity: 1.00

Generated Section: 'All this -- and more'

Original Section: 'All this -- and more'

Title of Original Poem: 'A Murmur in the Trees -- to note'

Similarity: 1.00

Generated Section: 'this -- and more --'

Original Section: 'All this -- and more'

Title of Original Poem: 'A Murmur in the Trees -- to note'

Similarity: 0.82

Generated Section: '-- and more -- if'

Original Section: '-- and more -- if'

Title of Original Poem: 'A Murmur in the Trees -- to note'

Similarity: 1.00

Generated Section: 'and more -- if I'

Original Section: '-- and more -- if'

Title of Original Poem: 'A Murmur in the Trees -- to note'

Similarity: 0.85

Generated Section: 'more -- if I troubled'

Original Section: 'see -- if I troubled'

Title of Original Poem: 'Why -- do they shut Me out of Heaven?'

Similarity: 0.88

Generated Section: '-- if I troubled them'

Original Section: '-- if I troubled them'

Title of Original Poem: 'Why -- do they shut Me out of Heaven?'

Similarity: 1.00

Generated Section: 'if I troubled them --'

Original Section: '-- if I troubled them'

Title of Original Poem: 'Why -- do they shut Me out of Heaven?'

Similarity: 0.86

Poem 2.12:

Then caught me -- like a Claw --

Came out to look at her --

I never felt at Home -- I know it!

Generated Section: 'Then caught me -- like'

Original Section: '-- Then caught me --'

Title of Original Poem: 'Three times -- we parted -- Breath -- and I --'

Similarity: 0.81

Generated Section: 'caught me -- like a'

Original Section: 'Then caught me -- like'

Title of Original Poem: 'Three times -- we parted -- Breath -- and I --'

Similarity: 0.83

Generated Section: 'me -- like a Claw'

Original Section: 'me -- like a Ball'

Title of Original Poem: 'Three times -- we parted -- Breath -- and I --'

Similarity: 0.82

Generated Section: '-- like a Claw --'

Original Section: '-- like a Ball --'

Title of Original Poem: 'Three times -- we parted -- Breath -- and I --'

Similarity: 0.82

Generated Section: 'like a Claw -- Came'
Original Section: 'like a Claw -- I'
Title of Original Poem: 'It would have starved a Gnat --'
Similarity: 0.86

Generated Section: '-- Came out to look'
Original Section: 'Came out to look at'
Title of Original Poem: 'I started Early -- Took my Dog --'
Similarity: 0.84

Generated Section: 'Came out to look at'
Original Section: 'Came out to look at'
Title of Original Poem: 'I started Early -- Took my Dog --'
Similarity: 1.00

Generated Section: 'out to look at her'
Original Section: 'stop to look at her'
Title of Original Poem: 'The Soul has Bandaged moments --'
Similarity: 0.86

Generated Section: 'to look at her --'
Original Section: 'To look at her --'
Title of Original Poem: 'When I was small, a Woman died --'
Similarity: 0.94

Generated Section: 'look at her -- I'
Original Section: 'To look at her --'
Title of Original Poem: 'When I was small, a Woman died --'
Similarity: 0.85

Generated Section: 'I never felt at Home'
Original Section: 'I never felt at Home'
Title of Original Poem: 'I never felt at Home -- Below'
Similarity: 1.00

Generated Section: 'never felt at Home --'
Original Section: 'I never felt at Home'
Title of Original Poem: 'I never felt at Home -- Below'
Similarity: 0.88

Generated Section: 'felt at Home -- I'
Original Section: 'felt at Home -- Below'
Title of Original Poem: 'I never felt at Home -- Below'
Similarity: 0.84

Generated Section: 'at Home -- I know'
Original Section: 'at Home -- I know'
Title of Original Poem: 'I never felt at Home -- Below'
Similarity: 1.00

Generated Section: 'Home -- I know it!'

Original Section: 'at Home -- I know'
 Title of Original Poem: 'I never felt at Home -- Below'
 Similarity: 0.80

As can be seen from poems 2.11, 2.12 and their plagiarism output. The system is still stitching different sections together. However, compared to the 4. order chain. It's stitching smaller sections and changes some words occasionally.

Here is the output of the system for a poem generated with a 2. order Markov chain

Poem 2.13:

With such a Fate -- to dream --

Is always as the Sun

But vanquished her -- How slowly

Generated Section: 'With such a Fate --'
 Original Section: 'To such a Fate --'
 Title of Original Poem: 'She's happy, with a new Content --'
 Similarity: 0.83

Generated Section: 'such a Fate -- to'
 Original Section: 'To such a Fate --'
 Title of Original Poem: 'She's happy, with a new Content --'
 Similarity: 0.82

Generated Section: '-- to dream -- Is'
 Original Section: '-- To dream -- to'
 Title of Original Poem: 'A Pit -- but Heaven over it --'
 Similarity: 0.82

Generated Section: 'Is always as the Sun'
 Original Section: 'Is always as the contents'
 Title of Original Poem: 'I thought that nature was enough'
 Similarity: 0.80

Generated Section: 'always as the Sun But'
 Original Section: 'always as the contents But'
 Title of Original Poem: 'I thought that nature was enough'
 Similarity: 0.81

Generated Section: 'Sun But vanquished her --'
 Original Section: 'down That vanquished her --'
 Title of Original Poem: 'The pretty Rain from those sweet Eaves'
 Similarity: 0.81

The stitching theme still continues but the sections are smaller compared to 3. order chains and more words in the sections are mutated.

Here is the output of the system for a poem generated with a 1. order Markov chain

Poem 2.14:

Protects our Freight

Beyond our Faith is Death to Your little doubt that

But when they all the Sun shone whole --

Potential plagiarism detected:

Generated Section: 'all the Sun shone whole'

Original Section: 'fair The Sun shone whole'

Title of Original Poem: 'The Trees like Tassels -- hit -- and swung --'

Similarity: 0.85

Generated Section: 'the Sun shone whole --'

Original Section: 'The Sun shone whole at'

Title of Original Poem: 'The Trees like Tassels -- hit -- and swung --'

Similarity: 0.86

Poem 2.15:

Nor tossed me -- upset the flock away -- lost her Dyes

To Me -- Wherefore fly?

Upon the furthest --

Generated Section: 'upset the flock away --'

Original Section: 'led the flock away --'

Title of Original Poem: 'I'll tell you how the Sun rose'

Similarity: 0.86

Generated Section: 'To Me -- Wherefore fly?'

Original Section: 'saw them -- Wherefore fly?'

Title of Original Poem: 'We pray -- to Heaven'

Similarity: 0.82

Generated Section: 'Me -- Wherefore fly? Upon'

Original Section: 'them -- Wherefore fly? Is'

Title of Original Poem: 'We pray -- to Heaven'

Similarity: 0.80

With first-order chains, the plagiarism problem is mostly resolved it is not copying whole sections but using small phrases from different poems. However, the meaning of the poems is mostly lost. With second order chains meaning is still not compromised. Therefore 2. order chains seem to be the optimum point balancing meaning with originality.

New Creativity Assesment 2.3.4

Up to this point, we assessed the creativity of the generated poems based on our tastes and opinions. After this point, we included the plagiarism checker's output in the assessment pipeline. Therefore we don't think the poems generated by 2. 3. and 4. Markov chains are "creative" since it is not novel. 1. order chains do produce novel output but the meaning is lost. Therefore we also don't consider most of the output created by 1. order chains as creative. In order to get a creative system with our new definition, we decided to add some randomness to the 2. order markow chains so that it doesn't copy large texts but still produces meaningful poems. Luckily markovify library provides us some parameters while generating.

```
line = self.model.make_sentence(  
    tries,  
    max_overlap_ratio,  
    max_overlap_total  
)
```

Sometimes the model will return None as a sentence if it can not return one within the specified constraints. We allow multiple retries to overcome that problem. max_overlap_ratio is the maximum allowed overlap between the words of a generated sentence and the words of any input sentence in the training data. A maximum number of overlapping words allowed between a generated sentence and any sentence in the training data. We mostly utilized the max_overlap_total parameter and set it between 3-5. This did prevent the system from making large copies. When we set it to 3 it copies sections of poems with max 3 words and it started to copy from texts much less often. Here are some of the results.

Poem 3.1

You'll know it -- in the Grave?

```
Generated Section: 'know it -- in the'  
Original Section: 'know it -- And then'  
Title of Original Poem: 'Going to Him! Happy letter!'  
Similarity: 0.89
```

```
Generated Section: 'it -- in the Grave?'  
Original Section: 'Mother -- in the Grave'  
Title of Original Poem: 'He told a homely tale'  
Similarity: 0.83
```

Poem 3.2

How know it -- in the night!

*You'll know it -- in the dark! *

Pernicious as the Sun -- It is Infinity.

Generated Section: 'How know it -- in'
Original Section: 'to know it -- And'
Title of Original Poem: 'Going to Him! Happy letter!'
Similarity: 0.82

Generated Section: 'know it -- in the'
Original Section: 'know it -- And then'
Title of Original Poem: 'Going to Him! Happy letter!'
Similarity: 0.89

Generated Section: 'it -- in the dark!'
Original Section: 'is in the dark! I'
Title of Original Poem: 'My wheel is in the dark!'
Similarity: 0.80

Generated Section: 'dark! Pernicious as the Sun'
Original Section: 'bar. Pernicious as the sunset'
Title of Original Poem: 'Through those old Grounds of memory,'
Similarity: 0.82

Poem: 3.3

Which for myself --

As eligible as the Sun

Close to the Air --

"Generated Section: '-- As eligible as the'
Original Section: 'way As eligible as the'
Title of Original Poem: 'Summer -- we all have seen --'
Similarity: 0.88

Generated Section: 'As eligible as the Sun'
Original Section: 'way As eligible as the'
Title of Original Poem: 'Summer -- we all have seen --'
Similarity: 0.82"

Poem: 3.4

If they the felicity

Pray do not dare to go,

My flowers from a Stem

Generated Section: 'Pray do not dare to'

Original Section: 'I do not dare to'

Title of Original Poem: 'Consulting summer's clock,'

Similarity: 0.86

Generated Section: 'do not dare to go,'

Original Section: 'would not dare to go,'

Title of Original Poem: 'Where Roses would not dare to go,'

Similarity: 0.87

Generated Section: 'not dare to go, My'

Original Section: 'not dare to go, What'

Title of Original Poem: 'Where Roses would not dare to go,'

Similarity: 0.84

Generated Section: 'to go, My flowers from'

Original Section: 'Morn -- My flowers from'

Title of Original Poem: 'As Children bid the Guest "Good Night"'

Similarity: 0.80

Generated Section: 'go, My flowers from a'

Original Section: '-- My flowers from a'

Title of Original Poem: 'As Children bid the Guest "Good Night"'

Similarity: 0.88

Generated Section: 'My flowers from a Stem'

Original Section: '-- My flowers from a'

Title of Original Poem: 'As Children bid the Guest "Good Night"'

Similarity: 0.81

Poem: 3.4

I missed mine -- in White!

Nor any know I heard

And parts of his Covert

Generated Section: 'I missed mine -- in'

Original Section: '-- I missed mine --'

Title of Original Poem: 'God permits industrious Angels'

Similarity: 0.84

Generated Section: 'heard And parts of his'

Original Section: 'today, And parts of his'

Title of Original Poem: 'I prayed, at first, a little Girl,'

Similarity: 0.80

Generated Section: 'And parts of his Covert'
Original Section: 'And parts of his far'
Title of Original Poem: 'I prayed, at first, a little Girl,'
Similarity: 0.84

Poem: 3.5 *Abhorrent is the friend*

The Ripple of our Faith --

To the stately sails

Generated Section: 'The Ripple of our Faith'
Original Section: 'The Ripple of our Theme'
Title of Original Poem: 'I learned -- at least -- what Home could be --'
Similarity: 0.83

Generated Section: 'Ripple of our Faith --'
Original Section: 'Ripple of our Theme --'
Title of Original Poem: 'I learned -- at least -- what Home could be --'
Similarity: 0.82

Generated Section: 'of our Faith -- To'
Original Section: 'of our Faith -- The'
Title of Original Poem: 'Peace is a fiction of our Faith --'
Similarity: 0.92

Generated Section: 'our Faith -- To the'
Original Section: 'of our Faith -- The'
Title of Original Poem: 'Peace is a fiction of our Faith --'
Similarity: 0.84

These generated poems fit within our definition of criteria and were rather pleasing to read. We tried to not over restrict the model such as giving very little overlap possibility otherwise the model would give repetitive poems such as:

Poem 3.6

But, which is not bold. Transport -- by the Snow. Wouldn't the Angels -- lone. When I was a dream. But, which is not bold.

Poem 3.7

A something in the ground. Wouldn't the Angels -- lone. Transport -- by the Snow. Wouldn't the Angels -- lone. But, which is not bold.,,

In the end, we found the settings below to be the most creative

```
line = self.model.make_sentence(  
    tries=20000,  
    max_overlap_ratio=0.8,  
    max_overlap_total=4  
)
```

2.4 Mixing Poets

With the optimum settings, we obtained from the experimentation above. We decided to apply to system with different poets and also create a system which generates poems with mixed influence.

Poem 4.1 William Blake: *Yet I am indeed:*

Who in sorrow too?

And with soft down

No plagiarism detected.

Poem: 4.2 Willaim Blake *Walks in the wilds*

When the night in disguise.

The sun does shine,

Generated Section: 'Walks in the wilds When'
Original Section: 'rages in the wilds Where'
Title of Original Poem: 'The Marriage of Heaven and Hell: The Argument'
Similarity: 0.81

Poem: 4.2 Oscar Wilde

That they will not despise.

Made for the holy night,

Blown rose of flame,

No plagarium detected

Poem 4.3 Oscar Wilde

These things are done

And some men weep,

Some do the secret mystery

Generated Section: 'are done And some men'
 Original Section: 'are young, And some when'
 Title of Original Poem: 'The Ballad Of Reading Gaol'
 Similarity: 0.80

Generated Section: 'And some men weep, Some'
 Original Section: 'and some men weep, And'
 Title of Original Poem: 'The Ballad Of Reading Gaol'
 Similarity: 0.80

Generated Section: 'some men weep, Some do'
 Original Section: 'some men weep, And some'
 Title of Original Poem: 'The Ballad Of Reading Gaol'
 Similarity: 0.80

Poem 4.4 Percy Bysshe Shelley: Or as the fabulous Thamondocana,-- A dark and bloody, But I can find

Generated Section: 'Or as the fabulous Thamondocana,--'
 Original Section: 'way, Beyond the fabulous Thamondocana,--'
 Title of Original Poem: 'The Witch of Atlas'
 Similarity: 0.81

Generated Section: 'as the fabulous Thamondocana,-- A'
 Original Section: 'way, Beyond the fabulous Thamondocana,--'
 Title of Original Poem: 'The Witch of Atlas'
 Similarity: 0.82

Generated Section: 'the fabulous Thamondocana,-- A dark'
 Original Section: 'the fabulous Thamondocana,-- Where, like'
 Title of Original Poem: 'The Witch of Atlas'
 Similarity: 0.83

Generated Section: 'A dark and bloody, But'
 Original Section: 'been dark and bloody, Yet'
 Title of Original Poem: 'The Cenci. a Tragedy in Five Acts'
 Similarity: 0.81

Poem 4.5 Emily Dickinson - Oscar Wilde:

Were in the place

But when the Grave --

And when the labouring oar,

Generated Section: 'the place But when the'
Original Section: 'the Queen. But when the'
Title of Original Poem: 'CHARMIDES'
Similarity: 0.80

Generated Section: 'But when the Grave --'
Original Section: 'Just when the Grave and'
Title of Original Poem: 'I cried at Pity -- not at Pain --'
Similarity: 0.82

Generated Section: 'when the Grave -- And'
Original Section: 'when the Grave and I'
Title of Original Poem: 'I cried at Pity -- not at Pain --'
Similarity: 0.83

Generated Section: 'the Grave -- And when'
Original Section: 'the Place -- And when'
Title of Original Poem: '"Heaven" has different Signs -- to me --'
Similarity: 0.86

Generated Section: 'Grave -- And when the'
Original Section: 'Life -- And when the'
Title of Original Poem: 'There is a flower that Bees prefer'
Similarity: 0.83

Generated Section: '-- And when the labouring'
Original Section: '-- And when the Sun'
Title of Original Poem: 'Nature -- the Gentlest Mother is,'
Similarity: 0.82

Generated Section: 'And when the labouring oar,'
Original Section: 'and when the labouring day'
Title of Original Poem: 'CHARMIDES'
Similarity: 0.87

Poem: 4.6 Oscar Wilde - William Blake

And the moon in the dark,

And at the day.

How can Lyca sleep?

Generated Section: 'And the moon in the'
Original Section: 'And the iron gin that'
Title of Original Poem: 'The Ballad Of Reading Gaol'
Similarity: 0.80

Generated Section: 'moon in the dark, And'
 Original Section: 'rose in the dark, And'
 Title of Original Poem: 'The Chimney-Sweeper'
 Similarity: 0.86

Generated Section: 'in the dark, And at'
 Original Section: 'rose in the dark, And'
 Title of Original Poem: 'The Chimney-Sweeper'
 Similarity: 0.80

Generated Section: 'the dark, And at the'
 Original Section: 'the dark, And got with'
 Title of Original Poem: 'The Chimney-Sweeper'
 Similarity: 0.86

Generated Section: 'day. How can Lyca sleep?'
 Original Section: 'child. How can Lyca sleep'
 Title of Original Poem: 'The Little Girl Lost'
 Similarity: 0.86

Poem: 4.7 Emily Dickinson - William Blake

There is a morn

Subject to Your Courtesies

Trodden with the voice

Generated Section: 'There is a morn Subject'
 Original Section: 'there is a morn But'
 Title of Original Poem: 'His Heart was darker than the starless night'
 Similarity: 0.81

Generated Section: 'a morn Subject to Your'
 Original Section: 'pass -- Subject to Your'
 Title of Original Poem: 'Some such Butterfly be seen'
 Similarity: 0.80

Generated Section: 'to Your Courtesies Trodden with'
 Original Section: 'to Your Courtesies Than this'
 Title of Original Poem: 'Take your Heaven further on'
 Similarity: 0.81

2.4 Art Generation

We chose some of the poems we enjoyed and used the stable diffusion model with the prompt

2.5 Self Assessment

2.5.1 Strengths:

The system successfully generates poems that reflect the styles of Emily Dickinson, Oscar Wilde, and William Blake. In addition to generating creative, stylistically consistent poems, the system successfully integrates visual representations using the Art Generator. The images generated by the Stable Diffusion model enhance the understanding and appreciation of the poems by adding a visual dimension. This is particularly effective in capturing the symbolic and aesthetic elements present in poems by William Blake and Oscar Wilde. The use of a Markov Chain with a state size of 2 ensures coherence while maintaining creativity.

2.5.2 Limitations:

The plagiarism check reveals a few areas where the generated text is too similar to the original. Additionally, the system occasionally produces repetitive lines, such as "Though the morning air," indicating that the model might benefit from further refinement. While the visual art generated enhances the poetic experience, there are occasional mismatches between the abstractness of the poems and the concreteness of the images produced by the Art Generator. In some cases, the visual output doesn't fully align with the deeper symbolic meaning of the poems.

2.5.3 Improvements

s: Future work could involve experimenting with different Markov Chain configurations or integrating a more advanced language model to reduce the likelihood of replication and improve the depth of the generated poems. Future work could involve refining the Art Generator to produce more realistic images that better capture the complexity of the poetry. Additionally, experimenting with different art styles and prompts could lead to more accurate visual representations.