8-31-2021

The Emily Dickinson Machine & Hybrid Poetry Generation

Juliana Shihadeh

Santa Clara University

Department of Computer Science & Engineering
Date: August 31, 2021

I HEARBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY

Juliana Shihadeh

ENTITLED

The Emily Dickinson Machine & Hybrid Poetry Generation

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTERS OF SCIENCE IN COMPUTER SCIENCE AND ENGINEERING

DocuSigned by:

6055A39379B649F...

Thesis Advisor: Dr. Margareta Ackerman

DocuSigned by:

Ahmed Amer

9C8772571AEC45B...

Thesis Reader: Dr. Ahmed Amer

DocuSigned by:

Nam Ling

96CE94A1A83A48A...

Chairman of Department: Dr. Nam Ling

Acknowledgments

To start with, I'd like to thank my advisor Professor Ackerman for giving me the opportunity to explore my ideas and pursue my research with her. I want to thank her for encouraging thinking outside of the box, nurturing an environment of innovative thinking and giving insightful feedback on my results, analysis, and writing which has helped me become a better writer and analyzer. This project originally started in one of her classes, Computational Creativity, and has been a privilege and fun learning journey to expand. I've learned a lot from Dr. Ackerman and am grateful for the opportunity to have worked with her. I'd like to thank Professor Amer for accepting and taking the time to be my Thesis reader; his input and attention to detail were invaluable. I want to thank all the professors at Santa Clara University I took a class with for having shaped my technical background, taking the time to answer my questions during office hours, and providing opportunities to grow as both an independent thinker and team contributor. I'd like to thank everyone who volunteered to participate in the user studies for our research; their participation was of valuable contribution. Lastly, I want to thank my parents & siblings for their continuous support, and my parents for encouraging me to follow my passions.

The Emily Dickinson Machine & Hybrid Poetry Generation

by Juliana Shihadeh
Advisor: Dr. Margareta Ackerman
2021

Department of Computer Science & Engineering
Santa Clara University
Santa Clara, California

Abstract

This thesis introduces EMILY, a machine that creates original poems in the style of renowned poet Emily Dickinson. Dickinson's succinct and syntactically distinct style with unconventional punctuation makes for an interesting challenge for automated poetry creation. Furthermore, we adapt EMILY to answer the following hypothetical question: What if Emily Dickinson had collaborated with another poet from a different time period? To this end, we introduce *Hybrid Generative Poetry*, which simultaneously integrates poetic elements from multiple poets. Using two distinct approaches to Hybrid Poetry generation, we create poetry in the combined styles of Emily Dickinson and Robert Frost. User studies are conducted to evaluate both EMILY and the Hybrid Poetry approaches.

Contents

1 Introduction

1.1 Background

Poetry writing is an artform dating back to ancient times [8]. A successful poem elicits imagery and evokes emotion through an interlock of relationships between semantics, syntax, grammar, punctuation, rhythm and rhyme. Machine-generated poetry is a subfield of natural language processing that emerged in the late 1990s [27]. Machine generated poetry can itself be viewed as an artform distinct from human made poetry, with computer generated poems created across human languages through a variety of computing techniques (see, for example, [19], [45] and [11]). Furthermore, many machine generated poetry methods focus on stylistic elements like rhyme, format, and topic ([43], [13], [21]).

Machine-generated poetry research has so far focused on two opposing aims: style imitation, which aims to replicate the style of individual poets, and poetry machines that aim to capture the essence of poetry, intentionally avoiding replicating any poet's specific style. Style imitation is typically accomplished by training on the data of one particular poet, and has been applied to the works of Italian poet Dante Alighieri [47], Bob Dylan lyrics [1], and the works of William Shakespeare and Oscar Wilde, amongst others [38]. Meanwhile, generative poetry aiming to create typical poems (without imitating the style of any particular poet) has been created across a variety of languages, including Finish [11], English [24], Chinese [45], Portuguese [26] and Spanish [25].

Poetic style imitation offers the opportunity to immortalize a poet by keeping their voice alive through novel works. From an evaluation standpoint, the generated works can be compared with those of the original creator, enabling a variation of the Computational Creativity (CC) Turing Test by checking whether unbiased observers are able to discern generated artifacts from original ones. Other variations involve comparing the original and generated works on important criteria (ex. stylistic elements of poetry) to help identify where improvement is needed.

1.2 Our contributions

One of the greatest English poets, Emily Dickinson (1830-1886), is known for effectively capturing feeling and imagery using few words [7]. Dickinson's style is revealed through unique use of punctuation, syntax, formatting and rhyme [7]. Her succinct and potent poetry makes Dickinson an interesting challenge for style imitation. In this thesis, we present EMILY, a poetry machine that aims to replicate the style of Emily Dickinson's poems. We present the methodology behind EMILY, along with a user study that compares machine-created poems with Emily Dickinson originals on several poetic criteria.

Furthermore, the advance of readily accessible systems such as OpenAI's GPT models are making poetry generation increasingly easier [34]. While the impact of this increased democratization may be complex, we additionally propose that one of its positive impacts is that it may give rise to opportunities to more readily explore conceptually novel creative spaces. What if instead of focusing on either extreme, we intentionally create poetry that simultaneously reflects the style of a small number of poets? This direction lets us explore an intriguing hypothetical question: What if two poets, who perhaps may not have lived in the same time and place, have had a chance to collaborate?

In this thesis, we further focus on expanding the boundaries of machine-made poetry by experimenting with the production of Hybrid Poetry that aims to simultaneously reflect the style of a small number of poets. We demonstrate our findings in Hybrid Poetry by focusing on one specific pair: Emily Dickinson and Robert Frost.

Robert Frost (1874-1963) is often compared to Dickinson in that they both take simple every day matters of life, a fact, object or person and extract its significance by transforming it into a piece of art illustrated with words [30]. Frost represents common yet fragile elements of life by drawing comparisons to nature, such as death in "Stopping by the woods on a Snowy Evening" and the end of the world in "Fire and Ice". He writes in more monologue and dialogue styles compared to Dickinson's succinct style. Robert Frost is also known for blending colloquialism and traditional meters, writing in unprecedented ways subsequently imitated by other poets [30]. By combining Frost an Dickinson's works, we merge two styles with such similar perspective on life yet distinct writing style. The combination of similarities and difference in their styles, makes Dickinson and Frost a particularly interesting pair for this initiate investigation into generative Hybrid Poetry.

1.3 Relevant Work

Poetry generation falls into three main categories: Mere generation, human enhancement, and computer enhancement [17]. Over the years, methods evolved across all three categories introducing various ways to create machine generated poetry.

In earlier poetry generation research, mere generation was most common, including approaches such as filling in templates. Approaches included adapting existing poems to new words and following grammatical rule-based systems to create a poem [27], [21], [43], [39]. One of the earliest statistical machine generated poetry techniques used n-grams, which are similar to Markov Chains [18]. Newer statistical models use techniques such as applying multiple constraints, filters, and orchestrating models to train per line rather than per stanza [15], [12]. Additional models applied to generate poetry include Support Vector Machines, which follow an n-gram method to predict what the next syllabus should be [4]. Lastly, Markov Chains has also been used to produce poetry [23], [41], [3], [40] and song lyrics which are similar to poetry in style such as in meter, use of rhyme, stanzas, and length[1].

In recent years, progress in deep learning has especially pushed what is possible in poetry generation, a computer enhancement technique. Deep learning has brought forward multiple applications to generate poetry using acombined Convolutional-Recurrent Neural Network (CNN-RNN) [21], Recurrent Neural Networks (RNNs) [21], [13], [44], [10], [9], [46], and Long Short Term Memory networks (LSTM) [38]. Other neural networks have been also used to generate rap poetry lyrics with LSTM [31]. Transformer models that use a mechanism named attention are the most recent breakthrough of deep learning architectures and have been applied to poetry generation [2], [14]. BERT and GPT-2 architectures are especially known as advancing the quality of artifacts attention models are able to create [20], [22].

Human Enhancement in poetry generation introduces a more interactive method for poetry generation. It involves more collaborative systems where a human is involved in the process of writing by editing the output. For example, Gnoetry is an ongoing experiment of collaborative poetry generation software that involves a human in the writing process. Its goal is for a human to help guide the creative process so there is equal collaboration between a person and the computer generating the poem [17], [6], [35]. As another related example, Lyric Studio acts like a co-writer to help individuals overcome their writer's block. Even though it focuses on lyrics, lyrics have a very similar form and style to a poem such as rhyme, meter, use of stanzas, and length. The machine provides suggestions for what to write and a person guides the creative process by picking the next phrase from the suggestions and altering different setting to shape the style of outputs.

Of the various methods discussed above to generate poetry, our work uses Markov Chains and Deep Learning, specifically the GPT-2 model. In general, it has been rare to see research conducted on developing custom Markov Chains. Even more, we found only a few applications of Markov Chains adopted to capture a specific poet's style like we do. Furthermore, we have only found two well established machine poetry generators that have focused on generating poetry distinctly in Emily Dickinson's style. However, they fall short on capturing the particular syntax elements in her poetry [40], [3]. Dickinson, though, is one of the most famous American poets thus showing opportunity and importance to dedicate more research in poetry generation on Dickinson specifically [7]. Moreover, previous works in poetry generation haven't looked at combining the style of a small group of poets like we do in our research.

1.4 Poetry Generation Methods

In this section, we provide an introduction to the most important poetry generation methods for the current work: Markov Chains and Transformer Models.

1.4.1 Markov Chains Background

Markov Chains originated from research conducted by mathematician Andreyevich Markov (1856-1922). A Markov Chain is a stochastic model, where the next state of a system depends only on the current state, not previous states [48]. Thus in Markov Chains, the history of a system does have a direct impact on the next state, only one step directly effects what the next state will be. Secondly, the transition of a state i to j from time k to $k+1$ does not depend on time k [48]. A state can be defined as any value within the problem's nature, such as one word in poetry.

Before Markov, research in probability mainly focused on independent variables[48]. However, Markov focused on exploring sequences of random variables that weren't independent of each other[48]. He initiated his research by looking at poetry. To explore the dependency of variables, he took segments from the poetry of "Eugeny Onegin" taking out spaces, punctuation marks, and soft letters (letters that aren't pronounced)[48]. He then studied the patterns of consonants and vowels in the sequences turning it into a 2-state Markov Chain with the state space $\{c, v\}$, trying to evaluate if there was a dependency between consonants and vowels in the sequence[48]. With that, Markov originated research on Markov Chains.

1.4.2 Transformer Models Background

Models used for sequence data tend to rely on Recurrent Neural Networks or Convolutional Neural Networks, however, the problems with such networks are (1) the sequential nature of the network and, (2) the distance that data needs to travel through the different layers. "Attention is All You Need" introduced Transformer models which use an attention mechanism that allows parallelization of the computation, overcoming limitations of sequential processing RNNs/CNNs[42]. Attention is a matrix that represents the weight of how much one token is affiliated to another to help represent how likely they are to follow each other in a generated sentence; a token can be a space, punctuation or word [42][36]. As a result, Transformer models can process longer sequences of information because they can keep track of more words in the past [42]. Transformers include the GPT-2 and BERT model [34], [36].

Typically sequence based neural networks use sequential layers in an encoder-decoder architecture [42]. For an attention dependent neural network like the Transformer to process in parallel rather than sequentially, two central components are used: multi-head attention layers and scaled dot-products [42]. One multi-head attention layer is built by combining multiple attention layers in which each use the scaled dot-product attention mechanism [42]. The attention is a simple mapping from an input to output represented as weights in a matrix that are derived as a result of these scaled dot-products [42]. All the attention layers within a multi-head attention layer run in parallel, allowing greater capability for relationships between multiple tokens to be analyzed at once rather than in sequence [42]. In turn, it helps the model better understand a language because it's able to see further into the past and have more context about a sentence or phrase [42]. The multi-head attention layer computes attention between tokens on 3 main levels: it attends to all tokens in an input sequence, it attends to all tokens from the encoder's previous layer's output, and it attends to every token up to the position of the current token the attention of which is being analyzed [42].

GPT-2, the model we use in our work, uses attention to predict what a next generated word should be [34], [28]. It learns the grammar of a language so it can be applied to various tasks and topics rather than only ones of the same nature of the dataset on which it trained [28]. As a result this allows GPT-2 to be more accessible to various domains. It can be used to automate answering questions, to analyze and comprehend text it is given, for machine translation and also for poetry generation [2], [14], [28].

1.5 Overview

Our contribution is twofold. First, we present EMILY, a poetry machine that aims to replicate the style of Emily Dickinson's poems. We specifically focus on creating custom Markov Chains that are able to capture both the syntax and format style of Emily Dickinson. Next, we investigate Hybrid Generative Poetry, which are poems that combine the style of a small group of poets to simulate the possibility of a real-life collaboration between them. We particularly concentrated on combining the style of

Emily Dickinson and Robert Frost in our research thus far. We present the methodology of both methods, along with user studies we conducted. Our user studies evaluate what stylistic elements are present in the generated poems through an adaption of the Computational Creativity Turing Test and a comparison of original to generated works on various artistic criteria.

2 EMILY: Poetry Generation in Emily Dickinson's Style

We now introduce EMILY a poetry generator that uses Markov Chains. We focus on capturing Dickinson's unique punctuation and syntax style to reproduce her writing manner more effectively. We custom build our Markov Chains to allow us more substantial guidance of granular details that help enhance the artistic outcome of the generated poems. Her style is addressed in all three stages of the process: pre-processing the data, selecting the words in the poem, and post-processing the selected words to tailor the poem's format accordingly. In order to generate poetry in Dickinson's style, the data we use is exported from three online publicly available books of her works. We discuss how our system works, share examples of the generated poetry and present the results of our user studies.

2.1 Method

We now describe our methodology. Specifically our use of data pre-processing, our custom Markov chains and how we post-process the selected words.

2.1.1 Data Preprocessing

EMILY was trained on publicly available Emily Dickinson poetry from the Gutenberg project: "Poems by Emily Dickinson, Three Series, Complete by Emily Dickinson" [5, 32]. The data consisted 444 poems, consisting of 10178 lines.

Punctuation meaningfully contributes to Dickinson's unique style and as such deserves careful treatment. We saved commas, periods, question marks, and semi-colons. Dickinson is well known for her uses of dashes [7], which were also preserved. Some punctuation, particularly all brackets, were omitted because they introduced noise without helping to capture Dickinson's style.

Dickinson used to number instead of title most of her poems. We discarded all roman numerals in our preprocessing since our focus is on generating the poems' bodies.

The final preprocessing step was to convert any fully-capitalized words found in the poem titles into lower case. This helped to enrich the data set of Dickinson's words. Words that start with capital letters were left unchanged because Dickinson used capitalized words in the middle of sentences [7].

2.1.2 Custom Markov Chains

To endow EMILY with Dickinson's style, we chose to build our own custom Markov Chains. This gave us greater control over the creative process, particularly as it pertains to punctuation, which is a central element of Dickinson's poetry. Barbieri [1] also observed that unmodified Markov Chains were insufficient for capturing style, in their case as it pertains to Bob Dylan's use of rhyming.

The Markov Chains implementation relies on a dictionary. We create the Markov Chains by iterating through all the words and reading them in reverse. Starting with the first word, we iterate for each word at index i checking if the prior word appears in the dictionary. If so, we add the word to its list of values. If the word before it is not

Figure 1: The poet Emily Dickinson (1830-1886). Photo Credit: Yale University Manuscripts Archives Digital Images Database.

in the dictionary, we add it to the dictionary and start its list of values with the current word as the first word. As a result, we map each word to all the words that proceed it in Dickinson's writing. Doing so lets us capture the relationship of what words show up after each one along with their frequency to create the dictionary that represents the Markov chains. Words with higher frequency have a higher probability of being generated. Our final dictionary had a total of 8610 keys. Markov Chains are used to generate the sequence of words for the poems. We format the generated words in the postprocessing phase.

Starting Word

For single stanza poems, we randomly select the initial word from all words used in Dickinson's writing. If the poem has more than one body, we rely on the final word in the previous body in order to generate the first word in the sequence body using the Markov process.

Body

Each stanza in a poem is 20 words long. This keeps the poems at approximately the length of Dickinson's poems, which consist of short stanzas of 4-5 lines each with 5-6 words per line. The number of stanzas generated for each poem is determined by a variable n passed to EMILY.

Closing Word

To help bring out Dickinson's style, concluding words were chosen from amongst those that had punctuation.

2.1.3 Postprocessing: Formatting the Poems

Not only is the choice of words in the poem important to capturing Emily Dickinson's style, but the format of the poem brings in important stylistic elements. We format the poems based on an analysis of Dickinson's poetry.

Dickinson starts poems with capitalized words, and also follows periods, exclamation marks, or question marks. Words that follow a comma or semi-colon are generally lowercase. More importantly, Dickinson is known for capitalizing words in the middle of sentences, not only words that begin a new line [7].

We traverse through the final list of words and set a flag based on the type of punctuation to determine if the following word should start with a capital or lowercase letter. Following Dickinson's style [7], any capitalized words not preceded by a comma or semi-colon are left unchanged. The generated list of words is then divided into five word sentences, and the first letter of each sentence is capitalized.

2.2 User Study

We evaluate EMILY by comparing its machine-created poems to Emily Dickinson originals on several criteria. This study seeks to gain an understanding of the quality of EMILY's poems. Larger and more in depth studies are left to future work.

We surveyed 17 participants, 9 female and 8 male. On a scale of 0-5, 0 being "Not at all Familiar" with Emily Dickinson's poetry and 5 being "Extremely Familiar", 3 participants responded with a 4, 5 responded with a 3, 4 with a 2, 1 with a 1 and 4 with a 0.

Participants were presented with a total of 12 poems, consisting of 10 of EMILY's poems and 2 poems by Emily Dickinson. The original poems are Poem 6, *"Faith" is a fine invention*, and Poem 12, *Come Slowly—Eden*, which capture many of her stylistic elements.

The choice of questions was influenced by previous work evaluating machine-made poetry [47, 11, 16]. For each of the 12 poems, participants were asked the following:

1. Is this a typical poem?
2. Is this poem understandable?
3. How much do you like the word choice in the poem?
4. Does the text evoke mental images?
5. Does the text evoke emotion?
6. Do you like this poem?

Each question was answered by selecting from a Likert scale: Strongly disagree (0), disagree (1), neutral (2), agree (3), strongly agree (4). The scores of each question were averaged across all respondents for each poem, as shown in Figure 2. The scores of each question were also averaged across all generated poems versus the original Emily Dickinson poems, shown in Figure 3.

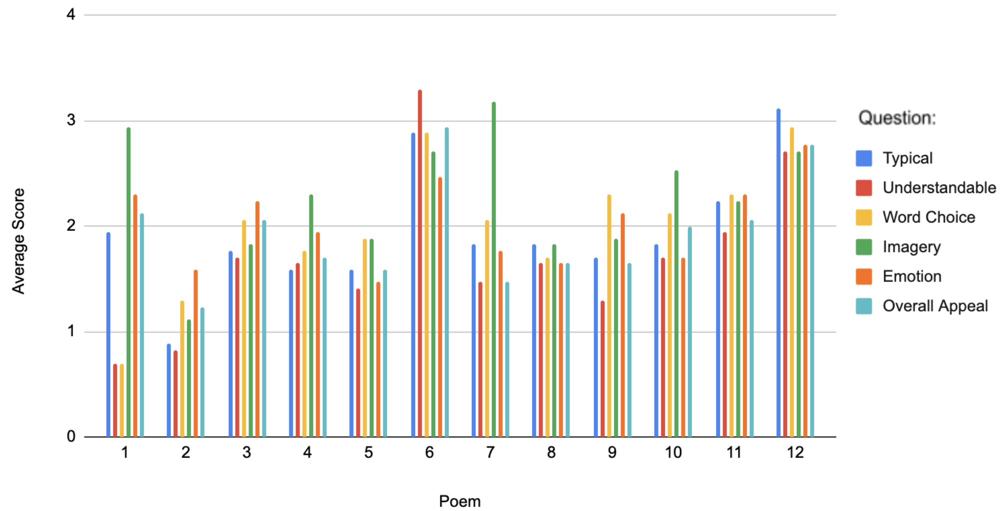

Figure 2: Average scores of the six listed questions above for each poem based on the Likert Scale. The legend shows the main theme of each question in the same order of the full questions, which are listed above. Poems 6 and 12 correspond to original Emily Dickinson poems, while the others were created by EMILY.

2.2.1 Results

Our survey shows that question 4, "Does the text evoke mental images?", had the highest average score of 2.17 of all questions for generated poems. Furthermore, *the average score of question 4 outranked the average score for Emily Dickinson's poems in 2 of the generated poems*. Poem 1, 7, and 10 had the highest score for question 4 as seen in Figure 2. Poems 1, 7 and 10 appear at the end of this section.

Three of our generated poems resulted in at least 3 out of the 5 questions averaging to a score higher than 2, in a range similar to Emily Dickinson's poems' average scoring of 2-3 (Poem 1, 3, and 11). Each of these poems performed well on a different set of questions.

The question "Is this poem understandable?" resulted in the lowest average score across all our generated poems as seen in Figure 3, with a score of 1.43 across all generated poems. Dickinson's poems averaged to a score of 3 on this question. The questions "Is this a typical poem?" and "Do you like this poem?" averaged to 1.72 and 1.76, respectively, identifying areas for improvement.

Most of EMILY's poems resulted in average scores of around 2. Dickinson's poems resulted in average scores closer to 3 with question 4 "Does the text evoke mental images?" and question 5 "Does the text evoke emotion?" averaging out to the mid-2s at about 2.71 and 2.6, respectively.

The overall average scores of Emily Dickinson's poems were higher across all questions compared to our generated poems as seen in Figure 2, which offers an interesting challenge in future work. EMILY's poems fared well compared to Emily Dickinsons' poems, averaging to a score of about 2 while Dickinson's averaged to scores closer to 3 with only two in the mid-2s range.

To give the reader a better sense for EMILY's poetry, we conclude this section with the 3 top performing generated poems in the study.

Poem 1

Some shook their yellow gown
And certainly her eye, they
Leap upon the rose smiling
To die. The orchards Eternity!

Poem 7

The wondrous dear, –An
Enemy is the gate the
Children caper when liked, –
Might but a year, hunted,

Tis all can put out
A little plan to his
Eternal chair, his notice to
Pass odors so dense notoriety.

Poem 10

Surrendering the 'house at Lexington,
And then of snow; the
Orchard sparkled like perfidy. A
Year, nor heedless were small,

For 't was to a
Watch, some sweet birds jocoser
Sung; the reason that could
Not put it until mystery!

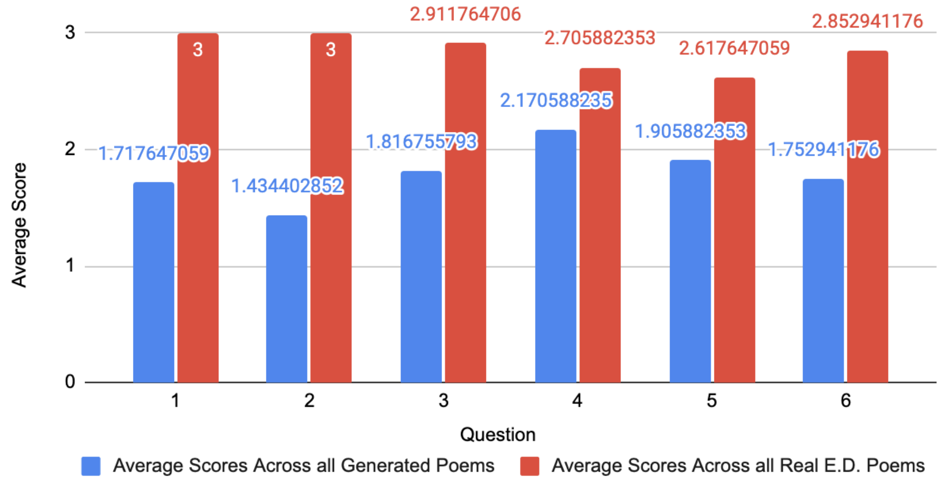

Figure 3: Average scores of questions across all EMILY poems versus average scores of questions across original Dickinson poems.

2.3 Comparing our custom made vs. python library Markov Chains

We compare the results of our custom markov chains model to using a built-in python Markov Chains library, PyPi's Markovchain 0.2.5[1], using the same Emily Dickinson poetry used for EMILY as the data to run Markovchain on. Our analysis suggests that our custom method is able to produce poems that capture Emily Dickinson's style more closely, with respect to punctuation, formatting, and overall stylistic similarly. Two examples of poems created with the python library Markovchain are shown below.

Example 1

In the pumpkins in dungeons are known her final inch, cham-
ber and firmaments row of the last included
both, danced to see by side, i failed to me.

Example 2

But murmuring of the bewildering thread;'s curtain fell, your
way soft descent among the sky!

[1]`https://pypi.org/project/markovchain/`

2.4 Summary

We presented EMILY, a machine that aims to create poems in the style of renowned poet Emily Dickinson. Dickinson's efficient and effective use of words to evoke emotion and imagery, along with distinct syntactic choices, make this an ambitious task. Future development of our work can be guided by the findings of our user study, which highlights areas for improvement.

Our user study compares EMILY's poems with original poetry by Emily Dickinson on several dimensions, such as typicality, understandable, and ability to evoke emotion and imagery. The analysis shows that the generated poetry evokes mental images, at times even better than Dickinson's poems. However, perhaps unsurprisingly, on average, the original poetry scored higher than the machine-made poems. This presents the interesting challenge of automatically creating poetry on par with Emily Dickinson's.

An examination comparing the poems generated using our custom Markov Chains with those made with a Markov python library, suggests that the custom model yields better results. Control over stylistic nuances, through, for example, saving words along with their punctuation, seems to help capture Dickinson's style, and may be relevant to poetic style imitation of other poets.

3 Hybrid Generative Poetry

This section presents our investigation into Hybrid Poetry generation, aimed at simulating collaboration between poets. Dickinson and Frost were selected for this study of Hybrid Poetry for their focus on similar themes yet different writing styles. We present Hybrid Poetry created with these two poets' styles using two approaches. First, we adapt EMILY [37], a poetry machine that replicates the style of Emily Dickinson, to the creation of Hybrid Poetry that combines the styles of Dickinson with that of Robert Frost. This is accomplished by allowing the machine to reflect Dickinson's style through her unique use punctuation and formatting, while training the machine on the words of Robert Frost. Our second approach utilizes OpenAI's GPT2 [34], a neural network model for various text generation tasks that include producing poetry. GTP2 is used to reflect the style of both poets by combining their poetry in the training data. We present our two approaches, share the resultant poetry, and present two user studies that begin to explore how to evaluate Hybrid Poetry machines.

3.1 Method

We create Emily Dickinson-Robert Frost Hybrid Poems by adapting EMILY [37] and utilizing the Transformer Model GPT-2 [34]. These two approaches are discussed below.

3.1.1 Data

The original EMILY [37] was trained on publicly available Emily Dickinson poetry from the Gutenberg project: "Poems by Emily Dickinson, Three Series, Complete by Emily Dickinson" [5, 32]. In order to create Hybrid Poetry we also used Robert Frost poems. Data was added from the Gutenberg project for Robert Frost: "Selected Poems by Robert Frost" [33], consisting of 43 poems, totalling 3,684 lines. Emily Dickinson's data is made up of 444 poems consisting of 10,178 lines. However, the number of words are similar in both datasets for Dickinson and Frost: 30,392 and 21,664 total words each respectively.

3.1.2 EMILY-Based Approach

EMILY [37] is a machine designed to create poetry in the style of Emily Dickinson. It utilizes custom Markov Chains, along with pre-processing and post-processing designed to bring out Dickinson's unique stylistic choices. EMILY is designed to capture Dickinson's use of punctuation and formatting (for example, stanza length, line length, and use of dashes and capitalization), which highlight some of the most unique aspects of Dickinson's writing.

In this section, we provide an overview of our adaption of EMILY to generate Hybrid Poetry in the combined style of Emily Dickinson and Robert Frost. To this end, EMILY's model is retrained on data of Robert Frost's poetry, without any Dickinson poetry in the data. In doing so we apply Dickinson's syntax style to Frost's diction, resulting in a hybrid style that reflects elements from both poets.

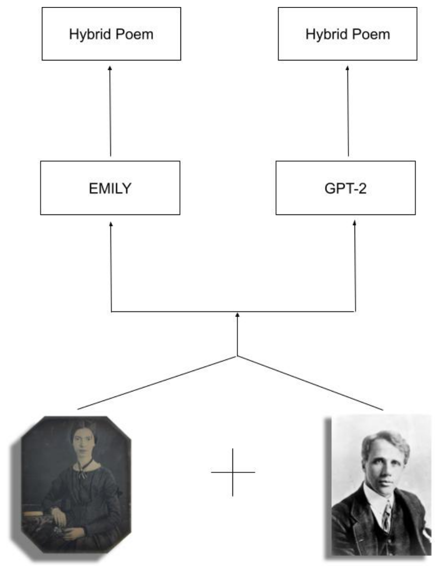

Figure 4: Hybrid Poetry combining the styles of Dickinson & Frost using two approaches: (1) an adaption of the EMILY machine and (2) GPT-2. Photos of Dickinson & Frost: Public Domain. Photo of Dickinson: Amherst College Archives and Special Collections. Photo of Frost: Library of Congress.

After pre-processing our datasets, the total words of Emily Dickinson and Robert Frost poetry used with EMILY are 29,060 and 21,025 each respectively. We run EMILY on one dataset at a time depending on the poem we aim to generate, a Dickinson poem or a hybrid Dickinson-Frost poem. The total unique words in the python dictionary we create to figure out the next state in Markov Chains, again after pre-processing is applied, are 8,609 and 5,097 for Dickinson and Frost respectively.

3.1.3 GPT-2 Approach

The GPT-2 Model we used is Max Woolf's wrapper GPT-2-Simple that embeds fine-tuning[2]. To create Hybrid Poetry, we fine-tuned the default small 124M GPT-2 model on data that contains both Dickinson's and Frost's poetry. The total words we used to fine-tune the GPT-2 model on Emily Dickinson and Robert Frost poetry are 30,392 and 21,664, respectively. In providing both of their poetry, the model studies the patterns of both the diction and syntactic style of both Dickinson and Frost. To generate poetry, we varied the length of the poem as a parameter and set the creativity parameter to 0.7 with the aim of increasing the artistic element of the poetry generated, without going too far from the training set. The GPT-2 model at times produces a single poem and at other times, multiple poems within a single generation.

[2]https://github.com/minimaxir/gpt-2-simple

3.2 User Study

We ran two studies in order to explore how Hybrid Poetry machines may be evaluated. One of the primary aims of the surveys is to determine whether the styles of the original poets remain identifiable. To this end, we wished to find individuals with extensive knowledge of poetry, reaching out to professors and students of poetry.

Our surveys are divided into two main parts. One section focuses on the participants' background and the second has them evaluate poems. The same three background questions are included in both surveys. We ask the user to identify the frequency at which they read poetry, the frequency of writing poetry, and how many years they studied poetry. Reading and writing poetry is evaluated using the Likert scale (Never, Very Rarely, Rarely, Occasionally, Frequently and Very Frequently). Years of studying poetry is evaluated in years categorized into 0, 1-2, 3-4, 5-6, 7+.

3.2.1 Survey 1

The poems evaluated in this study were Hybrid and Emily Dickinson generated poetry. Our goal was to evaluate whether the stylistic elements of the original poets were identifiable in the Hybrid Poetry. We include 4 Hybrid Generated Poems (2 EMILY-based Hybrid Poems and 2 GPT-2 Hybrid Poems) and 4 Dickinson-only generated poems (2 using EMILY and 2 using GPT-2). 3 participants took this survey. All participants studied poetry 7+ years and frequently read poetry, some either very rarely or occasionally wrote poetry. We asked participants to list themes & stylistic elements they identified and poets the poem's style reminded them of.

3.2.2 Survey 1 Findings

The experts were able to identify themes and styles similar to those often associated with Dickinson [29] and Frost's poetry [30]. Common themes identified are longing, despair, regret, love, fear. Additional themes participants mentioned were the impending nature of time, the promise of morning, observations of moments, nostalgic, doubt, rhetorical, and personal. A common stylistic element users noted is the use of free verse. Additional common stylistic elements listed were grammatical playfulness, metaphor, blank verse, and stream of consciousness. In additional free-form feedback, one participant wrote they saw a "play with capitalization," which connects to Dickinson's style.

For the free-form question "What poet/s does the style of the poem remind you of?" few responses directly listed Dickinson or Frost. Surprisingly, the poets were more often identified in Hybrid Generated Poetry (using both approaches) than in Dickinson-only generated poetry. Other poets listed include Matsuo Basho, Mary Oliver, Virginia Woolf, Rudyard Kipling, William Shakespeare, E.E Cummings. It is worth noting poets who followed Dickinson and Frost may have been influenced by both of their styles.

Poem I - Hybrid EMILY

Magnified apples appear and circling
Arms around the difference. The
Barn.—- We're giving back to
Live, just as much laugh,

When the breath of branches,
Climbing through, half an open
Book! moved out, and washing
Dishes after him so done.

Poem II - Hybrid GPT-2

The morning fluttered and fluttered,
The morning stood still,
Its tints like shadow
Still of itself like sand.

Poem III - E.D EMILY

So breathless ballot Lingers to
Be, till we may dim;
Haste! lest while never found
It settled with flowers. Resting,

Knelt in my heart, unshriven!
The verses all the journey.
Our faith to dwell in
Memory. Neither patriarch nor pay.

3.2.3 Survey 2

The poems evaluated in this study were both generated poetry and original poetry written by each of Dickinson and Frost. This survey used a more direct questioning approach. This was influenced in part by a study to evaluate style imitation in generated poetry [38]. The goal was to see what poet the participants would select as the author given a list of options. We include 6 poems: 2 Hybrid Generated Poems (one created using the EMILY-based approach, and another using GPT-2), 2 Dickinson-style generated poems (one using EMILY and one using GPT-2) and 2 original poems (one by Frost and another by Dickinson). 4 students who've studied poetry took the survey. Participants' study of poetry ranged between less than a year to 4 years. For each poem, we asked the participant to select the author of the poem from the options: Robert Frost, Emily Dickinson, A Computer.

3.2.4 Survey 2 Findings

All participants thought the EMILY computer generated Emily Dickinson poem (Poem III) was written by Emily Dickinson rather than the computer. However, for the Hybrid Poetry there was a mix of selections between all 3 options. This shows it may be more difficult to identify the authors of Hybrid Poetry. The GPT-2 and EMILY models performed similarly, resulting in the same number of participants selecting Frost, Dickinson and Computer for each version. 50% of the students misidentified the author of the real Robert Frost poem, selecting Emily Dickinson as the author instead. All successfully identified the real Emily Dickinson poems.

Poem IV - Hybrid GPT-2

I had no time to hate, because
The grave it was, and long ago I died
I lived.

I had no time to love; because
The reason I could not love was
The time that had just come, and since
The beginning had lived.
The way I looked at it made me love
So better than anyone else could,
And I loved it all the more because
The things that made me love them so.

3.3 Summary

We introduced Hybrid Poetry generation as a new direction for creative systems and investigate methods for evaluating Hybrid Generated Poetry. Hybrid Poetry simultaneously reflects the styles of multiple authors, allowing us to create novel works based on imagined collaborations between poets who may have lived in different times and places. This study considered two famous poets, Emily Dickinson and Robert Frost, who wrote on similar schemes using different stylistic methods.

Despite having a small number of participants, these two user studies were helpful in identifying promises and challenges in evaluating Hybrid Generated Poetry. Our surveys show findings that advanced poetry experts were able to identify Dickinson and Frost influences in the Hybrid Generated Poetry, without being given a list of options. However, poetic influences were more often than not misidentified.

Our analysis reveals that, while some experts were able to recognize Dickinson's and Frost's styles in the Hybrid Poetry, there is perhaps unreliability, including failing to identify a poet's own original poetry. Even though we had expert study participants, the survey results suggest difficulty of correctly identifying the authors in Hybrid Poetry. This is further complicated by the fact that some students were unable to correctly identify Frost's original poetry, making it challenging to rely on their feedback when it comes to identifying poetic influences in Hybrid Poetry. One solution would be to raise the bar for what qualified as a poetry expert (our first survey with experts with more than 7 years of experience showed better results), requiring, for example, a threshold on the number of years of poetry study. However, it is possible that this would not be sufficient, as there may be inherent challenges with detecting style when poetic influences are intermixed.

Another EMILY Emily Dickinson - Robert Frost Hybrid Poem

Spare me. By going all
Gone. Try to speak of
Jewels, a man's work, especially
By four or anything. Here.

June our dwelling place? We
Ran light, fumbling the woods
That slowly dawned behind her
Through my overalls, with them.

Time—-i never bore him—-such a
Wall to town. This is
Will turn the ground. What
Is the company. I'm work.

Come too. I'm going clear
The window; Ask them made
The fire's died and wonder
Where they're picking. The waited.

4 Conclusions & Future Work

In this thesis, our contributions were twofold: introducing EMILY, an Emily Dickinson machine, and exploring generative Hybrid Poetry. EMILY distinctly focuses on capturing the syntax and format style of Emily Dickinson, one of the greatest English poets known for succinct imagery invoking poetry with unique punctuation usage. Hybrid Poetry captures the style of a small group of poets rather than one poet only or a large group of poets. In particular, we unite Emily Dickinson and Robert Frost to merge their poetic styles and demonstrate a potential glimpse of what could have been if they had lived in the same time period and collaborated. For both systems, we present our methodologies and an analysis of our user studies.

Poetic style imitation can preserve the voice of a poet[3]. A poet's style can be revived through generative poems to create new poems that honor their style, from their choice of words to their syntax and formatting. With Hybrid Poetry, we introduce generation methods that expand the boundaries of machine generated poems. A Hybrid Poem represents what work the hypothetical collaboration of a small group of poets or specifically one pair of poets could have led to.

Several notable findings were made in the evaluation of EMILY and Hybrid Poetry generation. In particular, some of EMILY's generated poems evoke mental images even better than Dickinson's. However, overall Dickinson's original poetry performed better on the various stylistic elements evaluated compared to the generated pieces. Specifically, how understandable a poem is shows room for most improvement in future work. Future work embodies refining how well we are able to capture Dickinson's style in a generated poem. In our generated Hybrid Poetry, individuals were able at times to identify stylistic elements of both Frost and Dickinson but more often than not particular attributes were misidentified. Even though some participants identified Frost or Dickinson in generated works, there were some who also misidentified the author of original poetry. These findings illustrate some of the challenges in evaluating Hybrid Generated Poetry, offering intriguing challenges for future work. Future work on Hybrid Poetry also offers the opportunity of combining the styles of other poets, exploring the possibility of combining more divergent styles and exploring generative poetry that simultaneously reflects the style of several poets.

[3] As with many other technological innovations, the development of style imitation also has the potential for misuse. Exploration into potential misuse is left for future work.

References

[1] Gabriele Barbieri, Francois Pachet, Pierre Roy, and Mirko Degli Esposti. Markov constraints for generating lyrics with style. volume 242, 08 2012.

[2] Brendan Bena and Jugal Kalita. Introducing aspects of creativity in automatic poetry generation, 2020.

[3] Marie Chatfield. Markomposition a markov chain poetry generator.

[4] Amitava Das and Björn Gambäck. Poetic machine: Computational creativity for automatic poetry generation in bengali. In *ICCC*, 2014.

[5] Emily Dickinson. *Poems by Emily Dickinson, Three Series, Complete*. 2004.

[6] Eric Elshtain and Jon Trowbridge. Gnoetry, 2020. `https://beardofbees.com/index.html`.

[7] Emily Dickinson Museum. Major Characteristics of Emily Dickinson's Poetry. 2020. `https://www.emilydickinsonmuseum.org/emily-dickinson/poetry/tips-for-reading/major-characteristics-of-dickinsons-poetry/`.

[8] Ruth Finnegan. *Oral Literature in Africa*. Open Book Publishers, 2012.

[9] Marjan Ghazvininejad, Xing Shi, Yejin Choi, and Kevin Knight. Generating topical poetry. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1183–1191, Austin, Texas, November 2016. Association for Computational Linguistics.

[10] Marjan Ghazvininejad, Xing Shi, Jay Priyadarshi, and Kevin Knight. Hafez: an interactive poetry generation system. In *Proceedings of ACL 2017, System Demonstrations*, pages 43–48, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[11] Mika Hämäläinen and Khalid Alnajjar. Generating modern poetry automatically in Finnish. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5999–6004, Hong Kong, China, November 2019. Association for Computational Linguistics.

[12] Jing He, Ming Zhou, and Long Jiang. Generating chinese classical poems with statistical machine translation models. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, page 1650–1656. AAAI Press, 2012.

[13] Jack Hopkins and Douwe Kiela. Automatically generating rhythmic verse with neural networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 168–178, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[14] Jinyi Hu and Maosong Sun. Generating major types of chinese classical poetry in a uniformed framework, 2020.

[15] Long Jiang and Ming Zhou. Generating chinese couplets using a statistical mt approach. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, COLING '08, page 377–384, USA, 2008. Association for Computational Linguistics.

[16] Carolyn Lamb, Daniel G. Brown, and Charles L.A. Clarke. Human competence in creativity evaluation. 2015.

[17] Carolyn Lamb, Daniel G. Brown, and Charles L.A. Clarke. A taxonomy of generative poetry techniques. *Journal of Mathematics and the Arts*, 11(3):159–179, 2017.

[18] Irene Langkilde and Kevin Knight. The practical value of n'gran is in generation.

[19] Jey Han Lau, Trevor Cohn, Timothy Baldwin, Julian Brooke, and Adam Hammond. Deep-speare: A joint neural model of poetic language, meter and rhyme. *Association for Computational Linguistics*, 2018.

[20] Yi Liao, Yasheng Wang, Qun Liu, and Xin Jiang. Gpt-based generation for classical chinese poetry, 2019.

[21] Bei Liu, Jianlong Fu, Makoto P. Kato, and Masatoshi Yoshikawa. Beyond narrative description: Generating poetry from images by multi-adversarial training. *CoRR*, abs/1804.08473, 2018.

[22] Yusen Liu, Dayiheng Liu, Jiancheng Lv, and Yongsheng Sang. Generating chinese poetry from images via concrete and abstract information, 2020.

[23] Yubo Luo, Y. Huang, Fufang Li, and Chinchen Chang. Text steganography based on ci-poetry generation using markov chain model. *KSII Trans. Internet Inf. Syst.*, 10:4568–4584, 2016.

[24] Hugo Gonçalo Oliveira. Poetryme: a versatile platform for poetry generation. *Computational Creativity, Concept Invention, and General Intelligence*, 1:21, 2012.

[25] Hugo Gonçalo Oliveira, Raquel Hervás, Alberto Díaz, and Pablo Gervás. Adapting a generic platform for poetry generation to produce spanish poems. In *The International Conference on Computational Creativity (ICCC)*, pages 63–71, 2014.

[26] Hugo Gonçalo Oliveira, Tiago Mendes, Ana Boavida, Ai Nakamura, and Margareta Ackerman. Co-poetryme: interactive poetry generation. *Cognitive Systems Research*, 54:199–216, 2019.

[27] Hugo Gonçalo Oliveira. Automatic generation of poetry : an overview. 2009.

[28] openAI. Better language models and their implications, 2019. `https://openai.com/blog/better-language-models/`.

[29] Poetry Foundation. Emily dickinson. `https://www.poetryfoundation.org/poets/emily-dickinson`.

[30] Poetry Foundation. Robert frost. `https://www.poetryfoundation.org/poets/robert-frost`.

[31] Peter Potash, Alexey Romanov, and Anna Rumshisky. GhostWriter: Using an LSTM for automatic rap lyric generation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1919–1924, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[32] Project Gutenberg. Project gutenberg's poems: Three series, complete, by emily dickinson. `http://www.gutenberg.org/files/12242/12242-h/12242-h.htm`.

[33] Project Gutenberg. Selected poems by robert frost, 2021. `https://www.gutenberg.org/files/59824/59824-h/59824-h.htm`.

[34] A. Radford, Jeffrey Wu, R. Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[35] Scott Rettberg. Gnoetry, 2001. `https://elmcip.net/creative-work/gnoetry`.

[36] Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in bertology: What we know about how bert works, 2020.

[37] Juliana Shihadeh and Margareta Ackerman. Emily: An emily dickinson machine. In *The International Conference on Computational Creativity (ICCC)*, 2020.

[38] Aleksey Tikhonov and Ivan Yamshchikov. Sounds wilde. phonetically extended embeddings for author-stylized poetry generation. In *Proceedings of the Fifteenth Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 117–124, Brussels, Belgium, October 2018. Association for Computational Linguistics.

[39] J. Toivanen, Hannu Toivonen, A. Valitutti, and Oskar Gross. Corpus-based generation of content and form in poetry. In *ICCC*, 2012.

[40] Alsu Vakhitova. Poetry generation using markov chains.

[41] Alsu Vakhitova. Poetry generation using markov chains, 2012. `http://www.derczynski.com/innopolis/pdf/2019/vakhitova.pdf`.

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[43] R. Yan, Han Jiang, Mirella Lapata, S. Lin, X. Lv, and X. Li. i, poet: Automatic chinese poetry composition through a generative summarization framework under constrained optimization. In *IJCAI*, 2013.

[44] Xiaoyuan Yi, Ruoyu Li, and Maosong Sun. Generating chinese classical poems with RNN encoder-decoder. *CoRR*, abs/1604.01537, 2016.

[45] Xingxing Zhang and Mirella Lapata. Chinese poetry generation with recurrent neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 670–680, Doha, Qatar, October 2014. Association for Computational Linguistics.

[46] Xingxing Zhang and Mirella Lapata. Chinese poetry generation with recurrent neural networks. In *EMNLP*, 2014.

[47] Andrea Zugarini, Stefano Melacci, and Marco Maggini. Neural poetry: Learning to generate poems using syllables. *International Conference on Artificial Neural Networks*, 2019.

[48] T. M. Mills . C. Chan, C. T. Lenard. An introduction to markov chains. The Mathematical Association of Victoria for the forty-nineth annual conference, 2012.