Does ChatGPT Have a Poetic Style?

Melanie Walsh[1], Anna Preus[2] and Elizabeth Gronski[1]

[1]*Information School, University of Washington*
[2]*English Department, University of Washington*

Abstract

Generating poetry has become a popular application of LLMs, perhaps especially of OpenAI's widely-used chatbot ChatGPT. What kind of poet is ChatGPT? Does ChatGPT have its own poetic style? Can it successfully produce poems in *different* styles? To answer these questions, we prompt the GPT-3.5 and GPT-4 models to generate English-language poems in 24 different poetic forms and styles, about 40 different subjects, and in response to 3 different writing prompt templates. We then analyze the resulting 5.7k poems, comparing them to a sample of 3.7k poems from the Poetry Foundation and the Academy of American Poets. We find that the GPT models, especially GPT-4, can successfully produce poems in a range of both common and uncommon English-language forms in superficial yet noteworthy ways, such as by producing poems of appropriate lengths for sonnets (14 lines), villanelles (19 lines), and sestinas (39 lines). But the GPT models also exhibit their own distinct stylistic tendencies, both within and outside of these specific forms. Our results show that GPT poetry is much more constrained and uniform than human poetry, showing a strong penchant for rhyme, quatrains (4-line stanzas), iambic meter, first-person plural perspectives (we, us, our), and specific vocabulary like "heart," "embrace," "echo," and "whisper."

Keywords

LLMs, ChatGPT, poetry, style, form, text generation

1. Introduction

Generating poetry has become a popular application of LLMs, perhaps especially of OpenAI's widely-used ChatGPT. The chatbot's ability to produce poetry was frequently mentioned in early journalistic reports [36], and the question of whether it produced "good" or "bad" poetry quickly became the subject of hot debate [12, 5]. While these debates point to the significance of LLM-generated poetry as a topic, in this paper we are less interested in subjectively evaluating the quality of ChatGPT-generated poems and more interested in quantifying and describing key patterns in the poetry produced. What kind of poet is ChatGPT? Does ChatGPT have its own poetic style? Can it successfully produce poems in different forms and styles, like sonnets (14 lines with regular rhyme and iambic pentameter) or villanelles (19 lines with alternating first and third lines)?

In this paper, we present initial results from a larger study of LLM-generated poetry. We prompt the GPT-3.5 and GPT-4 models to generate English-language poems in 24 different poetic forms or styles, about 40 different subjects, and in response to 3 different templates.

CHR 2024: Computational Humanities Research Conference, December 4–6, 2024, Aarhus, Denmark
*Corresponding author.
melwalsh@uw.edu (M. Walsh); apreus@uw.edu (A. Preus)
0000-0003-4558-3310 (M. Walsh)

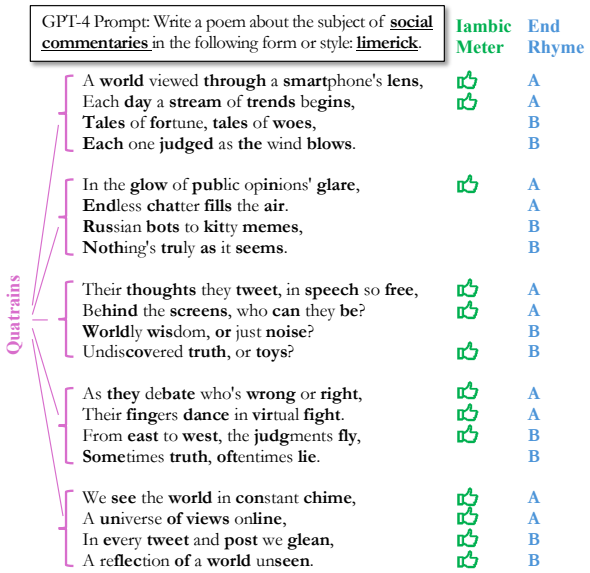

Figure 1: "Write a poem about the subject of *social commentaries* in the following form or style: *limerick*." An example poetry generation prompt and response by GPT-4, representing common tendencies of the model. While GPT-4 presents a comedic, topical take on a social commentary, it does not produce a typical *limerick* (usually 5 lines, anapestic meter, AABBA rhyme) but instead produces five quatrains with mostly iambic meter and AABB rhyme, what we suggest is its "default" mode.

We then analyze the resulting 5.7k poems, comparing them to a sample of 3.7k poems from the Poetry Foundation and the Academy of American Poets. We find that the GPT models, especially GPT-4, can successfully produce poems in a range of both common and uncommon English-language forms in superficial yet noteworthy ways, such as by producing poems of appropriate lengths for sonnets (14 lines), villanelles (19 lines), and sestinas (39 lines). But we find that the GPT models also exhibit their own distinct stylistic tendencies, both within and outside of these specific forms.

Our results show that the poetry produced by GPT-3.5 and GPT-4 is much more constrained and uniform than human poetry. Unless otherwise prompted (and sometimes when otherwise prompted), both GPT models have a tendency to produce rhymed lines in something like iambic meter—a regular pattern of unstressed and stressed syllables that characterizes the majority of English-language verse before the 20th century [29]. Both models also have a tendency to organize poetic lines into quatrains (4-line stanzas). And they display other distinct signatures, such as a curiously dominant first-person plural perspective and a penchant for words like "heart," "embrace," "echoes," and "whispers." We release our code,[1] and we plan to share our ChatGPT poetry corpus at a later stage of the project, with the aim of encouraging further analysis through computational and/or more traditional literary studies approaches.

[1]https://github.com/melaniewalsh/chatgpt_poetry

2. Related Work

The history of computational poetry generation dates back to at least the mid-20th century [15]. Poets, researchers, and hobbyists have experimented with a range of technical approaches [9], from rule-based systems [19, 8], to Markov chains [2], to most recently neural networks and LLMs [34, 35, 18, 13, 1, 3, 25].

While computational poetry has been an active area of inquiry for many decades [20, 30, 10, 7, 27, 21], it has arguably remained the purview of specialists until recently. But since the release of ChatGPT in November 2022, hundreds of millions of people have used and experimented with LLMs, opening up computational poetry generation to a broader public.

Training data and memorization are key considerations for LLM-generated poetry from ethical, legal, and technical perspectives. Models' ability to produce poetry is intimately tied to their training data, which partly consists of literary works by both living and dead writers. Much popular and scholarly attention related to LLMs and literature has focused—rightfully—on the ethics and legality of such literary training data [28, 17, 31, 32]. We believe these concerns are vitally important to examinations of ChatGPT's style, which is built from the words of other writers. We also think that it is valuable to ask questions about LLMs' poetic capacities because it can help inform debates about LLMs and creativity while also advancing our understanding of how poetry is being used and propagated in the contemporary world.

In their work on poetry memorization in ChatGPT specifically, D'Souza and Mimno [6] show that the most likely factor for a poem's memorization by the model was its inclusion in the 1983 *Norton Anthology of Literature*. This finding suggests that canonical poetry is disproportionately represented in the GPT models, which could influence the kind of poetry they produce. In a similar vein, we show in prior work [33] that 41% of a curated sample of poems from the Poetry Foundation and the Academy of American Poets (which we also use as a comparison corpus in this study) are likely memorized by GPT-4. Our previous analysis suggests that this memorization may enhance the models' ability to classify the form of the poems, but the results are not conclusive. More work is needed to evaluate the impact that memorization may have on poetry generation.

3. Data + Methods

3.1. Human Poetry Corpus

To provide a baseline comparison for our ChatGPT-generated poetry and to guide our prompting, we curate a dataset of poems, styles, and subjects from the Poetry Foundation and the Academy of American Poets. Both organizations are well-respected poetry institutions with websites that host tens of thousands of poems spanning hundreds of years, and many of the poems are tagged by style and subject on the websites.

We scrape up to 400 poems from these two sources for 23 different poetic forms or styles, which we also use as prompts for our ChatGPT-generated poetry corpus. Following prior work [33], we select poems in the following categories: FIXED FORMS, UNFIXED FORMS, and FORMAL ELEMENTS (which consists of both meters and stanza forms) (see Table 1). In total, the sample includes 3,874 poem/style pairs, or 3,692 unique poems.

Table 1
The distribution of poems by form and source.

Poetic Form x Source	Poetry Foundation & Academy of American Poets	GPT-3.5	GPT-4
Fixed Forms			
Ballad	110	120	120
Ghazal	40	120	120
Haiku	50	120	120
Limerick	7	120	120
Pantoum	25	120	120
Sestina	41	120	120
Sonnet	856	120	120
Villanelle	63	120	120
Formal Elements			
Meters			
Blank Verse	209	120	120
Free Verse	387	120	120
Common Measure	112	120	120
Stanza Forms			
Couplet	398	120	120
Quatrain	89	120	120
Tercet	94	120	120
Unfixed Forms			
Ars Poetica	94	120	120
Aubade	16	120	120
Concrete Poetry	24	120	120
Dramatic Monologue	191	120	120
Ekphrasis	145	120	120
Elegy	254	120	120
Ode	119	120	120
Pastoral	75	120	120
Prose Poem	475	120	120
"A Poem"	-	120	120
Total	**3,874** poem/form pairs	**2,880** poems	**2,880** poems

We manually remove prefatory text—such as dedications, dates, epigraphs, or other contextual information—from human-authored poems with traditionally fixed lengths (e.g., sonnets, villanelles, sestinas) if the poem is within 10 lines of the conventional length. We do not remove prefatory material from other poems; however, based on our qualitative analysis and review, we do not believe prefatory material is extensive in most other poems or significantly impacts results.

While the Poetry Foundation and the Academy of American Poets are among the largest tagged poetry collections available, they are also defined by various kinds of bias that are important to note. They both focus on English-language poetry, and the Academy of American Poets focuses especially on American poetry. Not all of the poems in their collections are tagged, and it is unclear why some poems are tagged and others are not. Neither site hosts a

Table 2
Subjects, styles, and writing prompt templates for the GPT-generated poetry corpus.

Category	
Subjects	**General:** activities, arts & sciences, living, love, mythology & folklore, nature, religion, relationships, social commentaries **Occasions:** anniversary, birth, birthdays, engagement, farewells, funerals, recovery, graduation, gratitude, toasts, weddings **Holidays:** cinco de mayo, christmas, easter, father's day, halloween, hanukkah, independence day, kwanzaa, memorial day, mother's day, new year, passover, ramadan, thanksgiving, yom kippur
Styles	**Fixed:** limerick, pantoum, ghazal, ballad, villanelle, sonnet, sestina, haiku **Unfixed:** epic, monologue, ars poetica, aubade, pastoral, ode, elegy, visual poetry, ekphrasis, prose poem **Formal Elements:** *meters:* common measure, blank verse, free verse *stanza forms:* quatrain, tercet, couplet
Prompt Templates	**General:** Write a poem about the subject of X in the following form or style: Y. **Figurative:** Write a poem about the subject of X in the following form or style: Y. Do not use the actual word(s) X or Y in the poem. **Specific:** Write a poem about the subject of X in the following form or style: Y. Make the poem about something specific.

representative collection of poems, in terms of poets' gender, race, sexuality, and time period (it is also difficult to know what a representative collection would be). They also over-represent prestigious and canonical poetry, which may be of particular note in comparison with ChatGPT since the model may be trained on (and perhaps even encouraged to produce) more popular, commercial, and colloquial poetry.

3.2. ChatGPT-Generated Poetry Corpus

To create our ChatGPT poetry corpus, we prompt **GPT-3.5 Turbo** and **GPT-4** [24] to generate poems in response to 3 different *writing prompt templates*, in 24 different *styles/forms*, and about 40 different *subjects*. The styles and subjects are selected from the tagging schema on the Poetry Foundation's website. We use zero-shot prompts (i.e., prompts that do not provide desired example outputs) because we are interested in testing the model's "out-of-the-box" capabilities in a mostly unmediated form.

We select styles and subjects from the Poetry Foundation because they offer an extensive and diverse poetic taxonomy that is developed by an authoritative external source and that is reflective of one of the largest existing collections of human poetry. For our "subjects," we select the 40 broadest level "topics" from the Poetry Foundation's tagging schema, which include the subcategories "subjects," "occasions," and "holidays" (see Table 2). For our "styles," we select

the 23 styles and poetic forms described in Section 3.1. We add the style of "a poem" because we are interested in the models' responses to the generic idea of a poem without a specified form. These combinations result in 2,880 generated poems per model, with 120 poems per style (per model) and 72 poems per subject (per model) (see Table 1).

We model the construction of our 3 *writing prompt templates* on popular approaches demonstrated on social media, in journalistic articles, and by LLM companies [14, 12, 22]:

1. **General:** Write a poem about the subject of X in the following form or style: Y.

2. **Figurative:** Write a poem about the subject of X in the following form or style: Y. Do not use the actual word(s) X or Y in the poem.

3. **Specific:** Write a poem about the subject of X in the following form or style: Y. Make the poem about something specific.

We include our "figurative" and "specific" templates after observing the models' tendency to repeat the words in the prompts and to be vague. These templates push the model to create more diverse outputs. We believe that prompting significantly impacts the kind of poetry that the GPT models produce, and we reflect on this more in Section 5.

4. Results

4.1. Poetic Length & Structure

We measure the number of lines and the number and kind of stanzas across all the poems by parsing line breaks. We visualize these distributions as boxplots (Figure 2) and heatmaps (Figure 3), revealing the most common lengths and shapes of the poems across styles and forms. These results show that when we prompt the models to generate poems in forms with typically fixed lengths—such as *sonnets* (14 lines)—they largely adhere to this convention, with notable improvement in GPT-4. While GPT-3.5 and GPT-4 both generate sonnets with a median average length of 14 lines, Figures 2 and 3 show that there is much more variability in GPT-3.5. As displayed in the boxplot, the upper 75% quartile extends to 32 lines, and the range extends to 55 lines. By contrast, the entire range of GPT-4 sonnets (minus outliers) falls at exactly 14 lines. Line lengths for sestinas (typically 39 lines) and villanelles (typically 19 lines) follow a similar pattern. The median lengths are appropriately 39 and 19 lines for both models, but GPT-4 demonstrates much more consistency, displaying a smaller interquartile range and spread of outliers (the same consistency is also displayed in the heatmap in Figure 3).

Interestingly, for these three forms, GPT-4 hues closer to "conventional" lengths than our sample of poems from the Poetry Foundation and the Academy of American Poets. In a small percentage of these human poems, we find that the longer lengths come from explicit or implicit play with or defiance of the forms. For example, Bino A. Realuyo concludes his 15-line poem, "Euler's Equation," with the line: "a rebellion, the fifteenth line of a sonnet." More often, in these longer poems, authors include a given form in multiples, such as Algernon Charles Swinburne's "double sestina" (12 stanzas of 12 lines each) in "The Complaint of Lisa."

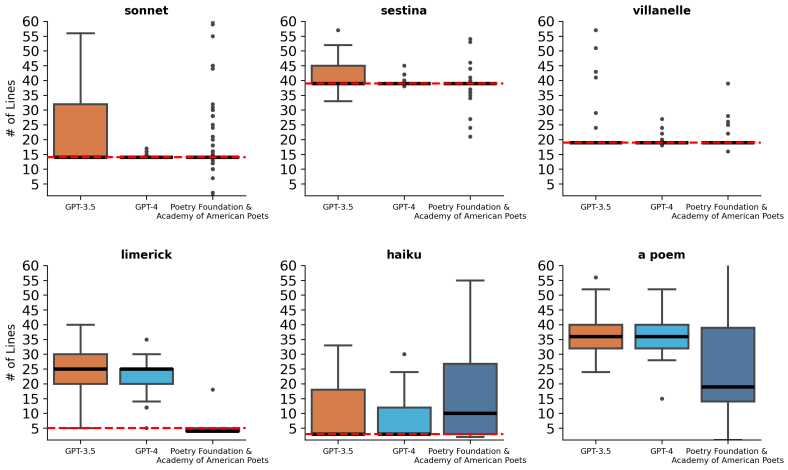

Figure 2: These boxplots represent the distribution of line lengths for poems with conventionally fixed lengths produced by **GPT-3.5**, **GPT-4**, and authors from the **Poetry Foundation and the Academy of American Poets**. The GPT models were also prompted with the generic style of "a poem"; to provide a comparison for the human poems, we include an aggregation of all poems from the sample. The boxes show the "interquartile range" (25% quartile-75% quartile) with a thicker line indicating the median average; the whiskers extend beyond the boxes by 1.5 times the IQR; the outliers are values that fall beyond the whiskers. The dotted red line indicates the expected number of lines for each form, e.g., a sonnet typically has 14 lines.

An obvious aberration for the GPT models is their atypically long limerick style. Where a traditional limerick is usually about 5 lines long, the median length for both GPT models is 25 lines. Upon closer inspection, it is clear that both models frequently bundle multiple, appropriately-lengthed limericks together. The heatmap in Figure 3 shows that the models often produce several limericks in a row. This is also the case for GPT-3.5's atypically long sonnets, which are usually multiple sonnets packed into one. While these multiples resemble the long poems that we observe in our human poetry sample, we think this tendency more likely suggests that, in certain cases, the GPT models know *how* to produce a particular kind of poem but don't know *when* to stop.

Table 3
Quatrains. Percentage of poems with at least one quatrain and percentage of quatrains of all stanzas.

Source	Poems with Quatrain	Stanzas with Quatrain
Poetry Foundation and Academy of American Poets	713 / 3,874 poems (**18.4%**)	3,014 / 18,052 stanzas (**16.7%**)
GPT-3.5 Turbo	2,027 / 2,880 poems (**70.4%**)	16,089 / 24,093 stanzas (**66.8%**)
GPT-4	1,824 / 2,880 poems (**63.3%**)	13,303 / 22,305 stanzas (**59.6%**)

Aside from limericks, the GPT models can broadly produce poems of appropriately diverse lengths for a range of fixed forms, but they demonstrate a penchant for producing poems of an almost "default" size when left to their own devices. For both models, the median average length for a generic "poem" is 36 lines (see Figure 2), and the overall median length across all

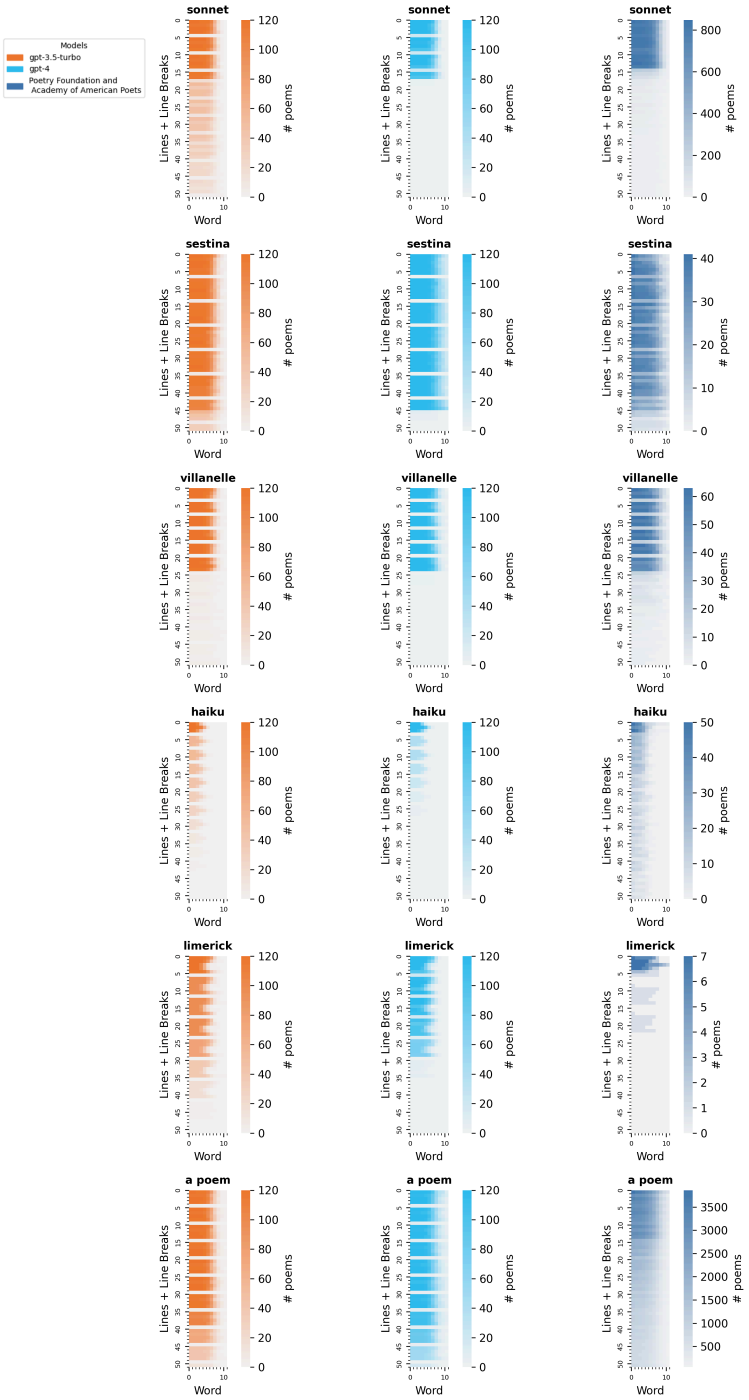

Figure 3: These heatmaps represent the distribution of words, lines, and line breaks for *fixed form* poems by **GPT-3.5**, **GPT-4**, and authors from the **Poetry Foundation and the Academy of American Poets**. Darker squares represent a higher concentration of words and lines in specific positions across the poems; lighter squares represent a higher concentration of white space and line breaks. The GPT models are also prompted with the generic style of "a poem"; to provide a comparison for the human poems, we include an aggregation of all poems from the sample.

styles is 32 lines.

Another striking feature of the GPT poems is the dominance of 4-line stanzas, or *quatrains*. We find that while just 16.7% of the human-authored stanzas are quatrains, a whopping 66.8% of all GPT-3.5 stanzas and 59.6% of all GPT-4 stanzas are quatrains (Table 3). The heatmaps in Figure 4 visually demonstrate how common quatrains are across the GPT-generated poems, showing clear line breaks in regular 4-line intervals, with no such regularity evident in the human poems.

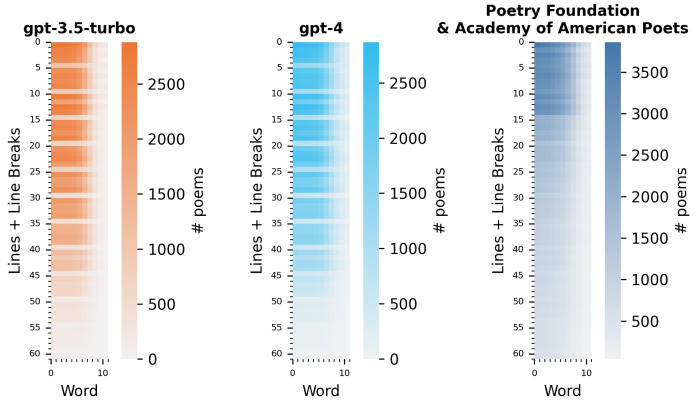

Figure 4: These heatmaps represent the distribution of words, lines, and line breaks for all poems by **GPT-3.5**, **GPT-4**, and authors from the **Poetry Foundation and the Academy of American Poets**. Darker squares represent a higher concentration of words and lines in specific positions across the poems; lighter squares represent a higher concentration of white space and line breaks. The unusual dominance of quatrains (line breaks after 4 consecutive lines) is evident in the GPT-generated poems.

4.2. Collective Perspective

We measure the normalized frequency of pronouns (Table 4) in each corpus, expressed per 100 words. We find that poems produced by GPT-3.5 and GPT-4 tend to use more first-person plural pronouns ("we," "us," "our") and fewer first-person singular pronouns ("i", "me," "myself") than poems written by humans (see Figure 5). For example, GPT-4 produced the following limerick about Memorial Day in response to our *figurative prompt* (which specifies not to include the style or subject words in the poem):

> *In May **we** stand strong, hearts ablaze,*
> *For those who've seen war's smoky haze.*
> ***We** honor the brave,*
> *Who life for **us** gave,*
> *In silence, **we** give them **our** praise.*

This limerick continues on for 20 more lines in 5-line, rhyming (AABBA) stanzas, consistent with the results presented in Section 4.1.

Because there are a large number of "holiday" and "occasion" subject prompts like "Memorial Day," which perhaps encourage meditation on collective experiences, we also show normalized frequency for the GPT-generated poems with these subjects removed (see the dotted lines in Figure 5). Without these subjects, the normalized frequency for the first-person plural decreases slightly, and it increases slightly for the third-person. But the curious dominance of the first-person plural is still present. We think this pattern may reflect the models' preprogrammed attitudes toward inclusivity, as well as its obvious lack of first-person singular experiences, but more work is needed to explore this trend further.

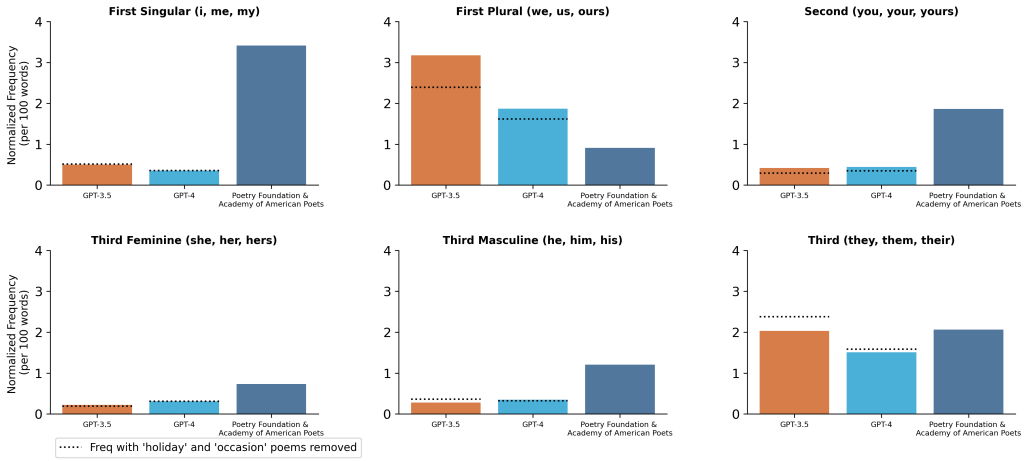

Figure 5: The normalized frequency of pronouns used in poems by **GPT-3.5**, **GPT-4**, and authors from the **Poetry Foundation and the Academy of American Poets**, expressed per 100 words. The dotted line indicates normalized frequency in the GPT poems with the "holiday" and "occasion" poems removed (showing that first-person plural in the GPT-generated poems decreases slightly, and third-person increases slightly).

Table 4
Pronouns by Category

Category	Pronouns
First Singular	i, me, my, mine, myself
First Plural	we, us, our, ours, ourselves
Second	you, your, yours, yourself, yourselves, thou, thee, thy, thine, thyself
Third Feminine	she, her, hers, herself
Third Masculine	he, his, him, himself
Third	they, them, their, theirs, themself, themselves, it, its, itself

4.3. Most Distinctive Words

We also analyze the most distinctive opening words and overall words across the poems using Monroe, Colaresi, and Quinn [23]'s "fightin' words" algorithm, which uses weighted log-odds

ratios with an informative Dirichlet prior. This method is designed to robustly compare word usage across unevenly distributed text corpora. We specifically use an implementation by Hessel [11][2] and restrict the vocabulary to words that appear in a minimum of 10 poems. We remove stopwords for our overall word analysis but not for our first word analysis. In Figures 6 and 7, we display words with the highest Z-scores for each category, representing the most distinctive words.

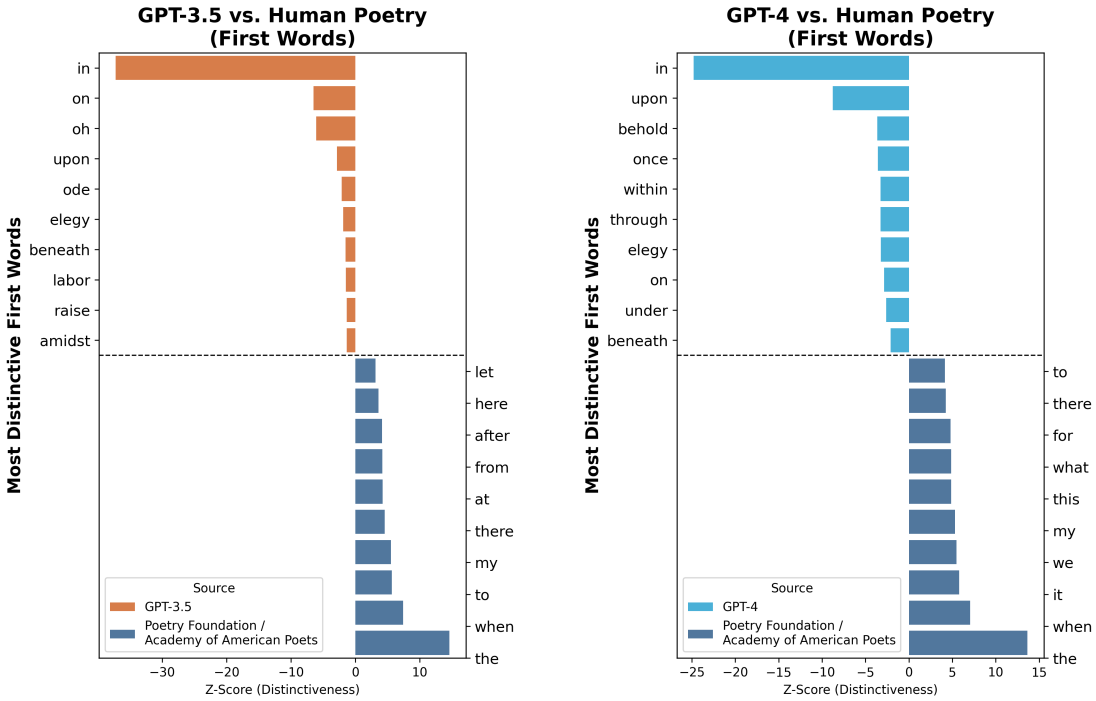

Figure 6: The 10 most distinctive first words in poems produced by GPT-3.5, GPT-4, or the Poetry Foundation and the Academy of American Poets. To identify these words, we use Monroe, Colaresi, and Quinn [23]'s algorithm for comparing language use across text corpora. Stopwords are not removed.

The most distinctive opening word in both the GPT-3.5- and GPT-4 generated poems is "In" (see Figure 6). This preposition is included across a wide range of poems and contexts, such as:

> ***In*** *autumn's blaze of golden hue... (GPT-3.5 — pantoum, Thanksgiving)*
> ***In*** *the darkest days, a flicker of light...(GPT-3.5 — ars poetica, Hanukkah)*
> ***In*** *the girth of world-kaleidoscope, we are birthed into living, (GPT-4 — free verse, living)*

The next most distinctive first word in GPT-4 poems is "Upon," which also seems to be a frequent way for the model to initiate iambic meter:

[2]ttps://github.com/jmhessel/FightingWords

***Upon** a stage where shadows nightly reign... (GPT-4 — sonnet, Halloween)*
***Upon** this day, we sing the laborer's song,... (GPT-4 — sonnet, Labor Day)*
***Upon** the chill of winter's breath descends,... (GPT-4 — blank verse, Hanukkah)*

The word "upon" is an iamb (the basic unit of iambic meter), meaning it consists of an unstressed syllable followed by a stressed syllable. This is also the case for other distinctive GPT-4 first words, such as "beneath," "behold," and "within." First words in poems from the Poetry Foundation and Academy of American Poets show no such distinctive patterns, mostly consisting of articles and pronouns.

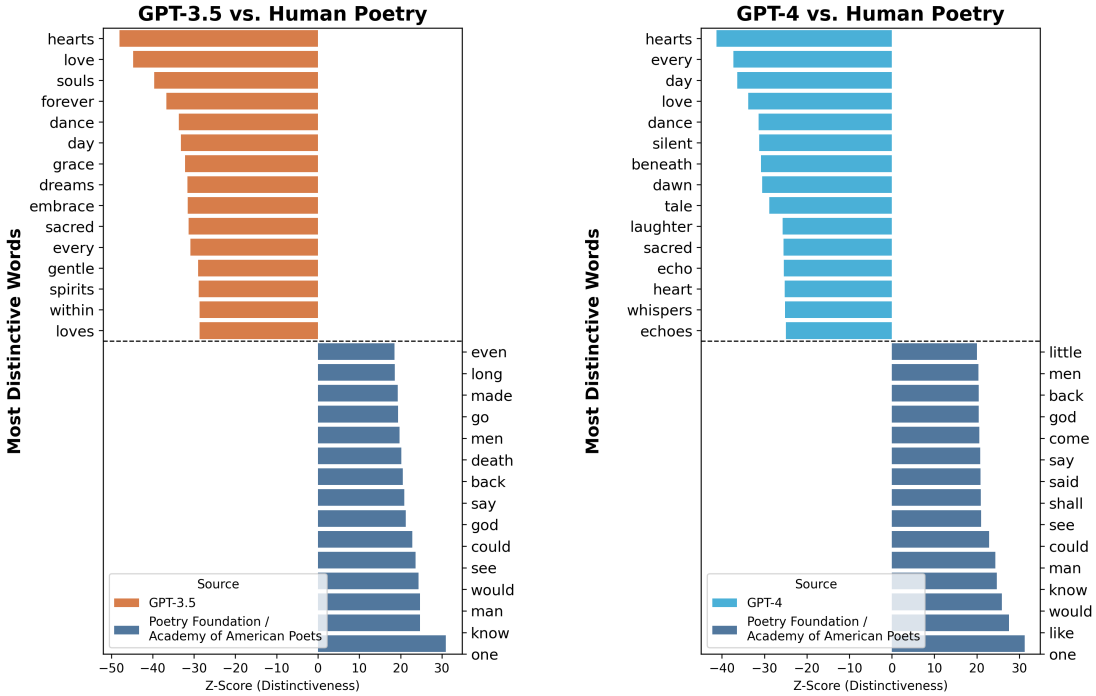

Figure 7: The 15 most distinctive words in poems produced by **GPT-3.5** and **GPT-4** vs. poems found in the **Poetry Foundation and the Academy of American Poets**. To identify these words, we use Monroe, Colaresi, and Quinn [23]'s algorithm for comparing language use across text corpora. In this case, stopwords are removed.

Overall, the distinguishing vocabulary for the GPT models consists of words associated with love ("heart," "love," "souls,"), words that rhyme ("grace", "embrace"), and words that are acoustic ("echo," "whisper"). For GPT-3.5, words like "embrace," "grace," "dance," and "dreams" are touchstones. At least one of these words shows up in 87% of the GPT-3.5 poems. For GPT-4, either "echo" or "whisper" shows up in 75% of the poems. For example, both words appear in this *dramatic monologue* about the *arts & sciences*:

Look upon me, ageless I stand, the crossing of arts and sciences,

*In the **echoing** hallways of knowledge, beneath glimmering frescoed edifices.*
*I am the **whisperer** in marbled alcoves, the scribe of thinkers' existence,*
Caught in endless dialogue, between creativity and discipline's persistence.
-GPT-4 (dramatic monologue, arts & sciences)

4.4. Prosody Analysis (Rhyme & Meter)

"Prosody" refers to patterns of sound in poetry, encompassing rhyme and meter. Analyzing prosody across a large corpus poses challenges even in human-authored poems because it relies on the pronunciation of particular words in relation to each other. The text of a poem does not provide direct access to its prosody because the same word may have different pronunciations in different forms of English, and even with the same pronunciation a word might be stressed or unstressed depending on its context. For example, in Alfred, Lord Tennyson's poem "Ulysses," the word "I" is unstressed at the beginning of a line and stressed toward the end: "I cannot rest from travel: I will drink." Analyzing prosody across GPT-generated poems poses additional challenges because unlike human-authored poems, which are often either clearly free verse or clearly aimed at embodying a particular metrical pattern, the meter of GPT outputs can be less precise and harder to define with a single metrical label.

Table 5
Rhyme Usage. Percentage of poems with rhyme and average percentage of rhymed lines, based on quantitative analysis with the CMU Pronouncing Dictionary. Rhymed lines include AA, ABAB, ABBA, and ABCB rhymes.

Source	Poems with at least One Rhyme	Avg. Percent Rhymed Lines
Poetry Foundation and Academy of American Poets	2,518 / 3,874 poems (**65.0%**)	29.45%
GPT-3.5 Turbo	2,599 / 2,880 poems (**90.2%**)	63.87%
GPT-4	2,578 / 2,880 poems (**89.5%**)	65.20%

To measure prosody, we thus conduct both a quantitative *and* qualitative analysis. For our manual analysis, we take a random sample of poems in each form and, drawing on our domain expertise, hand-annotate various prosodic elements where they are discernible, including dominant meter, line-length (in terms of poetic feet—tetrameter, pentameter, etc.), rhyme scheme, and stanza patterns. We evaluate 144 poems produced by GPT-3.5 (6 in each form) and 144 poems produced by GPT-4 (6 in each form) for just over 5% of the GPT-generated corpus. We also analyze 138 poems from the human-authored corpus (6 poems in 23 forms), making up just over 3.7% of the human-authored corpus.

Over 80% of the GPT-generated poems in our random sample contain patterns of end rhyme, as compared with around 50% of the human-authored poems. Over 60% of the GPT-authored poems had a dominant iambic meter, compared to just under 40% of poems from the human corpus. When we break these results down between the GPT-3.5 and GPT 4 models, it appears that the dominance of iambic meter is lessening somewhat in the newer model. Only around 53% of GPT-4-authored poems had a dominant iambic meter compared to almost 74% of GPT-3.5-authored poems. Rather than indicating a shift in the model's default tendencies in relation to poetry, we think this change may reflect GPT-4's increased ability to *not* produce iambic

meter when it is prompted to produce poems in forms that do not traditionally include regular meter—for example haiku, prose poetry, or free verse.

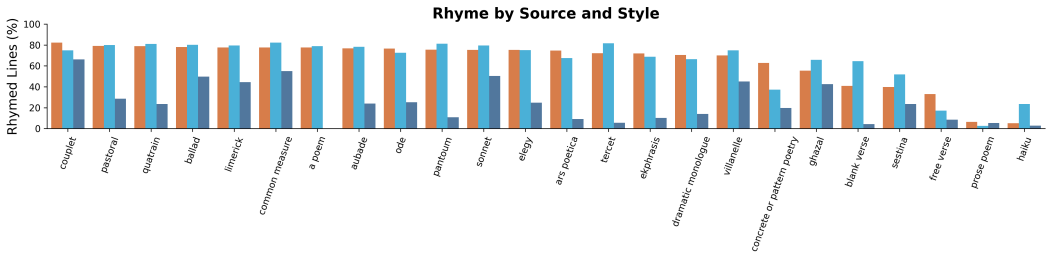

Figure 8: These bar plots show percentage of lines rhymed in poems by GPT-3.5, GPT-4, and authors from the Poetry Foundation and the Academy of American Poets. Rhymes were calculated with the CMU Pronouncing Dictionary and include AA, ABAB, ABBA, and ABCB rhymes. Styles are sorted from most to least rhyme based on poems by the GPT-3.5 model.

We also conduct a quantitative analysis of rhyme across all poems using Parrish [26]'s Python implementation of the CMU Pronouncing Dictionary.[3] Using this dictionary, we measure how many lines end with AA, ABAB, ABBA, or ABCB rhymes. Around 90% of the poems generated by both the GPT-3.5 and GPT-4 models contained at least one end rhyme, compared to around 65% of our human-authored corpus. Moreover, for certain forms like *sonnets* and *ballads*, the models *always* produced poems with at least one end rhyme. This was also the case for unfixed forms like *aubade* or *pastoral*, which do not typically require specific patterns of rhyme. The percentage of rhyming lines for all styles and sources is displayed in Figure 8.

Taken together, our analyses suggest that there is a sort of default poetic mode in GPT models, which favors quatrains, iambic meter, and end rhyme. The models can be prompted to produce writing in other styles, but sometimes the persistent iambic/quatrain/end rhyme style still breaks through. For example, although prompted to produce a *social commentary*-related *limerick*, which would typically be a five-line stanza with anapestic meter and AABBA rhyme, the GPT-4 model produced the following five quatrains with iambic meter and AABB rhyme:

*A world viewed through a smartphone's **lens**,*
*Each day a stream of trends **begins**,*
*Tales of fortune, tales of **woes**,*
*Each one judged as the wind **blows**.*

*In the glow of public opinions' **glare**,*
*Endless chatter fills the **air**.*
*Russian bots to kitty **memes**,*
*Nothing's truly as it **seems**.*

*Their thoughts they tweet, in speech so **free**,*
*Behind the screens, who can they **be?***

*Worldly wisdom, or just **noise?***
*Undiscovered truth, or **toys?***

*As they debate who's wrong or **right,***
*Their fingers dance in virtual **fight.***
*From east to west, the judgments **fly,***
*Sometimes truth, oftentimes **lie.***

*We see the world in constant **chime,***
*A universe of views **online,***
*In every tweet and post we **glean,***
*A reflection of a world **unseen.***

While this poem is a funny and topical take on a social commentary ("Russian bots to kitty memes / Nothing's truly as it seems"), it also highlights some of the model's limitations. Even

[3] https://github.com/aparrish/pronouncingpy

when instructed to produce more specific or varied forms, the model's default poetic mode—quatrains, iambic meter, and end rhyme—often resurfaces, favoring structured patterns over more diverse styles.

5. Discussion

By showing that GPT-generated poetry contains clear stylistic quirks and characteristics, we build upon emerging research on the style of LLM-generated texts. For example, after analyzing millions of biomedical article abstracts, Kobak, González-Márquez, Horvát, and Lause [16] show that "hundreds of words have abruptly increased their frequency after ChatGPT became available." They reveal particular spikes in the use of "style-affecting verbs and adjectives that ChatGPT-like LLMs prefer," such as "delve," "significant," and "crucial." Although Kobak, González-Márquez, Horvát, and Lause [16] are primarily interested in LLMs' impact on academic research, their findings contribute to growing knowledge about LLM style in specific genres. Other related research in this area has focused on analyzing bias and stereotypes present in LLM-generated texts [4]. We believe there are exciting opportunities for digital humanities scholars and language experts to study more of the artistic dimensions of LLM-generated texts.

The results of our poetry prompting experiments highlight both advancements and notable limitations with GPT-generated poetry. The models' ability to produce poems of appropriate lengths for a wide variety of forms and styles—without any fine-tuning—marks a significant development in automatic poetry generation, especially since the models manage to do so while incorporating rhyme and meter and maintaining general clarity. Yet, overall, the models also exhibit far less variation, diversity, and creativity than the human-authored poems. However, we want to flag that *prompting*—what the user asks the model to generate and how that ask is constructed—plays a major role in shaping the poetry that the models produce. In related experiments, when we prompted the models with specific author names, our results seemed to shift and become more complex. In this specific study, our goal was not to produce the most creative and interesting poetry possible, but rather to understand the broad contours of the models and their outputs. If we wanted to produce more interesting poetry, we would likely use different prompts.

6. Conclusion

We prompt the GPT-3.5 and GPT-4 models to generate English-language poems in 24 different poetic forms or styles, about 40 different subjects, and in response to 3 different templates. We compare these GPT-generated poems to a sample of poems from the Poetry Foundation and the Academy of American Poets, showing that the GPT models are much more formulaic and constrained than the human-authored poetry. We argue that the GPT models have a "default" poetic mode, characterized by quatrains with rhymed lines in iambic meter; first-person plural perspectives; and the repetition of words like "heart," "embrace," "echoes," and "whispers." This default mode sometimes breaks through even when otherwise prompted. We share the code

that we used to conduct this analysis,[4] and we share all the public domain human-authored poems and form/style annotations from prior work.[5] We plan to share our ChatGPT poetry corpus at a later stage of the project. In future work, we plan to explore a wider range of prompts (potentially including author names) and models, and to study the poems more closely with traditional literary studies approaches.

Acknowledgments

We would like to thank Zoe LeBlanc, John Ladd, Matt Lavin, and Gabi Kirilloff for early feedback on this project. This research was partly supported by the NEH-funded "AI for Humanists" project and by an "AI, Creativity, and Humanities" collaboration grant from the Simpson Center for the Humanities at the University of Washington.

References

[1] R. Agarwal and K. Kann. "Acrostic Poem Generation". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by B. Webber, T. Cohn, Y. He, and Y. Liu. Online: Association for Computational Linguistics, 2020, pp. 1230–1240. DOI: 10.18653/v1/2020.emnlp-main.94. URL: https://aclanthology.org/2020.emnlp-main.94.

[2] A. Astigarraga, J. M. Martínez-Otzeta, I. Rodriguez, B. Sierra, and E. Lazkano. "Markov Text Generator for Basque Poetry". In: *Text, Speech, and Dialogue*. Ed. by K. Ekštein and V. Matoušek. Cham: Springer International Publishing, 2017, pp. 228–236. DOI: 10.1007/978-3-319-64206-2_26.

[3] T. Chakrabarty, V. Padmakumar, and H. He. "Help me write a Poem - Instruction Tuning as a Vehicle for Collaborative Poetry Writing". In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Ed. by Y. Goldberg, Z. Kozareva, and Y. Zhang. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, 2022, pp. 6848–6863. DOI: 10.18653/v1/2022.emnlp-main.460. URL: https://aclanthology.org/2022.emnlp-main.460.

[4] M. Cheng, E. Durmus, and D. Jurafsky. *Marked Personas: Using Natural Language Prompts to Measure Stereotypes in Language Models*. 2023. DOI: 10.48550/arXiv.2305.18189. URL: http://arxiv.org/abs/2305.18189.

[5] L. Clarke. *ChatGPT Is Pretty Bad At Poetry, According To Poets*. 2023. URL: https://www.vice.com/en/article/7kx9d9/chatgpt-is-pretty-bad-at-poetry-according-to-poets.

[6] L. D'Souza and D. Mimno. "The Chatbot and the Canon: Poetry Memorization in LLMs". In: *Computational Humanities Research*. 2023. URL: https://ceur-ws.org/Vol-3558/paper5712.pdf.

[4] https://github.com/melaniewalsh/chatgpt_poetry
[5] https://github.com/maria-antoniak/poetry-eval

[7] A. Das and B. Gambäck. "Poetic Machine: Computational Creativity for Automatic Poetry Generation in Bengali". In: 2014. URL: https://www.semanticscholar.org/paper/Poetic-Machine%5C%3A-Computational-Creativity-for-Poetry-Das-Gamb%5C%C3%5C%A4ck/40c52a0ad0322ee0e02105d578d561c35edbb5e2.

[8] P. Gervás. "An expert system for the composition of formal Spanish poetry". In: *Knowledge-Based Systems* 14.3 (2001), pp. 181–188. DOI: 10.1016/s0950-7051(01)00095-8. URL: https://www.sciencedirect.com/science/article/pii/S0950705101000958.

[9] H. Gonçalo Oliveira. "Automatic generation of poetry: an overview". In: (2009).

[10] H. Gonçalo Oliveira, A. Cardoso, and F. Pereira. "Tra-la-Lyrics: An approach to generate text based on rhythm". In: 2007, pp. 47–55.

[11] J. Hessel. *jmhessel/FightingWords*. 2024. URL: https://github.com/jmhessel/FightingWords.

[12] W. Hunter. *What Poets Know That ChatGPT Doesn't*. 2023. URL: https://www.theatlantic.com/books/archive/2023/02/chatgpt-ai-technology-writing-poetry/673035/.

[13] H. Jhamtani, S. V. Mehta, J. Carbonell, and T. Berg-Kirkpatrick. "Learning Rhyming Constraints using Structured Adversaries". 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)*. Ed. by K. Inui, J. Jiang, V. Ng, and X. Wan. Hong Kong, China: Association for Computational Linguistics, 2019, pp. 6025–6031. DOI: 10.18653/v1/D19-1621. URL: https://aclanthology.org/D19-1621.

[14] Jim Al-Khalili [jimalkhalili]. *I just asked ChatGPT to write me a poem on the two-slit experiment in quantum mechanics in the style of Robert Burns. My work on this earth is done. https://t.co/e4dTTodT62*. Tweet. 2023. URL: https://x.com/jimalkhalili/status/1621454981097209857.

[15] J. Joyce. "Poetry Generation and Analysis". In: *Advances in Computers*. Ed. by M. Rubinoff and M. C. Yovits. Vol. 13. Elsevier, 1975, pp. 43–72. DOI: 10.1016/s0065-2458(08)60655-5. URL: https://www.sciencedirect.com/science/article/pii/S0065245808606555.

[16] D. Kobak, R. González-Márquez, E.-Á. Horvát, and J. Lause. *Delving into ChatGPT usage in academic writing through excess vocabulary*. 2024. DOI: 10.48550/arXiv.2406.07016. URL: http://arxiv.org/abs/2406.07016.

[17] C. S. Kulkarni. "Ethical Implications of Large Language Models in Content Generation". In: *Journal of Artificial Intelligence, Machine Learning and Data Science* 1.8 (2022), pp. 62–67. DOI: 10.51219/JAIMLD/chinmay-shripad-kulkarni/32. URL: https://urfjournals.org/open-access/ethical-implications-of-large-language-models-in-content-generation.pdf.

[18] J. H. Lau, T. Cohn, T. Baldwin, J. Brooke, and A. Hammond. "Deep-speare: A joint neural model of poetic language, meter and rhyme". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by I. Gurevych and Y. Miyao. Melbourne, Australia: Association for Computational Linguistics, 2018, pp. 1948–1958. DOI: 10.18653/v1/P18-1181. URL: https://aclanthology.org/P18-1181.

[19] H. Manurung. “Chart Generation of Rhythm Patterned Text”. In: *First International Workshop on Literature in Cognition and Computers* 1 (1999), pp. 15–19.

[20] H. Manurung, G. Ritchie, and H. Thompson. *Towards A Computational Model Of Poetry Generation*. Technical Report. The University of Edinburgh, 2000. URL: https://era.ed.ac.uk/handle/1842/3460.

[21] F. Mélanie-Becquet, C. Plancq, C. Grunspan, M. Maignant, M. Raffard, M. Roussel, F. Ghedini, and T. Poibeau. “Exploring Combinatorial Methods to Produce Sonnets: An Overview of the Oupoco Project”. In: *Digital Humanities Quarterly* 18.1 (2024). URL: https://www.digitalhumanities.org/dhq/vol/18/1/000734/000734.html.

[22] Microsoft. *How to Write Poetry Using Copilot | Microsoft Bing*. 2023. URL: https://www.microsoft.com/en-us/bing/do-more-with-ai/write-poetry-with-bing-compose.

[23] B. L. Monroe, M. P. Colaresi, and K. M. Quinn. “Fightin’ Words: Lexical Feature Selection and Evaluation for Identifying the Content of Political Conflict”. In: *Political Analysis* 16.4 (2017), pp. 372–403. DOI: 10.1093/pan/mpn018. URL: https://www.cambridge.org/core/journals/political-analysis/article/fightin-words-lexical-feature-selection-and-evaluation-for-identifying-the-content-of-political-conflict/81B3703230D21620B81EB6E2266C7A66.

[24] OpenAI et al. *GPT-4 Technical Report*. 2024. DOI: 10.48550/arXiv.2303.08774. URL: http://arxiv.org/abs/2303.08774.

[25] A. Ormazabal, M. Artetxe, M. Agirrezabal, A. Soroa, and E. Agirre. *PoeLM: A Meter- and Rhyme-Controllable Language Model for Unsupervised Poetry Generation*. 2022. URL: http://arxiv.org/abs/2205.12206.

[26] A. Parrish. *aparrish/pronouncingpy*. 2024. URL: https://github.com/aparrish/pronouncingpy.

[27] A. Parrish. *Articulations*. Using electricity. Denver, Colorado: Counterpath, 2018.

[28] H. Tan, M. Duan, D. Liu, HaojieLu, YuexinMu, L. Zhou, A. Ren, Y. Tan, and K. Zhong. “Rethinking Literary Plagiarism in LLMs through the Lens of Copyright Laws”. In: 2024. URL: https://openreview.net/forum?id=sWZy2Xirwt.

[29] M. Tarlinskaja. “Meter and Mode: English Iambic Pentameter, Hexameter, and Septameter and Their Period Variations”. In: *Style* 21.3 (1987), pp. 400–426. URL: https://www.jstor.org/stable/42946214.

[30] M. Tsan, A. Hon, and A. Chun. “Automatic Haiku generation using vsm”. In: (2008).

[31] C. Veltman. “AI is contentious among authors. So why are some feeding it their own writing?” In: *Npr* (2024). URL: https://www.npr.org/2024/04/30/1246686825/authors-using-ai-artificial-intelligence-to-write.

[32] G. D. Vynck. “AI learned from their work. Now they want compensation.” In: *Washington Post* (2023). URL: https://www.washingtonpost.com/technology/2023/07/16/ai-programs-training-lawsuits-fair-use/.

[33] M. Walsh, A. Preus, and M. Antoniak. “Sonnet or Not, Bot? Poetry Evaluation for Large Models and Datasets”. In: *Findings of the Association for Computational Linguistics: EMNLP 2024*. arXiv, 2024. DOI: 10.48550/arXiv.2406.18906. URL: http://arxiv.org/abs/2406.18906.

[34] Z. Wang, W. He, H. Wu, H. Wu, W. Li, H. Wang, and E. Chen. “Chinese Poetry Generation with Planning based Neural Network”. In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Ed. by Y. Matsumoto and R. Prasad. Osaka, Japan: The COLING 2016 Organizing Committee, 2016, pp. 1051–1060. URL: https://aclanthology.org/C16-1100.

[35] X. Yi, M. Sun, R. Li, and W. Li. “Automatic Poetry Generation with Mutual Reinforcement Learning”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Ed. by E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii. Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 3143–3153. DOI: 10.18653/v1/D18-1353. URL: https://aclanthology.org/D18-1353.

[36] M. Zahn. “What is ChatGPT, the artificial intelligence text bot that went viral?” In: *ABC News* (2022). URL: https://abcnews.go.com/Technology/chatgpt-artificial-intelligence-text-bot-viral/story?id=94857599.