so much depends / upon / a whitespace: Why Whitespace Matters for Poets and LLMs

Sriharsh Bhyravajjula[◇] Melanie Walsh[◇] Anna Preus[◇] Maria Antoniak[♠]
[◇]University of Washington [♠]University of Colorado Boulder

Abstract

Whitespace is a critical component of poetic form, reflecting both adherence to standardized forms and rebellion against those forms. Each poem's whitespace distribution reflects the artistic choices of the poet and is an integral semantic and spatial feature of the poem. Yet, despite the popularity of poetry as both a longstanding art form and as a generation task for large language models (LLMs), whitespace has not received sufficient attention from the NLP community. Using a corpus of 19k English-language published poems from Poetry Foundation, we investigate how 4k poets have used whitespace in their works. We release a subset of 2.8k public-domain poems with preserved formatting to facilitate further research in this area. We compare whitespace usage in the published poems to (1) 51k LLM-generated poems, and (2) 12k unpublished poems posted in an online community. We also explore whitespace usage across time periods, poetic forms, and data sources. Additionally, we find that different text processing methods can result in significantly different representations of whitespace in poetry data, motivating us to use these poems and whitespace patterns to discuss implications for the processing strategies used to assemble pretraining datasets for LLMs.

1 Introduction

For many text datasets, whitespace is treated as a minor concern, not critical to a text's meaning. It is often standardized or stripped before further processing. But in poetry, whitespace matters. It is vitally important, perhaps even the most defining feature of the genre (at least on the page). Van Dijk (2011) argues that "there is only one characteristic which immediately distinguishes modern poetry from prose: the blank space surrounding the text." In poetry, whitespace—including line and stanza breaks, indentation, space between words, and more—is not merely stylistic flair but integral to structure, meaning, and the reading experience.

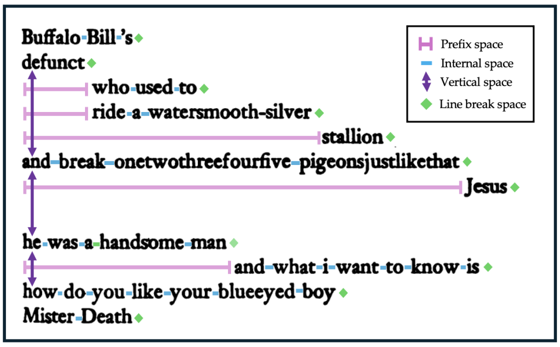


Figure 1: An excerpt from "[Buffalo Bill 's]" (1926) by E.E. Cummings, annotated using our whitespace typology, WISP (Whitespace In Spatial Poetics). WISP distinguishes between five categories of whitespace usage: `line breaks`, `prefix` space, `internal` space, `vertical` space, and `line length`.

Yet whitespace has received relatively little attention in NLP. One reason is that it is often deemed inconsequential. Another is that it is technically challenging to represent and preserve. There are over 25 different Unicode characters that encode whitespace of varying widths and functions. On the web, whitespace is often represented with HTML and CSS styling—a difficult task in its own right, and one that also poses problems for converted plain text formatting. What's more, with poetry, it's not always possible to tell whether a line break or other whitespace reflects the author's intent, the original typesetting, or an artifact of reprinting or digitization. In the digital humanities (DH), scholars studying poetry often painstakingly encode layouts using the XML-based TEI (Text Encoding Initiative),[1] which underscores how central—and labor-intensive—whitespace preservation can be.

Whitespace not only has consequences for poetry but for NLP more broadly. For LLMs, it turns out, whitespace also matters. Research has shown

[1] https://www.tei-c.org

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 35156–35173
November 4-9, 2025 ©2025 Association for Computational Linguistics

that some models fail to account for the visual dimensions of text, and that adversarial attacks can exploit LLM challenges with vertical and horizontal space (Li et al., 2025b; Cai and Cui, 2023). Recent efforts have begun to address these and other issues by trying to preserve structure and whitespace during pretraining data preparation, highlighting growing awareness that layout carries meaning, especially for programming and mathematical data (Paster et al., 2023; Overwijk et al., 2022).

In this work, we conduct a large computational study of poetry focused on whitespace: both its poetic implications and its challenges for LLMs.

Our contributions include:

- a large-scale analysis of whitespace usage across 19k published, 12k unpublished, and 51k generated poems (all in English),
- a focused whitespace usage analysis of published poems across 500 years, 4k poets, and diverse poetic forms,
- a dataset of 2.8k public-domain poems with preserved formatting to facilitate further research in this area,[2]
- the Whitespace In Spatial Poetics (WISP✍) typology, that unifies different poetic studies of whitespace,
- and evaluation of various pretraining linearization systems, including established methods as well as experimental systems using multimodal models, leading to reflections on whitespace handling for LLMs.

2 Related Work

2.1 Humanities Studies of Whitespace

Whitespace is an important textual element of poetry (Drucker, 2006; Brinkman, 2009; Van Dijk, 2011; Johnston, 2010) and a key expressive tool for poets, especially modern and contemporary poets (Halter, 2015; Peterson, 1995; Rollison, 2003; Drucker, 1994). The term “white space” emerged in English print in the late 1800s, referring to “the blank areas of a page or other piece of printed matter,” which were often “regarded collectively as an element of layout and design” (OED, 2025). The *Oxford English Dictionary* notes the term’s shift to a single word with the rise of computing, where it is used to indicate “blank space in electronic text produced by one or more keyed characters, as spaces, tabs, line-breaks, etc” (OED, 2025). We use the one-word term to refer to typographic whitespace in both print and digital contexts.

Long before either version of the term was coined, whitespace performed important functions in written verse, visually signaling the division of poems into lines and line groups. The line is a fundamental concept in poetry (Van der Zee, 2011). While the poetic line is not defined by its visual representation on the page, practically, lines are indicated through whitespace that surrounds them.

Conceptions of the poetic line, poetic form, and the relationship between poems and their visual representation on the page are also historically and culturally specific (Prins, 2008; Martin, 2012; Jackson, 2023). In early modern and 18th-century English poetry, the division of poetic texts into lines generally corresponded to repeated patterns of sound, specifically patterns of meter and rhyme (Fussell, 1965; Brogan, 1981; Attridge, 1982; Martin, 2012). With the rise of free verse in the late 19th and early 20th centuries (Hartman, 1980; Finch, 2000; Beyers, 2001), however, lines were no longer necessarily defined by such metrical patterns, and many poets began influentially incorporating new and varied forms of lineation into their poetry (Hollander, 1975; Berry, 1989; Peterson, 1995; Gross, 1996; Beyers, 2001; Johnston, 2010).

Within this context, whitespace became an increasingly important expressive tool for poets, who incorporated variant spacing within and between their poetic lines and experimented with boundary-pushing typography and layouts (Perloff, 1986; McGann, 1993; Brinkman, 2009; Cundy, 1981). These expressive usages of whitespace are important interpretive aspects of poems. In digital and digitized texts, however, standard and expressive uses of whitespace in poetry may be encoded in a range of ways, and their particularities can get flattened through various technical processes.

2.2 Computational Studies of Whitespace

In the digital humanities, studies of whitepace in poetry have focused on line breaks and enjambment. Most closely related to our work is a study of enjambment by Ruiz Fabo et al. (2017), which considers different types of enjambment in a small dataset of 3.7K Spanish-language sonnets. Using hand-crafted rules and constituency and dependency parses, they detected the presence and type of enjambment and provided a visualization of which line position are more likely to contain

[2] Our data, code, and an interactive dashboard are available at https://github.com/darthbhyrava/wisp

enjambments. Similar work on very small datasets (e.g., $N = 69$) has used audio files as well as syntactic analysis to study types of enjambment (Hussein et al., 2018). Monget (2020) provides a useful overview of this prior work on computational analyses of enjambment.

Prior work in NLP has mostly treated all whitespace uniformly. The places where whitespace has been seriously considered have mostly been (1) language-specific tokenization (Wiechetek et al., 2019) and (2) correction of OCR errors (Soni et al., 2019; Bast et al., 2023). However, there has been recent attention to whitespace formatting in pretraining datasets dedicated to programming and mathematical datasets and tasks (Paster et al., 2023).

Standard processes of macrodata refinement include quality filtering, removal of "junk" text, and tokenization. A critical but sometimes overlooked step is *linearization*, the process by which web scraped data is transformed from HTML to text ready for use in pretraining (Soldaini et al., 2024). Commercial tools exist to support this process, but while some comparisons have been done (Li et al., 2025a), overall the research community has focused on the (also important) effects of quality filters (Lucy et al., 2024), curation (Wettig et al., 2025), and tokenization (Ali et al., 2024; Wang et al., 2025; Whittington et al., 2024; Singh and Strouse, 2024; Zheng et al., 2025), where whitespace is usually treated as a separator rather than a feature with its own importance.

Category	**Sub-Category**	**Definition**
`line break`	standard (!e)	breaks at sentence boundary
`line break`	lexical (e)	a word is split across two lines
`line break`	clausal (e)	a clause (noun and verb) is split across two lines
`line break`	phrasal (e)	a phrase (e.g., adjective and noun) is split across lines
`prefix`	standard	no indent
`prefix`	standard indent	repeated indent that aligns with a form
`prefix`	non-standard	all other indents
`internal`	standard	single white space between tokens
`internal`	non-standard	multiple spaces between words or within a word
`vertical`	standard	a single newline character
`vertical`	standard stanza	two newline characters between stanzas
`vertical`	non-standard	multiple newline characters not at stanza boundaries
`line length`	standard	uniform line lengths across the poem
`line length`	non-standard	non-uniform line lengths

Table 1: Our typology of whitespace usage in poems. (e: enjambed, !e: not enjambed)

3 WISP🪶: A Whitespace Typology

Poets use whitespace in a variety of ways, both as a standard feature of traditional poetic forms (via line or stanza breaks) and as a more idiosyncratic, expressive tool (e.g., through inline spacing, indentation, or irregular line lengths). To formalize these usages, we develop a practical typology of whitespace usage categories: the Whitespace In Spatial Poetics (WISP🪶) typology. These categories can be combined, repeated, interjected, and used for larger patterns to shape the visual structure of a poem. An overview of the typology is in Table 1.

Line Breaks Line breaks refer to spaces that mark the end of a line of text, affecting the length, position, metrical composition, and rhythmic qualities of poetic lines (Beyers, 2001; Rosko and Zee, 2011a; Hazelton, 2014). Line breaks correspondingly hold significant weight for poets (Levertov, 1979; Fagan, 2011; Halter, 2015), often marking the place for rhymes, and in many ways defining the relationship between line and syntax (Longenbach, 2008). Line breaks may come at the end of sentences or syntactic units or they may fragment these units, carrying words, phrases, clauses, or sentences across vertical space.

Line Prefix Space Line prefix spaces refer to instances of leading whitespace before a line, which introduce indentation from the left-margin. Many usages of prefix spaces in printed poetry are fairly standardized. As Ruwet (2014) notes, "The width of the left margin is generally uniform, though the beginnings of some lines may be indented, often at regular intervals." Jacobson (2008) and Pacheco (2006) explore conventions for poetry publication through early printers' manuals, discussing a number of different conventional uses of indentation, including at the beginning of stanzas and to align pairs of rhymed lines. Prefix spacing can also be used in more unconventional ways (Matore, 2024; Drucker, 2006), however, moving beyond traditional indentation to break up the text more radically, as in the poem in Figure 1.

Internal Space Internal space refers to non-standard whitespace that occurs within lines, appearing between or within words. With the shift toward more focus on the visual elements of poetry (Van Dijk, 2011), use of internal space within lines became more important. In her work on letterpress printing, Drucker (1984) notes that "Writing produces a visual image: the shapes, sizes and placement of letters on a page contribute to the message produced, creating statements which cannot always be rendered in spoken language." The use of internal spacing can create this kind of visual feature, and also has other potential effects, including indicating a pause, breaking up a semantic unit, or contributing to broader visual patterning.

Vertical Space Vertical space refers to blank spaces between lines of text, which in digital poems are created through at least two line breaks, which create one or more lines of whitespace between lines of text. In conventional poetry printing, these blank lines were generally used to separate stanzas and line groups (Jacobson, 2008). However, they can also be used in more unconventional and expressive ways. Writing about modern poetic forms in his influential "Lecture on Modern Poetry," Hulme (1908), suggests that the "new verse resembles sculpture rather than music," arguing that "it appeals to the eye rather than to the ear." Vertical spacing is a key element of this kind of sculptural poetry as well as a standard way of dividing up groups of poetic lines.

Line Lengths Line length refers to the number of visible characters or words in a poetic line. The length of a poetic line may be defined by patterns of sound or by visual choices and in either case it holds poetic meaning (Rosko and Zee, 2011b). Hollander (1975) highlights changes in common lengths of poetic lines in American poetry with the rise of free verse, suggesting, there is "a widespread, received, free-verse style marked by a narrow (25-30 em) format, strong use of line-ending as a syntactical marker, etc., which plays about the same role in the ascent to paradise as the received Longfellow style did a century ago." This favor for short, sculptural lines is often associated with 20th-century poets like William Carlos Williams, who Dolin (1993) argues, "created a revolution in poetic form" by emphasizing "the visual properties of the line," especially via concision.

4 Data

We collect three sources of English-language poetry data for comparison: published poems featured on the Poetry Foundation's website, unpublished poems shared in an online community, and LLM-generated poems. We provide an overview of these datasets and their sizes in Table 2.

4.1 Unpublished Poems

We gather 12k poems from `r/OCPoetry` using ConvoKit (Chang et al., 2020). Most of these poems are not tagged with their form by the poet, so we automatically tag each poem with a form using the prompting framework from Walsh et al. (2024), which reported high precision and recall for free-verse using GPT-4. Using this method with GPT 4.1,[3] we identify 7,862 free-verse poems, 2,234 quatrains, 1,237 couplets, 608 tercets, and a smaller number of other forms. These form labels allow us to directly compare whitespace usage in free-verse poems across data sources.

4.2 LLM-Generated Poems

We use GPT-4 (OpenAI) and Sonnet 3.7 (Anthropic) to generate new datasets of poems.[4] To generate poems on diverse themes, we randomly sample one poem for each poet in our Poetry Foundation dataset, resulting in 4,330 poems that we use as seeds whose title and poet are inserted in the prompt, generating three new poems per seed poem. We use two prompt variations: (1) a prompt that explicitly requests the model to use whitespace creatively and (2) a simplified prompt that does not mention whitespace (see Appendix B). Manual examination of the generated poems and explanations reveals that they are nearly all free verse, and so we use these poems in comparison only to free verse poems from Poetry Foundation and Reddit.

4.3 Published Poems

We scrape 19k poems from the public website of the Poetry Foundation, a U.S.-based nonprofit organization that amplifies poetry for a global audience through grants, awards, fellowships, digital outreach, and publication of the *Poetry* magazine.

[3] `gpt-4.1-2025-04-14`

[4] `gpt-4`, `claude-3-7-sonnet-20250219`

Category	Source	Poem Count	Mean Line Count	Most Common Form
Published Poems	Poetry Foundation	19,457	38.1	sonnet
Unpublished Poems	r/OCPoetry (Reddit)	11,984	26.5	free-verse
Generated Poems	GPT-4 (OpenAI)	12,838	11.9	free-verse
	GPT-4 (OpenAI)	12,645	10.5	free-verse
	Sonnet 3.7 (Anthropic)	12,988	12.5	free-verse
	Sonnet 3.7 (Anthropic)	12,987	11.3	free-verse

Table 2: The three datasets that we collect. Vocabulary density represents unique token counts divided by the total token counts. Small differences in poem counts across generated poems are due to generation/parsing errors.

All the poems we analyze are freely available online, but some of the poems are in-copyright. To ensure responsible data sharing, we release only the subset of poems that are in the public domain. In the U.S., as of 2025, works published in or before 1929 have entered the public domain. We share poems that were published in or before 1929, poems whose authors died in or before 1929, and poems that are explicitly marked as in the public domain.

We follow the methods described in Walsh et al. (2024) to measure how many of the poems appear in the Dolma pretraining dataset and how may of the poems were likely seen by a large, industry LLM. We find that 42.5% of a random sample of 3,692 of the Published Poems contain lines with exact matches in Dolma, using the What's In My Big Data (WIMBD) toolkit (Elazar et al., 2024). When attempting to replicate the LLM probes, we found that both OpenAI and Anthropic models now refuse such completion queries.

5 Linearization Methods and Evaluation

Crucially, for our dataset of Published Poems, it is not sufficient to scrape a poem's webpage; that webpage (its HTML or screenshot) must be parsed and converted into text that isolates the poem of interest while preserving whitespace formatting. To transform the scraped data to poem texts, we test a series of linearization and image-to-text systems.

5.1 Methods

HTML to Text We compare a series of linearization systems for converting the scraped HTML to text. These include resiliparse (Bevendorff et al., 2018), trafilatura (Barbaresi, 2021), and jusText.[5] These tools have been used in production of pretraining datasets such as the Pile (Gao et al., 2020), Dolma (Soldaini et al., 2024), the RefinedWeb Dataset (Penedo et al., 2023), OpenWebMath (Paster et al., 2023), and DataComp-LM (Li et al., 2025a). Where possible, we have prioritized using default settings to simulate the processes leading to real pretraining datasets and the real effects of these parsers on poetry data.[6] We run each pipeline over parts of the scraped webpages that isolate the <div> elements that contain the poems. Importantly, these three methods operate on the scraped HTML without accounting for CSS styling or Javascript. As noted by Clueweb (Overwijk et al., 2022), "the HTML alone provides a partial view of a web page," and so this is a limitation of these methods.

WISP-ify As a baseline comparison, we develop a custom HTML-to-text pipeline, WISP-ify, that accounts for the Poetry Foundation's diverse formatting practices. The site uses whitespace in a variety of ways to convey lineation, stanza breaks, and visual emphasis. Our parser accommodates four major styles, including line- and stanza-level <div> elements, single paragraphs with <br> line breaks, multiple <p> tags for stanzas, and center-aligned lines. We convert left-margin spacing from inline CSS styles (e.g., margin-left) into corresponding plain-text indentation. We also normalize typographic features such as ligatures, small caps, and rare Unicode space characters. While our

[5] https://github.com/miso-belica/jusText

[6] **Resiliparse**: preserve_formatting = True, main_content = True, list_bullets = True, alt_texts = False, links = False, form_fields = False, noscript = False, comments = True, skip_elements = None (replicated from the code used to create the Dolma dataset (Soldaini et al., 2024)); **Trafilatura**: include_comments = False, include_links = False, include_tables = False, no_fallback = False, favor_precision = False, favor_recall = False, include_formatting = False (NB: changing include_formatting to True does not alter results for poetry data) (replicated from the code used for DataTrove (Penedo et al., 2024)); **jusText**: justext.get_stoplist(’English’),length_low = 0, length_high = 100000, stopwords_low = 0.0, stopwords_high = 1.0, max_link_density = 1.0, no_headings = False (NB: stopwords are given but not used because of the thresholds) (attempted reasonable defaults).

Method	Macro	Weighted	Composite	Pure	PREFIX	INTERNAL	LINE_BREAKS	VERTICAL	OCR-ERROR
Resiliparse	**51.66**	**52.22**	**49.28**	53.79	**48.44**	**45.83**	63.16	**71.90**	7.89
WISP-ify	50.44	51.04	43.80	**55.88**	45.31	**45.00**	63.16	**70.95**	17.11
jusText	3.35	4.15	2.86	3.41	0.00	0.00	34.21	0.00	15.79
trafilatura	3.11	3.86	2.95	3.28	0.00	0.00	34.21	0.00	**5.26**
Claude Sonnet 4	45.48	46.13	35.41	**56.35**	38.00	42.16	72.13	56.55	31.15
Gemini 2.5 Pro	45.08	45.74	41.47	46.38	33.85	42.74	**78.67**	57.14	16.0
o3	42.80	43.77	33.79	48.56	33.33	37.50	65.79	57.14	31.58

Table 3: Human evaluation of linearization method performance across WISP whitespace types. Italicized methods are image-to-text, the rest are HTML-to-text. Scores representing best performance ±0.1 are bolded.

approach captures many of the site's formatting conventions, others remain unsupported, and the site's underlying structure may evolve in ways that challenge long-term reproducibility.

Image to Text HTML-only linearizers are constrained by an inability to capture CSS/Javascript styling essential to preserving whitespace. We capture "screenshots" of the poem using Playwright[7] browser automation over Poetry Foundation HTML content, specifically targeting $.poem - body$ elements with fixed 1920x1080 viewport rendering. Each poem is thus converted to a PNG file. We pass the image to three instruction-following multimodal models (o3, claude-sonnet-4, gemini-2.5-pro) prompting them to return whitespace-preserving text blocks (Appendix D).

5.2 Human Evaluation Setup

We introduce WISP-Bench to evaluate whitespace preservation fidelity across various linearization methods. WISP-Bench consists of a three-tiered set of pass-or-fail unit-tests, each of which asks: *Given the ground truth image of the poem, does the linearized text accurately capture a specific whitespace property?* This design was inspired by olmOCR (Poznanski et al., 2025), and the unit test guidelines are shown in Appendix C.

We curate a dataset of 76 poems that include whitespace features.[8] For each of our seven linearization methods, the four authors evaluate the linearized text against the corresponding poem "screenshot" on WISP-Bench unit tests, such that each poem-method instance has at least two annotations. As this is very difficult task, requiring careful attention to small changes in whitespace, we resolve disagreements by always preferring labels marking mistakes.

We report pass rates across different WISP types for each method. For aggregation, we use four scores to capture different aspects of the method: (1) Macro: Mean of pass-rates across WISP types, treating each type equally; (2) Weighted: Weighted mean of type pass-rates, biased towards the most frequent whitespace types; (3) Composite: A custom heuristic that penalizes OCR errors (see Appendix C), and (4) Pure: Pass rate across all annotations that have no OCR errors at all.

5.3 How well do different linearization methods capture whitespace patterns?

Results of our human evaluation are shown in Table 3. The relatively low macro scores highlight the complexity of preserving whitespace via linearization methods across modality, a facet not explicitly captured in traditional LLM-OCR benchmarks (Fu et al., 2025). We note that specialized tools parsing HTML structure outperform general extraction methods, particularly due to the presence of hallucinated whitespace in LLMs (high OCR error-rate). We also note that LLMs exhibit similar strengths (line breaks) and weaknesses (prefix/internal spacing), possibly reflecting the common nature of their pretraining practices.

Figure 12 in Appendix A.3 shows prefix and internal whitespace patterns for three methods: resiliparse, trafilatura, and our custom pipeline (see §4.3). We find no meaningful difference between our pipeline and resiliparse, but trafilatura removes all prefix spacing. We find that resiliparse very closely approximates our custom pipeline, while trafilatura and jusText mostly fail to preserve non-standard whitespace usages. Trafilatura in particular is an interesting case, as it is designed to preserve whitespace only in detected code blocks.[9]

We show an extended example in Figure 9 in the Appendix, which highlights the challenges in choosing a linearization pipeline. None of the tested HTML to text methods fully reproduce the spatial arrangement that can be seen on the Poetry

[7] https://playwright.dev

[8] line break: 76 poems, vertical: 70, prefix: 64, internal: 40

[9] https://github.com/adbar/trafilatura/blob/master/trafilatura/htmlprocessing.py#L324

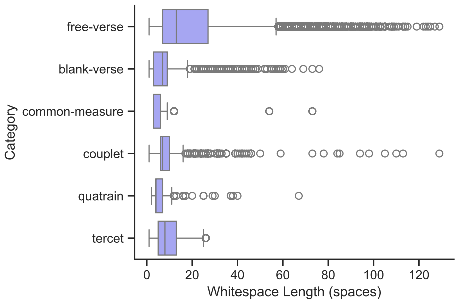


Figure 2: Prefix whitespace lengths, Published Poems.

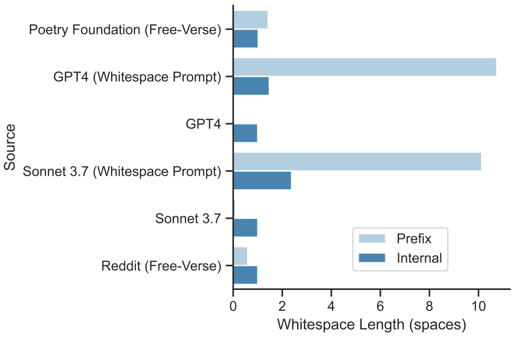


Figure 3: Comparison of prefix and internal mean whitespace usage across the source datasets. To ensure a fair comparison, we compare the generated poems (which are almost all free-verse) only to free-verse poems from Poetry Foundation (as tagged on the website) and Reddit (as predicted using a prompt; see §4.1).

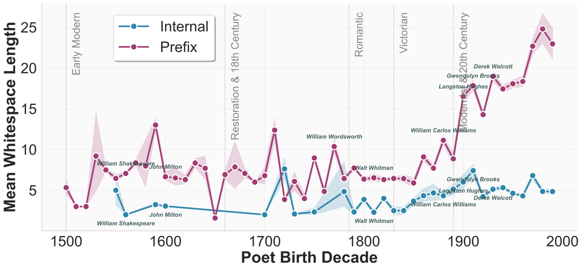


Figure 4: Prefix and internal whitespace usages over time. The y-axis shows the mean number of spaces included in the whitespace, for all non-standard whitespace usages (we excluded non-standard usages from the denominator to highlight increasingly bold usages over time). Shaded areas show 95% confidence intervals, and period lines are based on the *Norton Anthology of English Literature*, 11th edition.

Highest **Prefix** Whitespace Usage			
Tag	N	Proportion	Example Poet
Gay-Lesbian-Queer	184	0.418	Wendy Videlock
Persona	145	0.388	Gottfried Benn
Epigraph	144	0.370	Nick Carbó
Gender-Sexuality	788	0.359	Wendy Videlock
Stars-Planets-Heavens	320	0.347	Amy E. Sklansky
Popular Culture	467	0.345	Allen Ginsberg
Free Verse	4881	0.345	Elizabeth Bishop
Lowest **Prefix** Whitespace Usage			
Tag	N	Proportion	Example Poet
Common Measure	122	0.007	Elinor Wylie
Ballad	117	0.018	[...] Montagu
Funerals	108	0.030	Jean Nordhaus
Quatrain	151	0.031	Adam Zagajewski
Verse Forms	912	0.037	Deborah Paredez
Sonnet	622	0.046	Deborah Paredez
Animals-1	115	0.048	*anonymous*

Table 4: Tags with highest/lowest prefix whitespace.

Foundation website, though some methods come closer than others. Ultimately, the spatial arrangement is a visual problem, which our findings underscore, and this will need to be handled using multimodal models in future work.

In our following analyses, we rely on texts generated with resiliparse, as it is a popular tool and had reasonable performance on WISP🖋-Bench (especially for prefix and internal whitespace).

6 Analysis

Due to space and feasibility constraints, we focus our computational analysis in this paper on three categories: line breaks, prefix spacing, and internal spacing. Our experiments explore whitespace as a stylistic choice and compare whitespace across data sources, tags, and forms.

6.1 How does whitespace vary over published, unpublished, and generated poems?

We find that published poems include more creative or non-standard whitespace (especially prefix spacing) than poems on Reddit, at least when written in free verse (Figure 3), possibly due to formatting difficulties on Reddit. When prompted to generate a poem with no explicit mention of whitespace in the prompt, GPT-4 and Sonnet 3.7 almost never produce poems with non-standard prefix spacing. However, they are clearly capable of producing whitespace-heavy poems. When we use our whitespace specific prompt, the models generate poems with more prefix whitespace on average than the Poetry Foundation poems.

In Figure 5, we observe different kinds of dependency triples occurring at line breaks across datasets. The most common triple across published poems, unpublished human poems, and the default LLM prompt is `VERB -> PUNCT`. This suggests that enjambment often occurs after complete syntactic units, especially after verbs followed by punctuation. It reflects a poetic style that uses enjambment for rhythm, pacing, or breath, not necessarily to

Form	Most Common Punctuation at Line End (Per Total Lines)	Most Likely Punctuation at Line End (Per Punctuation Token Usage)
free-verse	, **(12.6%)** . (10.1%) – (1.1%) ? (0.9%)	; **(41.1%)** ? (33.1%) : (33.1%) . (32.8%)
couplet	, **(26.0%)** . (10.9%) ; (7.8%) : (3.6%)	; **(79.1%)** : (72.5%) ? (52.6%)) (44.7%)
quatrain	, **(18.5%)** . (9.0%) ; (2.5%) – (1.4%)	; **(58.7%)** ? (38.0%) : (36.6%) , (36.4%)
blank-verse	, **(25.6%)** . (8.4%) ; (3.7%) : (2.1%)	) **(48.0%)** . (43.2%) – (42.8%) ? (41.9%)
tercet	, **(10.9%)** . (9.2%) : (0.7%) ? (0.6%)	: **(25.0%)** . (24.9%) , (21.3%) ? (20.2%)
common-measure	, **(29.2%)** ; (10.9%) . (6.6%) ! (1.5%)	; **(89.4%)** , (51.6%) ! (30.5%) . (29.2%)

Table 5: The most common punctuation at line breaks across poetic forms. Left: proportion of lines ending in a punctuation token, normalized by the total number of lines. Right: proportion of a punctuation token ($N >= 100$) appearing at the end of a line, normalized by that token's total usage in any place in a poem.

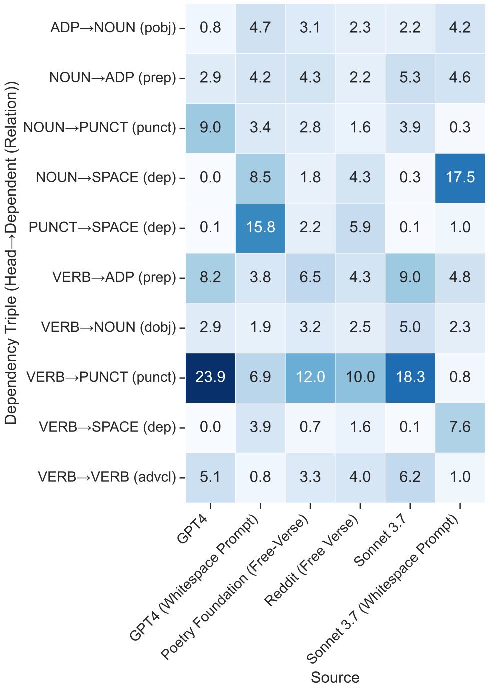


Figure 5: Comparison of most frequent dependency triples that span line breaks across the source datasets.

break grammar mid-thought. It may also reflect how parsers attach punctuation to verbs, making this a common dependency pair in any sentence-final line—especially in free verse.

By contrast, we find that LLMs with the explicit whitespace prompt most often produce `NOUN -> SPACE` or `PUNCT -> SPACE` triples that span across line breaks. In other words, generated poems not only use internal and prefix spacing more frequently, they also use whitespace differently (with different types of line break enjambements) than human-written published or unpublished poems.

6.2 How does whitespace vary by poetic form?

Across all forms, free verse contains the widest variation of whitespace and the most prefix space on average (Figure 2), while couplets include the most internal space on average (Figure 13).

As in §6.1, `VERB -> PUNCT` is the most common dependency triple spanning a line break for all forms in published poems (Figure 11). Table 5 shows differences in the punctuation preceding line breaks across the different forms. Commas are the most common punctuation at line end across all the forms. However, colons (":") and semicolons (";") are more likely to appear at line end than elsewhere in the line, especially for couplets and common measure. Significantly, free verse poems overall have less frequent punctuation at line breaks, reflecting the creative spatial organization that is representative of this form.

6.3 Has whitespace usage changed over time?

Figure 4 suggests that poets have steadily used more whitespace over the last 500 years. We represent poems temporally by the decade of the author's birth year. Birth year has been used in prior work to examine innovation in literary and cultural change (Griebel et al., 2024). We do not control for the number of data points per poet, as poets can and do adapt their stylistic choices over time, and such changes are themselves of literary interest. For any instance of prefix spacing or non-standard internal space, we find the mean number of spaces. We do so to highlight bold and idiosyncratic choices. We see that the size of such whitespace usage is increasing, especially in the 20th century, and especially for prefix spacing.

6.4 How does whitespace vary by topic?

To characterize the kinds of poems with the highest and lowest whitespace usage, we first determine which poems include whitespace lengths above the 75th percentile (calculated using all whitespace lengths from every poem and every tag). We then find the proportion of poems assigned to each tag (manual labels applied by Poetry Foundation) that are in this high whitespace usage category. Tables 4 and 6 show the top tags for prefix and internal whitespace, with example poets whose poem(s) have the highest/lowest whitespace usage among all poems with that tag. We only show tags assigned to at least $N = 100$ poems. As expected, we see tags for traditional forms like "Sonnet" ranked lowest for whitespace usage, while we see tags for modern topics like "Gender-Sexuality" and physicalities like "The Body" ranked highest.

7 Discussion

Paying closer attention to whitespace opens up new avenues for computational literary and cultural analysis, enabling macro-level studies of how poetic form and visual layout have changed over time. In the twentieth century, advancements in printing and typesetting technologies gave poets greater freedom to experiment spatially, and whitespace has become integral to meaning-making, rhythm, and reader engagement. Our findings confirm this scholarly narrative and demonstrate how researchers can explore innovation across historical periods, literary movements, or national traditions.

But we find that distinguishing deliberate whitespace from formatting artifact noise is extremely challenging when a poem has been transferred through various mediums (manuscript to print, print to print, print to digital) and formats (HTML/image/text), due to the inherent typographic inconsistencies of diverse rendering engines, font metrics, character encoding, and responsive layouts. We have also observed, in the dataset of Reddit poems, the importance of different platforms, whose affordances can shape poets' choices. Given the rarity of standardized ground truth (and the difficulties of adjudicating a "ground truth" in this setting, where even archival scholarship might not produce an obvious ranking of one version over another), the development of accurate whitespace linearization methods is crucial for preserving authorial intent—even if mediated by different formats.

More ambitiously, modeling whitespace at this scale might lead to advancements in computational tools for poetry scholarship and digital literary preservation. Multimodal LLM tools could assist in or even partially automate the labor-intensive process of encoding poetic texts using systems like the Text Encoding Initiative (TEI). However, we caution that such systems must always keep domain experts in the loop, as encoding poetry in TEI is a fundamentally interpretive act that involves annotating specific elements of texts for particular goals (Flanders et al., 2016). While some affordances of TEI would be difficult to productively automate, accurately capturing whitespace could cut down significantly on the labor involved in reproducing the layouts of poetic texts (Micir and Preus, 2025).

For LLM data collectors and model builders, poetry provides an instructive test case. While much attention has been given to the formatting of programming and mathematical inputs (Paster et al., 2023), whitespace in poetry is more idiosyncratic, and we do not know of existing off-the-shelf linearization systems that are designed to handle poetry. As prior work has argued (Walsh et al., 2024), poetry is a popular generation task and a "lightning rod" for public imagination around artificial intelligence capabilities, and is worthy of research attention. Practically, we recommend resiliparse as a baseline linearization method for scraped poetry data. However, none of our tested methods faithfully captured all whitespace usage as shown visually on the Poetry Foundation website. Future work will need to tackle the CSS and other styling outside of the HTML and incorporate more advanced multimodal and vision model pipelines.

8 Conclusion

Our work introduces a whitespace typology for poetry, which we use to investigate how 4k poets from the Poetry Foundation have linguistically and syntactically used whitespace in 19.4k poems across 500 years. We compare this usage to 51.4k LLM-generated poems and 11.9k unpublished poems posted in the subreddit `r/OCPoetry` and discuss differences in their distribution. We also discuss the impact of different linearization methods on our results. Finally, we release 2.8k public-domain poems with preserved whitespace formatting to facilitate future work.

9 Limitations

Our whitespace and linguistic analysis is limited to English-language poems in the Roman script and may not translate to poetry in other languages or scripts. Similarly, our representation of poets across time is also restricted to their digital presence on the Poetry Foundation, and hence our conclusions are not truly representative of all English poets of any given time. These poems overrepresent poets from the North American region. In addition, LLMs can “memorize” training data, which often contains copyright-protected literary work. During generation, these models may bear resemblance to the original poems despite our explicit prompt instruction to not reuse original text.

Of course, poems are present in pretraining datasets not only through scraped web data but also through book data (Chang et al., 2023). We observe this even in our scraped poems, which when searched for in Dolma, as described in §4.3, return the most hits from a single domain from Google Books. It is likely that poem texts taken from books also suffer from whitespace issues due to OCR and other errors, but we leave this investigation to future work.

10 Ethical Considerations

The literary community of poets, readers, editors, and publishers faces significant challenges due to recent advances in LLMs and synthetically generated poetry that mimics human verse with unprecedented fidelity on the syntactic level (Porter and Machery, 2024). A poem is a human artistic endeavor that captures the agency, expression, reflection, and communal meaning-making of the poet’s lived experiences. Synthetically generated poems lack this sense of meaning; literary magazines and publishers aiming to filter out such synthetically generated submissions are struggling with the complexity of the task and the increased load of submissions.[10] As *Rattle Magazine* succinctly puts it, “Poetry is a tool for expanding the human spirit, which means poems should be written by humans.”[11] We encourage future work in the computational study of poetry to use WISP🖋 for building effective analysis and detection tools to help the literary community, but acknowledge that our work can also be misused for generative optimizations which hinder such causes instead.

We used Claude (Anthropic) to assist in the generation of boilerplate code used to process the data and produce early versions of figures. All code was tested and most code was re-written after using Claude for brainstorming.

11 Acknowledgments

This work was supported by Doing AI Differently, a joint initiative of The Alan Turing Institute and University of Edinburgh, funded by Arts and Humanities Research Council (AHRC-UKRI). We would like to thank to Kyle Lo and Luca Soldaini (for advice and feedback) and Lynn Cherny, Amit Chaudhary, Barry Haddow, and Mithun Hunsur (for sharing key references). We also thank the Simpson Center for the Humanities at the University of Washington for their general support of digital humanities scholarship. Thank you to the Poetry Foundation, and thank you to the poets Shankar Narayan, Bill Carty, and Jeanine Walker for their inspiration.

References

Mehdi Ali, Michael Fromm, Klaudia Thellmann, Richard Rutmann, Max Lübbering, Johannes Leveling, Katrin Klug, Jan Ebert, Niclas Doll, Jasper Buschhoff, Charvi Jain, Alexander Weber, Lena Jurkschat, Hammam Abdelwahab, Chelsea John, Pedro Ortiz Suarez, Malte Ostendorff, Samuel Weinbach, Rafet Sifa, Stefan Kesselheim, and Nicolas Flores-Herr. 2024. Tokenizer choice for LLM training: Negligible or crucial? In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3907–3924, Mexico City, Mexico. Association for Computational Linguistics.

Derek Attridge. 1982. *The rhythms of English poetry*. London; New York: Longman.

Adrien Barbaresi. 2021. Trafilatura: A Web Scraping Library and Command-Line Tool for Text Discovery and Extraction. In *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 122–131. Association for Computational Linguistics.

Hannah Bast, Matthias Hertel, and Sebastian Walter. 2023. Fast whitespace correction with encoder-only transformers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 389–399, Toronto, Canada. Association for Computational Linguistics.

Eleanor Berry. 1989. Visual form in free verse. *Visible Language*, 23(1).

[10] https://clarkesworldmagazine.com/clarke_04_23/

[11] https://rattle.com/page/submissions/

Janek Bevendorff, Benno Stein, Matthias Hagen, and Martin Potthast. 2018. Elastic ChatNoir: Search Engine for the ClueWeb and the Common Crawl. In *Advances in Information Retrieval. 40th European Conference on IR Research (ECIR 2018)*, Lecture Notes in Computer Science, Berlin Heidelberg New York. Springer.

Chris Beyers. 2001. *A History of Free Verse*. University of Arkansas Press. Google-Books-ID: imhZD-wAAQBAJ.

Bartholomew Brinkman. 2009. Making Modern "Poetry": Format, Genre and the Invention of Imagism(e). *Journal of Modern Literature*, 32(2):20–40. Publisher: Indiana University Press.

T. V. F. (Terry V. F.) Brogan. 1981. *English versification, 1570-1980: a reference guide with a global appendix*. Baltimore: Johns Hopkins University Press.

Shuyang Cai and Wanyun Cui. 2023. Evade Chat-GPT Detectors via A Single Space. *Preprint*, arXiv:2307.02599.

Jonathan P. Chang, Caleb Chiam, Liye Fu, Andrew Wang, Justine Zhang, and Cristian Danescu-Niculescu-Mizil. 2020. ConvoKit: A toolkit for the analysis of conversations. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 57–60, 1st virtual meeting. Association for Computational Linguistics.

Kent K. Chang, Mackenzie Cramer, Sandeep Soni, and David Bamman. 2023. Speak, memory: An archaeology of books known to chatgpt/gpt-4. *Preprint*, arXiv:2305.00118.

David Cundy. 1981. Marinetti and Italian Futurist Typography. *Art Journal*, 41(4):349–352. Publisher: [Taylor & Francis, Ltd., College Art Association].

Sharon Dolin. 1993. Enjambment and the erotics of the gaze in williams's poetry. *American Imago*, 50(1):29–53.

Johanna Drucker. 1984. Letterpress language: Typography as a medium for the visual representation of language. *Leonardo*, 17(1):8–16.

Johanna Drucker. 1994. The visible word: experimental typography and modern art, 1909-1923. Book Title: The visible word : experimental typography and modern art, 1909-1923 ISBN: 9780226165011 Place: Chicago [Illinois.

Johanna Drucker. 2006. Graphical Readings and the Visual Aesthetics of Textuality. *Text*, 16:267–276. Publisher: Indiana University Press.

Yanai Elazar, Akshita Bhagia, Ian Helgi Magnusson, Abhilasha Ravichander, Dustin Schwenk, Alane Suhr, Evan Pete Walsh, Dirk Groeneveld, Luca Soldaini, Sameer Singh, Hanna Hajishirzi, Noah A. Smith, and Jesse Dodge. 2024. What's in my big data? In *The Twelfth International Conference on Learning Representations*.

Kathy Fagan. 2011. *In Praise of Line Breaks*. University of Iowa Press. Google-Books-ID: 5HR53zXJIPAC.

Annie Finch. 2000. *The Ghost of Meter: Culture and Prosody in American Free Verse*. University of Michigan Press. Google-Books-ID: aXyEYoR2ruIC.

Julia Flanders, Syd Bauman, and Sarah Connell. 2016. Text encoding. In *Doing Digital Humanities*, pages 140–158. Routledge.

Ling Fu, Zhebin Kuang, Jiajun Song, Mingxin Huang, Biao Yang, Yuzhe Li, Linghao Zhu, Qidi Luo, Xinyu Wang, Hao Lu, Zhang Li, Guozhi Tang, Bin Shan, Chunhui Lin, Qi Liu, Binghong Wu, Hao Feng, Hao Liu, Can Huang, Jingqun Tang, Wei Chen, Lianwen Jin, Yuliang Liu, and Xiang Bai. 2025. Ocrbench v2: An improved benchmark for evaluating large multimodal models on visual text localization and reasoning. *Preprint*, arXiv:2501.00321.

Paul Fussell. 1965. *Poetic meter and poetic form*. New York, Random House.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2020. The pile: An 800gb dataset of diverse text for language modeling. *Preprint*, arXiv:2101.00027.

Sarah Griebel, Becca Cohen, Lucian Li, Jaihyun Park, Jiayu Liu, Jana Perkins, and Ted Underwood. 2024. Locating the leading edge of cultural change. *Computational Humanities Research Conference*.

Harvey Seymour Gross. 1996. *Sound and form in modern poetry*. Ann Arbor: University of Michigan Press.

Peter Halter. 2015. The Poem on the Page, or the Visual Poetics of William Carlos Williams. *William Carlos Williams Review*, 32(1-2):95–115. Publisher: Penn State University Press.

Charles O. Hartman. 1980. *Free Verse: An Essay on Prosody on JSTOR*. Princeton University Press.

Rebecca Hazelton. 2014. Learning the poetic line.

John Hollander. 1975. *Vision and resonance: two senses of poetic form*. New York: Oxford University Press.

T.E. Hulme. 1908. *Lecture on Modern Poetry*. University of Minnesota Press, Minneapolis, UNITED STATES.

Hussein Hussein, Burkhard Meyer-Sickendiek, and Timo Baumann. 2018. Automatic detection of enjambment in german readout poetry. *Proceedings of Speech Prosody*.

Virginia Jackson. 2023. *Before Modernism: Inventing American Lyric*. Princeton University Press. Google-Books-ID: IOOCEAAAQBAJ.

Jean Alice Jacobson. 2008. *How should poetry look? The printer's measure and poet's line*. Ph.d., University of Minnesota, United States – Minnesota.

Carol Ann Johnston. 2010. Theorizing Typography: Printing, Page Design, and the Study of Free Verse. *The American Poetry Review*, 39(3):45–47. Publisher: American Poetry Review.

Denise Levertov. 1979. On the function of the line. *Chicago Review*, 30(3):30–36.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Gadre, Hritik Bansal, Etash Guha, Sedrick Keh, Kushal Arora, Saurabh Garg, Rui Xin, Niklas Muennighoff, Reinhard Heckel, Jean Mercat, Mayee Chen, Suchin Gururangan, Mitchell Wortsman, Alon Albalak, Yonatan Bitton, Marianna Nezhurina, Amro Abbas, Cheng-Yu Hsieh, Dhruba Ghosh, Josh Gardner, Maciej Kilian, Hanlin Zhang, Rulin Shao, Sarah Pratt, Sunny Sanyal, Gabriel Ilharco, Giannis Daras, Kalyani Marathe, Aaron Gokaslan, Jieyu Zhang, Khyathi Chandu, Thao Nguyen, Igor Vasiljevic, Sham Kakade, Shuran Song, Sujay Sanghavi, Fartash Faghri, Sewoong Oh, Luke Zettlemoyer, Kyle Lo, Alaaeldin El-Nouby, Hadi Pouransari, Alexander Toshev, Stephanie Wang, Dirk Groeneveld, Luca Soldaini, Pang Wei Koh, Jenia Jitsev, Thomas Kollar, Alexandros G. Dimakis, Yair Carmon, Achal Dave, Ludwig Schmidt, and Vaishaal Shankar. 2025a. Datacomp-lm: In search of the next generation of training sets for language models. *Preprint*, arXiv:2406.11794.

Zhecheng Li, Yiwei Wang, Bryan Hooi, Yujun Cai, Zhen Xiong, Nanyun Peng, and Kai-wei Chang. 2025b. Vulnerability of LLMs to Vertically Aligned Text Manipulations. *Preprint*, arXiv:2410.20016.

James Longenbach. 2008. *The Art of the Poetic Line*. Graywolf Press, Minneapolis, MN.

Li Lucy, Suchin Gururangan, Luca Soldaini, Emma Strubell, David Bamman, Lauren Klein, and Jesse Dodge. 2024. AboutMe: Using self-descriptions in webpages to document the effects of English pretraining data filters. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7393–7420, Bangkok, Thailand. Association for Computational Linguistics.

Meredith Martin. 2012. *The Rise and Fall of Meter: Poetry and English National Culture, 1860–1930*. Princeton University Press. Google-Books-ID: GcePuomhDXEC.

Daniel Matore. 2024. *The Graphics of Verse: Experimental Typography in Twentieth-Century Poetry*. Oxford University Press. Google-Books-ID: T8ThEAAAQBAJ.

Jerome J. McGann. 1993. *Black riders: the visible language of modernism*. Princeton University Press.

Melanie Micir and Anna Preus. 2025. Feminist modernist collaboration, then and now: Digitizing Hope Mirrlees's Paris. *Modernism/modernity Print Plus*.

Eulalie Monget. 2020. Computational stylistics: A study of enjambment.

OED. 2025. white space, n.

Arnold Overwijk, Chenyan Xiong, Xiao Liu, Cameron VandenBerg, and Jamie Callan. 2022. Clueweb22: 10 billion web documents with visual and semantic information. *Preprint*, arXiv:2211.15848.

HS Pacheco. 2006. Conventions of typography related to traditional poetry. *DRS Biennial Conference Series*.

Keiran Paster, Marco Dos Santos, Zhangir Azerbayev, and Jimmy Ba. 2023. Openwebmath: An open dataset of high-quality mathematical web text. *Preprint*, arXiv:2310.06786.

Guilherme Penedo, Hynek Kydlíček, Alessandro Cappelli, Mario Sasko, and Thomas Wolf. 2024. Datatrove: large scale data processing.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: Outperforming curated corpora with web data, and web data only. *Preprint*, arXiv:2306.01116.

Marjorie Perloff. 1986. The futurist moment: avant-garde, avant guerre, and the language of rupture. Book Title: The futurist moment : avant-garde, avant guerre, and the language of rupture ISBN: 9780226657318 Place: Chicago.

Rai Peterson. 1995. Readable Silence: Blank Space in E. E. Cummings' Poetry. *Spring*, (4):45–56. Publisher: E.E. Cummings Society.

Brian Porter and Edouard Machery. 2024. Ai-generated poetry is indistinguishable from human-written poetry and is rated more favorably. *Scientific Reports*, 14(1):26133.

Jake Poznanski, Aman Rangapur, Jon Borchardt, Jason Dunkelberger, Regan Huff, Daniel Lin, Christopher Wilhelm, Kyle Lo, and Luca Soldaini. 2025. olmOCR: Unlocking trillions of tokens in PDFs with vision language models. *arXiv preprint arXiv:2502.18443*.

Yopie Prins. 2008. Historical poetics, dysprosody, and "the science of english verse". *PMLA*, 123(1):229–234.

Damian Judge Rollison. 2003. The Poem on the Page: Graphical Prosody in Postmodern American Poetry. *Text*, 15:291–303. Publisher: Indiana University Press.

Emily Rosko and Anton Vander Zee. 2011a. *A Broken Thing: Poets on the Line*. University of Iowa Press. Google-Books-ID: 5HR53zXJIPAC.

Emily Rosko and Anton Vander Zee. 2011b. *A Broken Thing: Poets on the Line*. University of Iowa Press. Google-Books-ID: 5HR53zXJIPAC.

Pablo Ruiz Fabo, Clara Martínez Cantón, Thierry Poibeau, and Elena González-Blanco. 2017. Enjambment detection in a large diachronic corpus of Spanish sonnets. In *Proceedings of the Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 27–32, Vancouver, Canada. Association for Computational Linguistics.

Nicolas Ruwet. 2014. *8. Typography, Rhymes, and Linguistic Structures in Poetry*, page 103–130. University of Texas Press.

Aaditya K. Singh and DJ Strouse. 2024. Tokenization counts: the impact of tokenization on arithmetic in frontier llms. *Preprint*, arXiv:2402.14903.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Harsh Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Pete Walsh, Luke Zettlemoyer, Noah A. Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse Dodge, and Kyle Lo. 2024. Dolma: an Open Corpus of Three Trillion Tokens for Language Model Pretraining Research. *arXiv preprint*.

Sandeep Soni, Lauren Klein, and Jacob Eisenstein. 2019. Correcting whitespace errors in digitized historical texts. In *Proceedings of the 3rd Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 98–103, Minneapolis, USA. Association for Computational Linguistics.

Anton Van der Zee. 2011. *Introduction: New Minds, New Lines*. University of Iowa Press. Google-Books-ID: 5HR53zXJIPAC.

Yra Van Dijk. 2011. Reading the form: the function of typographic blanks in modern poetry. *Word & Image*, 27(4):407–415. Publisher: CAA Website _eprint: https://doi.org/10.1080/02666286.2011.589569.

Melanie Walsh, Anna Preus, and Maria Antoniak. 2024. Sonnet or not, bot? poetry evaluation for large models and datasets. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 15568–15603, Miami, Florida, USA. Association for Computational Linguistics.

Dixuan Wang, Yanda Li, Junyuan Jiang, Zepeng Ding, Ziqin Luo, Guochao Jiang, Jiaqing Liang, and Deqing Yang. 2025. Tokenization matters! degrading large language models through challenging their tokenization. *Preprint*, arXiv:2405.17067.

Alexander Wettig, Kyle Lo, Sewon Min, Hannaneh Hajishirzi, Danqi Chen, and Luca Soldaini. 2025. Organize the web: Constructing domains enhances pretraining data curation. *Preprint*, arXiv:2502.10341.

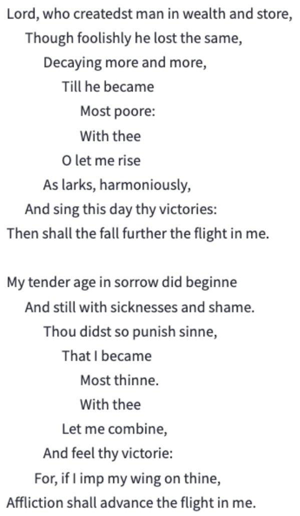
Philip Whittington, Gregor Bachmann, and Tiago Pimentel. 2024. Tokenisation is np-complete. *Preprint*, arXiv:2412.15210.

Linda Wiechetek, Sjur Nørstebø Moshagen, and Kevin Brubeck Unhammer. 2019. Seeing more than whitespace — tokenisation and disambiguation in a North Sámi grammar checker. In *Proceedings of the 3rd Workshop on the Use of Computational Methods in the Study of Endangered Languages Volume 1 (Papers)*, pages 46–55, Honolulu. Association for Computational Linguistics.

Brian Siyuan Zheng, Alisa Liu, Orevaoghene Ahia, Jonathan Hayase, Yejin Choi, and Noah A. Smith. 2025. Broken Tokens? Your language model can secretly handle non-canonical tokenizations. *Preprint*, arXiv:2506.19004.

A Appendix

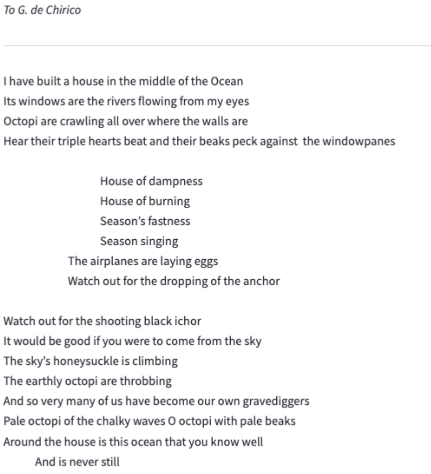
We show examples of poems with complex whitespace usages and provide further results in this Appendix.



Lord, who createdst man in wealth and store,
Though foolishly he lost the same,
Decaying more and more,
Till he became
Most poore:
With thee
O let me rise
As larks, harmoniously,
And sing this day thy victories:
Then shall the fall further the flight in me.

My tender age in sorrow did beginne
And still with sicknesses and shame.
Thou didst so punish sinne,
That I became
Most thinne.
With thee
Let me combine,
And feel thy victorie:
For, if I imp my wing on thine,
Affliction shall advance the flight in me.

Figure 6: “[Easter Wings]” by George Herbert (1593—1633), from the Poetry Foundation.



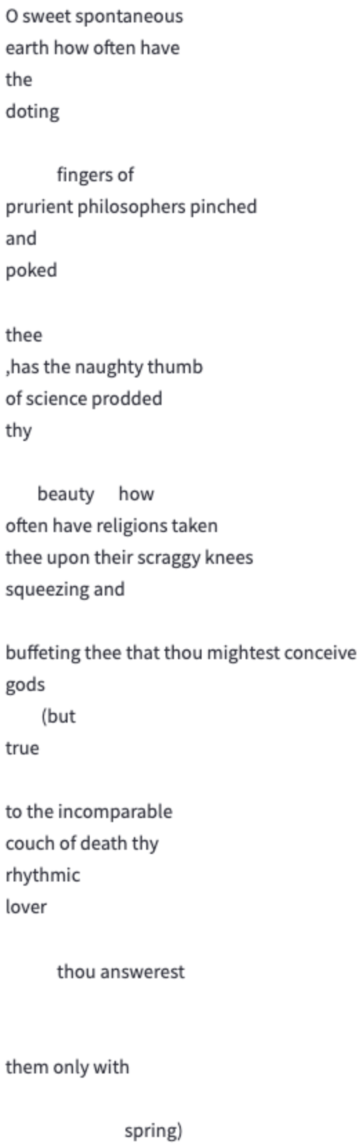
To G. de Chirico

I have built a house in the middle of the Ocean
Its windows are the rivers flowing from my eyes
Octopi are crawling all over where the walls are
Hear their triple hearts beat and their beaks peck against the windowpanes

House of dampness
House of burning
Season's fastness
Season singing
The airplanes are laying eggs
Watch out for the dropping of the anchor

Watch out for the shooting black ichor
It would be good if you were to come from the sky
The sky's honeysuckle is climbing
The earthly octopi are throbbing
And so very many of us have become our own gravediggers
Pale octopi of the chalky waves O octopi with pale beaks
Around the house is this ocean that you know well
And is never still

Figure 7: “[Ocean of Earth]” by Guillaume Apollinaire (1880-1918), translated from French by Ron Padgett



O sweet spontaneous
earth how often have
the
doting

fingers of
prurient philosophers pinched
and
poked

thee
,has the naughty thumb
of science prodded
thy

beauty how
often have religions taken
thee upon their scraggy knees
squeezing and

buffeting thee that thou mightest conceive
gods
(but
true

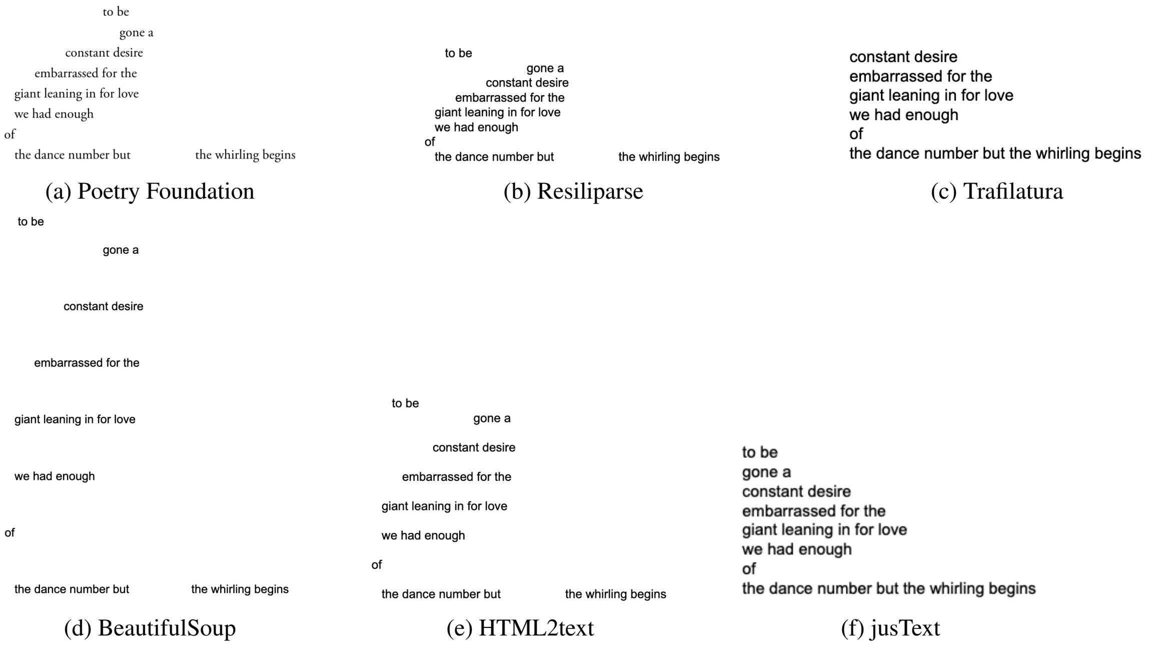
to the incomparable
couch of death thy
rhythmic
lover

thou answerest

them only with

spring)

Figure 8: “[O sweet spontaneous]” (©1923) by E.E. Cummings, from the Poetry Foundation.


to be
gone a
constant desire
embarrassed for the
giant leaning in for love
we had enough
of
the dance number but
the whirling begins
the dance number but the whirling begins

(a) Poetry Foundation (b) Resiliparse (c) Trafilatura

(d) BeautifulSoup (e) HTML2text (f) jusText

Figure 9: Comparisons of the opening lines of the poem “Mars.1” (2016) by CAConrad across different HTML to text methods.

A.1 Comparison of HTML to Text Methods

A.2 Whitespace, Part-of-Speech, and Dependency Triples by Poetic Form

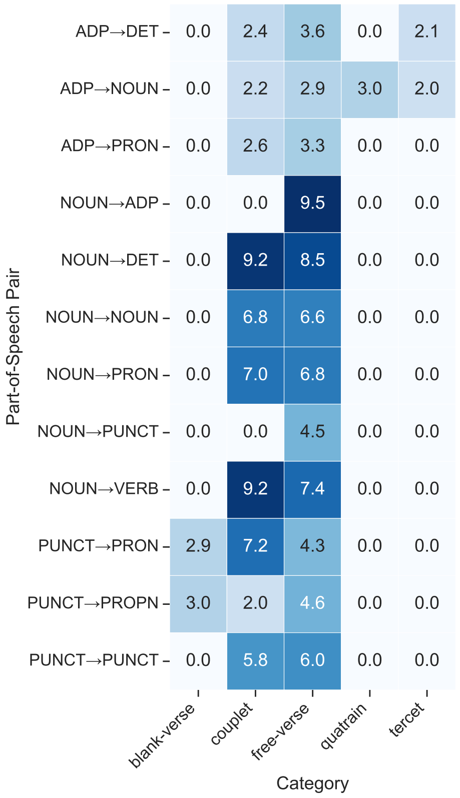


Figure 10: The average internal whitespace length between pairs of POS tags for the Published Poems parsed using resiliparse.

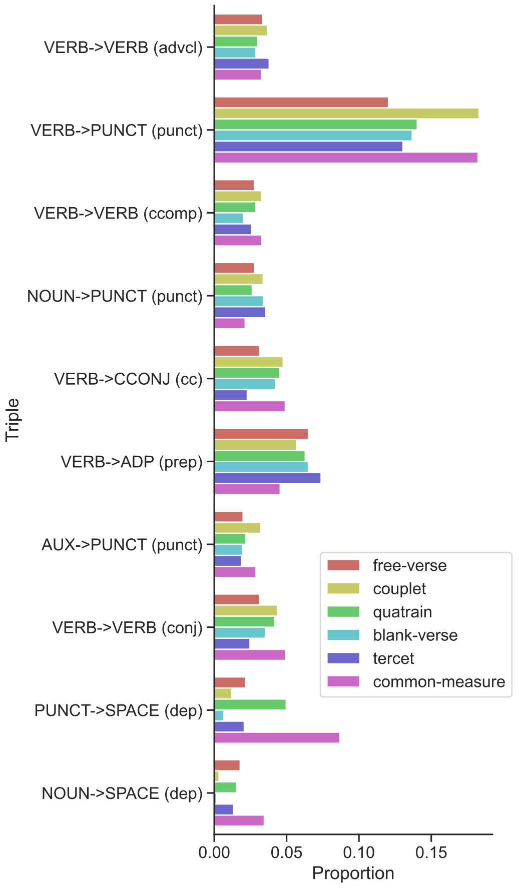


Figure 11: The proportions of the most common dependency triples (head `POS->dependent POS (relation type)`) that span across line breaks for the Published Poems parsed using resiliparse. These proportions represent only lines *not* ending at a sentence boundary.

A.3 Linearization Comparison

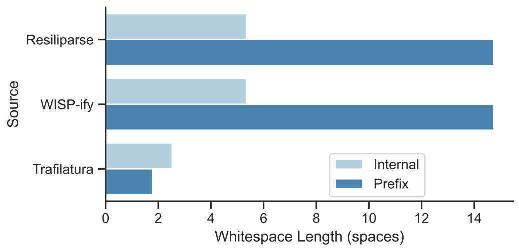


Figure 12: Comparison of prefix and internal mean whitespace lengths across three HTML to text methods, including our custom pipeline described in §4.3. These results are normalized only by the total number of non-standard usages, not the total number of lines or internal spaces, to highlight differences.

A.4 Forms and Whitespace

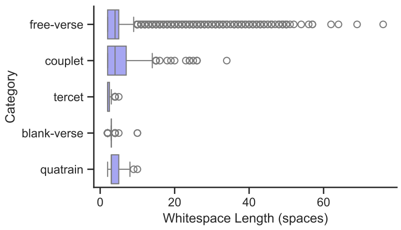


Figure 13: Lengths of internal whitespace usages for Published Poems.

A.5 Tags and Whitespace

Highest **Internal** Whitespace Usage			
Tag	N	Proportion	Example Poet
Ghosts-the-Supernatural	163	0.453	Ching-In Chen
Gender-Sexuality	788	0.373	May Swenson
Refrain	162	0.347	Adam O. Davis
Series-Sequence	271	0.326	Toi Derricotte
Grief	1840	0.323	Terisa Siagatonu
Theater-Dance	130	0.322	Penelope Shuttle
The Body	1737	0.311	Toi Derricotte
Lowest **Internal** Whitespace Usage			
Tag	N	Proportion	Example Poet
Common Measure	122	0.000	Robert W. Service
Valentine's Day	119	0.000	Sir Philip Sidney
Blank Verse	235	0.006	Robert Pinsky
Tercet	121	0.006	Tom Sleigh
Funerals	108	0.008	Jean Nordhaus
Simile	113	0.009	[...] Anne Finch
Rhymed Stanza	1702	0.027	Edmund Spenser

Table 6: Tags with highest/lowest internal whitespace.

B Poem Generation Prompt

Poem Generation Prompt (Whitespace)

I'm very interested interested in how you use whitespace for poetry data. Could you display your capabilities by writing three new poems inspired by the themes of the poem "`poem_title`" by `poet_name`.

I want your new poems to use whitespace creatively, in ways that are appropriate for each poem. Each poem should use whitespace differently. This could include enjambment, vertical spacing between lines, prefix spacing before the first word in a line, or line-internal spacing between or within words.

Do not use any text from the original poem. Print your new poems inside <poem></poem> tags and then provide explanations of your whitespace usage inside <explanation></explanation> tags. Make sure your output is in plain text and do not include a title.

C WISP🪶-Bench

C.1 A Three Tiered Benchmark

Given the "spectrum of correctness" of whitespace fidelity, WISP-Bench has three hierarchical tiers of evaluation:

- **Presence Match** Structural Fidelity - do the basic spatial elements (line break/prefix/internal/vertical spacing) exist where they should?

- **Fuzzy Match** Relational Fidelity - are the proportional relationships between whitespace elements preserved? For example, if two consecutive whitespace elements in the image are 2 and 4 spaces, and their respective textual counterparts are 4 and 8 spaces, relative spatial presence is said to be preserved.

- **Exact Match** Absolute Fidelity - has the precise visual layout and appearance been preserved? While this is difficult to evaluate due to the challenge of transforming pixels to characters, this requires exact correspondence of structure.

C.2 Unit Tests in the Benchmark

1. **Line Break Test (Presence)**
   Question: Does the text capture line breaks where they should be?
   Check If: The first and last words of the printed line N (between two \ns) in the text match their corresponding positions in the image, for all N.

2. **Prefix Space Tests**

- **2a. Prefix (Presence)**
  Question: Is indentation preserved at all?
  Check If: There is at least one instance of a prefix whitespace being preserved.
- **2b. Prefix (Fuzzy)**
  Question: Are relative indentation levels preserved?
  Check If: Ranking of indentation depths matches (line A more indented than B), if there's more than 1 prefix whitespace line in the poem.
- **2c. Prefix (Exact)**
  Question: Are exact indentation levels preserved?
  Check If: Number of leading spaces/tabs matches within tolerance (± 1 space). Does this pass the eye test—does the prefix spacing *look* perfectly preserved?

3. **Internal Space Tests**

- **3a. Internal (Presence)**
  Question: Is extra spacing between words preserved?
  Check If: There is at least one instance of an internal whitespace being preserved.
- **3b. Internal (Fuzzy)**
  Question: Are relative internal spacing levels preserved?
  Check If: Ranking of internal space depths is preserved (word pair AB more indented than CD), if there's > 1 internal whitespace word pair in the poem.
- **3c. Internal (Exact)**
  Question: Are exact internal spacing amounts preserved?
  Check If: The number of internal spaces matches within tolerance. Eye test—does the internal spacing *look* right?

4. **Vertical Space Tests**

- **4a. Vertical Space (Presence)**
  Question: Is vertical spacing (> 1 newline) preserved?
  Check If: There is at least one instance of 2 newline characters / 1 blank line present between lines.
- **4b. Vertical Space (Relative)**
  Question: Are relative vertical spacing levels preserved?
  Check If: Ranking of vertical space matches (line pair AB more separated than CD), if there's > 1 vertical-space line pair in the poem.
- **4c. Vertical Space (Exact)**
  Question: Are exact vertical spacing amounts preserved?
  Check If: The number of newlines between the lines is preserved (no tolerance since newlines are conspicuous). Eye test: Do the new lines *look* right?

NOTE: We have left out line_lengths from the annotation due to challenges in devising unit tests for this type of whitespace usage.

C.3 Scoring Metrics

Let U denote the set of unit tests, A_u the annotations containing unit test u, and T_u true accepts for option u. Let annotation sets be partitioned as *catastrophic*: C (only OCR Error is labeled true, other tests are marked false); *mixed*: M (OCR Error is true, but there is at least one unit test that has passed); and *pure*: P (OCR Error is false).

Reliability Factor

$$R = 1 - \left(\frac{|C|}{|A|} + 0.5 \times \frac{|M|}{|A|} \right) \tag{1}$$

Macro Score

$$\text{Macro} = \frac{1}{|U|} \sum_{u \in U} \frac{|T_u|}{|A_u|} \times 100 \tag{2}$$

Weighted Macro Score

$$\text{Weighted} = \frac{\sum_{u \in U} |T_u|}{\sum_{u \in U} |A_u|} \times 100 \tag{3}$$

Composite Score

$$\text{Composite} = \text{Macro} \times R \tag{4}$$

Pure Score

$$\text{Pure} = \frac{1}{|U|} \sum_{u \in U} \frac{|T_u \cap P|}{|A_u \cap P|} \times 100 \tag{5}$$

D OCR Transcription Prompt for Multimodal LLMs

```
SYSTEM_PROMPT = """
## Objective:
Convert the poem image into plain text with exact preservation
       of its visual layout (spacing, alignment, and line
      breaks). Prioritize fidelity to the image structure and
      visual layout over standard formatting. Your task is
      purely transcription with layout preservation. Do not
      interpret, explain, or modify the text.


## Formatting Guidelines:
 Here are some guidelines to help with edge cases:
 - Use □ for unreadable characters
 - Ignore all typographical formatting like *italics*, **bold
      **, 'underline', or strikethrough. Transcribe only the
      text and its spacing.
 - **DO NOT** auto-wrap long lines. If a line in the image is
      very long, it must be preserved as a single line in the
      output, as line breaks (enjambment) are a poetic device.
 - In case of columnar poems, maintain the column structure
      using spaces in each row to preserve visual structure.
      Make sure the rows are aligned correctly across all
      columns.
 - If text is centered or right-aligned, replicate the
      alignment using spaces so it visually matches the image.
 - If there are gaps within a line (e.g., scattered words or
      concrete poetry effects), preserve the spacing exactly
      as in the image.
 - Alignment/indentation: Align word positions precisely with
      reference lines above/below, preserving exact
      indentation levels between successive lines. For
      instance, if the word 'foo' in the second line is spaced
       in a way that the 'f' aligned with the 'b' in the word
      'bar' in the previous line in the image, then it should
      be reflected similarly in the text.
 - In case of newlines/vertical spacing, preserve the exact
      number of newlines and vertical gaps as seen in the
      image.
 - In case of concrete poems / scattered poems, the visual
      layout of the image is a part of the semantics of the
      poem. Capture it faithfully as possible with spaces.
 - Accurately represent all non-English and special characters
       (é, ç, ß, etc.) using their exact Unicode code points.
      Do not use approximations (e.g., don't replace é with e).

- Use appropriate single Unicode characters for superscripts/
      subscripts (e.g., ², ₁).
 - For erasure/blackout poetry, transcribe only the visible
      text and use spaces to represent the blacked-out areas,
      preserving the position of the remaining words.
 - In case of page numbers and sections breaks, preserve the
      layout and spacing exactly as it appears in the image.
 - For superscript/subscript/interpolation of multiple
      characters, use the appropriate Unicode characters (e.g.,
       ² for superscript 2, ₁ for subscript 1) and ensure they
       are placed correctly in relation to the surrounding
      text.
 - In case of rotated/upside-down characters, use the
      corresponding Unicode character wherever possible.
 - **Ligatures:** Decompose typographic ligatures into their
      constituent characters (e.g., transcribe '' as 'fi', ''
      as 'fl', and 'æ' as 'ae').

## Prioritization in Cases of Conflict
All guidelines serve the primary objective, but if rules
      appear to conflict, follow this strict priority order:
 - **Most Important** Global Layout > Local Spacing:
      Prioritize the overall "shape" and structure. If
      maintaining the exact space count between two words
      causes a column or a centered block to become misaligned,
       always prioritize the global alignment (the column's
      starting position, the text's center point) over the
      exact local space count.
 - **Specific Poem Types > General Rules:** Rules for specific
       types (like `erasure poetry`) **always override**
      general formatting rules (like `ignore all...
      strikethrough`).
 - Visual Alignment > Semantic Characters: The highest
      priority is to make the text output *look* like the
      image. Instructions to use specific Unicode characters (
      like `²` or `₁`) or to decompose ligatures (like `` to `
      fi`) must **be ignored** if following them would alter
      the character count or width in a way that breaks the
      poem's visual alignment. In such a conflict, transcribe
      the characters *exactly as needed to hold the visual
      shape*, even if it means using standard characters (like
       `f` and `i` separately) to match the layout.

## Output Format:
- Output must consist of exactly one fenced code block
      containing only the transcription. Do not include
      explanations, labels, or commentary outside the block.
- Output must be valid UTF-8 text using only ASCII spaces (U
      +0020) and standard line breaks (LF: U+000A) for
      whitespace.
"""
```