Lyrical Logits: Exploring Neural Poetry Generation

Saga Hansson

Abstract

This study explores the topic of poetry generation using two types of recurrent neural networks (RNN): Long Short-Term Memory networks (LSTM) and Gated Recurrent Unit networks (GRU). The data used to train the model, obtained from poetryfoundation.org, consists of just under 16K poems by approximately 3K different poets. In this study, the issues surrounding defining and evaluating poetry are also discussed. Evaluation was performed by consulting a small group of human judges. Comparing the two types of RNNs, no substantial differences were found between the poetry generated by LSTM-based and GRU-based models.

1 Introduction

As humans, we would probably like to think that creativity is a phenomenon unique to us, but whether or not that is the case is debatable. Poetry is one outlet of creativity, and the desire to automatically generate it has existed since the 1950's. Strachey (1954) created a type of poetry/love letter generator by defining sentence templates, in which a part-of-speech is specified for each slot, and having a computer randomly selecting words for those slots. Since then, a multitude of approaches to poetry generation has been explored. In recent years, the most popular approach is undoubtedly neural networks. Gonçalo Oliveira (2017) and Franceschelli and Musolesi (2021) both provide detailed overviews of the field.

Much of the previous research has been done on Chinese, for example, Zhang and Lapata (2014) and Yan (2016) both combined convolutional neural networks (CNN) and RNNs to generate Chinese quatrains, a type of four line poem. Similarly, both Wang et al. (2016) and Yi et al. (2017) generated Chinese quatrains using a RNN accompanied by an attention mechanism to generate Chinese quatrains. Yan et al. (2016) generated Chinese couplets, two line poems following certain constraints, also by employing a RNN with attention.

Another focus of the previous research on poetry generation has, as expected, been on the English language. Many researchers have generated sonnets – traditionally 14-line verses written in iambic pentameter – including Ghazvininejad et al. (2016), who combined a Finite-State Acceptor with a LSTM; Hopkins and Kiela (2017), who used a character-level LSTM, combined with a cascade of Weighted Finite-State Transducers; Xie et al. (2017), who experimented with different character- and word-level neural networks: vanilla RNN, LSTM and GRU; Lau et al. (2018), who combined word- and character-level LSTMs with an attention mechanism.

Over the past few years, the research on poetry generation has expanded to many other languages, including, but not limited to, the following: Zugarini et al. (2019) generated Italian tercets, three-line poems, in the style of Dante Alighieri's Divine Comedy, by training a syllable-level LSTM; Milanova et al. (2019) implemented a LSTM to generate both news articles and poetry in Macedonian; Chy et al. (2020) experimented with character-, word- and sentence-level representations when training a LSTM/Bidirectional LSTM (BiLSTM) to generate Bengali poetry; implementing a LSTM with an attention mechanism, You et al. (2020) generated Korean poetry, while Mukhtar and Joglekar (2021) generated Urdu and Hindi poetry, whereas Kumar (2021), taking it one step further, also added a CNN to generate poetry in Hindi.

As detailed above, LSTMs have been widely used in the area of poetry generation, however, GRUs are not nearly as popular. Therefore, both LSTM-based and GRU-based models will be employed to generate poetry in this project, and will subsequently be compared.

In Section 2, a definition of poetry is examined in detail, and a combination of two other definitions is suggested. Section 3 introduces the dataset and how it was processed, as well as the general architecture of the LSTM/GRU models. Thereafter, in Section 4, the problem of evaluating poetry is tackled, and the method of evaluation in this study is presented. The results of the study are detailed and discussed in Section 5, after which, in Section 6, some final conclusions about the study and propositions of future work are presented.

2 Defining poetry

In order to generate and evaluate poetry, the subject of defining poetry must be discussed first. Manurung et al. (2012, p. 47) noted the following: "a poem is a natural language artifact that simultaneously satisfies the constraints of grammaticality, meaningfulness and poeticness", where the constraints limits the poem to be syntactically correct, to be "meaningful under some interpretation" and to possess characteristics that non-poetic text lacks (e.g. metre). These constraints, and similar ones, have been used by various authors for evaluating machine-generated poetry (e.g., Zhang and Lapata (2014)). Although the constraints may work well for classical poetry, they disregard modern poetry. For example, much of E.E. Cummings' work would not be considered poetry, since it is occasionally ungrammatical.[1]

A broader definition of poetry can be found in the Oxford English Dictionary, where poetry is described as a "[l]iterary work in which the expression of feelings and ideas is given intensity by the use of distinctive style and rhythm".[2] Similarly, in the Merriam-Webster Dictionary, poetry is defined as "writing that formulates a concentrated imaginative awareness of experience in language chosen and arranged to create a specific emotional response through meaning, sound, and rhythm".[3]

Although these definitions would include modern poetry (such as that of E.E. Cummings), they also theoretically include gibberish, as long as one can argue that feelings and ideas are expressed. Despite this caveat, this project will use the wider definition of poetry.

3 Materials and methods

3.1 Poetry Foundation Dataset

The Poetry Foundation Dataset is comprised of over 15K poems written by 3310 different poets.[4] Among the authors are legends such as Emily Dickinson, William Shakespeare and William Wordsworth, but also modern poets such as Maya Angelou and Louise Glück. Each poem in the dataset is assigned a unique Poetry Foundation ID, which can be used to locate the poem on https://www.poetryfoundation.org/.

3.1.1 Processing

As the dataset contains a number of old English poems, poems consisting only of old English characters were filtered out. Although this procedure only removes six poems, it seemed to be necessary to prevent the generated poems from containing old English words.

Punctuation was padded with whitespace in order to avoid word pairs like *hello* and *hello!*, which would increase the size of the vocabulary drastically. All words in the poems were lowercased, in an additional effort to keep the size of the lexicon small.

3.2 Method

The models consist of an embedding layer, one or multiple GRU/LSTM layers, followed by a single linear layer to obtain logits, which are subsequently turned into probabilities using softmax.

As shown in the Table 1, eight hyperparameter combinations were tested, both using LSTM and GRU, for a total of 16 models. A batch size of 40 was consistently used for two reasons: increasing the batch size was not possible due to computational limitations, and decreasing the batch size would theoretically increase the already quite long training time of the network.

The text generation was done using *top-k sampling* (Fan et al., 2018), with testing done using top-k values from 1 to 50.[5]

[1] The poem *anyone lived in a pretty how town* is one such example, available at https://www.poetryfoundation.org/poetrymagazine/poems/22653/anyone-lived-in-a-pretty-how-town.

[2] The entry for *poetry* in the Oxford English Dictionary can be found here: https://www.lexico.com/definition/poetry.

[3] The Merriam-Webster entry for *poetry* can be found here: https://www.merriam-webster.com/dictionary/poetry#synonyms.

[4] The dataset is available at: https://www.kaggle.com/johnhallman/complete-poetryfoundationorg-dataset

[5] As this resulted in quite a large number of texts, only one top-k value will be evaluated in this project. The remaining poems are available at https://github.com/sagahansson/lt2316-ml-project/tree/main/models-outputs.

Model	Batch	Seq. length	Emb. size	Hidden size	Epochs	LR	GRU	Layers
m1gru	40	32	32	32	20	0.001	True	1
m1lstm	40	32	32	32	20	0.001	False	1
m2gru	40	32	32	32	10	0.001	True	1
m2lstm	40	32	32	32	10	0.001	False	1
m3gru	40	32	32	64	20	0.001	True	1
m3lstm	40	32	32	64	20	0.001	False	1
m4gru	40	32	32	128	20	0.001	True	1
m4lstm	40	32	32	128	20	0.001	False	1
m5gru	40	32	128	32	20	0.001	True	1
m5lstm	40	32	128	32	20	0.001	False	1
m6gru	40	50	32	32	20	0.001	True	1
m6lstm	40	50	32	32	20	0.001	False	1
m7gru	40	32	32	32	20	0.001	False	10
m7lstm	40	32	32	32	20	0.001	True	10
m8gru	40	32	32	32	20	0.01	True	1
m8lstm	40	32	32	32	20	0.01	False	1

Table 1: Hyperparameter settings for the implemented model variations. The *GRU* setting refers to which type of RNN was used: GRU (*True*) or LSTM (*False*) – this is also reflected in the model name.

4 Evaluation

Evaluating poetry is no easy feat, as is evident from the large number of evaluation methods used in previous research (see e.g. Gonçalo Oliveira (2017)). Two of the most common methods for evaluating poetry are human evaluation and the BLEU score. As BLEU, traditionally used for evaluating machine translation, compares the generated text with a gold standard, it is not suitable for this project, since none of the generated poems have a gold standard. Therefore, this project is evaluated with human judgements.

A survey was sent out to 12 participants, out of which four answered. In the survey, the participants were asked to determine whether 18 texts could be considered poetry or not,[6] based on their own intuition. Out of the 18 texts, 14 are generated by the models detailed in Table 1 with `topk=5`,[7] and four were poems randomly chosen from the dataset. To minimise the difference between the human-made and the model-generated poems, all poems are 100 tokens (words or punctuation) each. The human-made poems were cut off at 100 tokens, whereas the models were set to only generate 100 tokens.[8]

5 Results & discussion

As mentioned in Section 4, models *m7gru* and *m7lstm* were discarded as they were deemed to be useless. The two poems generated by these models, Poems 3 and 4, shown in Appendix B, are very similar and repetitive. With the other models, this repetition occured at a lower top-k setting in the generation process, typically `topk=1`. The cause of this issue is likely the depth of the neural network, as models *m7gru* and *m7lstm* were the only models with stacked LSTMs/GRUs, and the only models that suffered from this issue.

Figure 1 shows the human judgements of the question *Is this poetry?* of the `topk=5` poems by each model or poet. It is clear that the human-made poems, in general, were likelier to be considered poetry, with each of them scoring three *Yeses* each, compared to the model-generated poems, which scored an average of 1.36 *Yeses* (see Table 2). The highest number of *Yeses*, however, was achieved by the poem generated by model *m5gru* (shown in Poem 3), which all annotators agreed could be

[6] The exact question in the survey, in relation to each model-generated text or human-made poem, was *Is this poetry?* to which the survey participant could answer *Yes*, *No* or *Maybe*. The general survey instructions are available in Appendix A.

[7] Texts by models *m7gru* and *m7lstm* were not included, as they were determined to obviously not be poetry, see Section 5 for further discussion and Appendix B to view the discarded poems.

[8] Due to a computational error, the models generated poems containing 102 tokens. This error was discovered after the survey was distributed, and consequently, was not amended. The difference in length between the human-made and the model-generated poems is determined to be negligible.

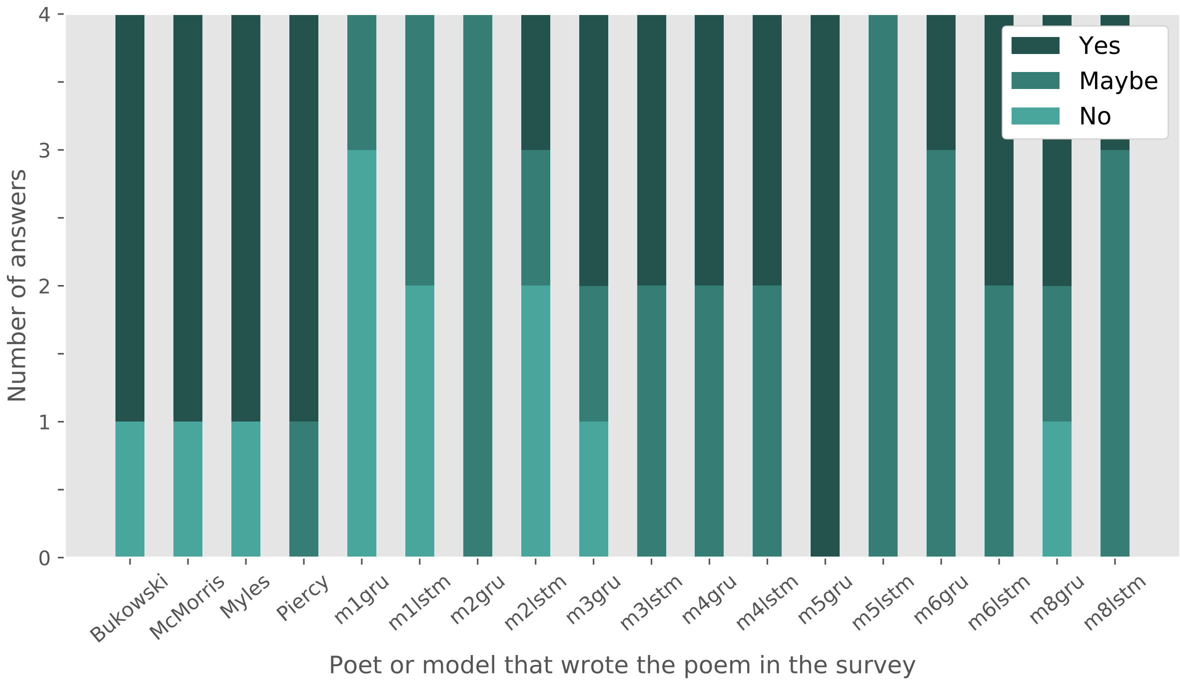

Figure 1: Results of the human judgements. The poet/model names on the x-axis represent the `topk=5` poems written by each poem/model.

	Yes	**Maybe**	**No**
LSTM	1.14	2.30	0.60
GRU	1.60	1.71	0.71
Total	1.36	2.00	0.64

Table 2: Average scores achieved by the different model variations.

considered poetry. This particular poem does not follow syntactic rules, which substantiates the need for a broader definition of poetry than the one suggested by Manurung et al. (2012).

There is no apparent difference between the results of the LSTM models and the GRU models in Figure 1 – *m4gru* and *m4lstm* got the exact same results, and most other LSTM/GRU versions of the same model got fairly similar results (e.g. *m1gru* scored one *Maybe* and three *Noes*, while *m1lstm* scored two *Maybes* and two *Noes*). The largest difference is perhaps between model *m5gru*, which, as previously mentioned, got four *Yeses*, and *m5lstm*, which got four *Maybes*.

As shown in Table 2, the GRU models achieved a higher average score of *Yeses* and *Noes* than the LSTM averages and the total. The LSTM models, on the other hand, obtained a higher average score of *Maybes* than the GRU average and the total. This suggests that the evaluation participants may have been more confident regarding evaluating the texts generated by the GRU models, and more uncertain when evaluating the texts by the LSTM models.

To measure the reliability of the annotator evaluations, I used the `statsmodels` implementation of Fleiss' kappa (κ) (Fleiss, 1971).[9] The inter-annotator agreement score for the text evaluation showed slight agreement ($\kappa = 0.17$) according to the seemingly most widely used interpretation, provided by Landis and Koch (1977). It should be noted that the suggested interpretation by Landis and Koch (1977), and similar ones, have been criticised (see e.g. Bakeman et al. (1997)). Nevertheless, $\kappa = 0.17$ does indicate that the annotators are rather far from agreeing with one another. This either suggests that the concept of poetry is highly subjective, or that the task is ill-defined.

Regarding the definitions of poetry suggested in Section 2, wherein poetry is defined as "[l]iterary work in which the expression of feelings and ideas is given intensity by the use of distinctive style and rhythm" and "writing that formulates a concentrated imaginative awareness of experience in language chosen and arranged to create a specific emotional response through meaning, sound, and

[9]The `statsmodels` implementation of Fleiss' kappa, among other metrics, is available here: https://www.statsmodels.org/stable/generated/statsmodels.stats.inter_rater.fleiss_kappa.html#statsmodels.stats.inter_rater.fleiss_kappa.

rhythm", it is difficult to decide whether to consider the texts in Appendix C poetry. On the one hand, it is hard to imagine a neural network expressing feelings and ideas, and trying to elicit an emotional response through text, as those are arguably human things, but on the other hand, it is not impossible to imagine that I, as the creator of the models, am the poet who is expressing my feelings and ideas by choosing the data, training the models and selecting the starting words of each poem. Thereby, one could argue that any output of a language model meant to generate poetry is, in fact, poetry.

6 Conclusions and future work

One of the main goals of this project was to compare how LSTM and GRU models fare regarding poetry generation, as GRU models have rarely been used for this purpose in previous research. As detailed in Section 5, there seemed to be no significant differences in the evaluation of the poetry generated by the two types of RNNs. This may be due to the size of the study – there were only four study participants, and only one poem per network was presented to those participants. An interesting continuation of this project would therefore be to increase the number of study participants, and present more poems by each model. In theory, this would allow conclusions to be drawn regarding certain models (for example, if all poems from a specific model were rated similarly, the conclusion would be that there is a model specific problem/advantage).

Further research into the definition of poetry is also necessary for projects like this one. There is a certain value in intuition-based human judgements, as humans are the ones who enjoy poetry, typically without a definition of what poetry is. Thus, it would be interesting to examine what humans think poetry is. In turn, this could provide additional insights into the low kappa score of this study, and whether it is a result of poetry being subjective or the task being ill-defined. Furthermore, examining the human interpretation of poetry would aid in developing a standard for obtaining human judgements of language model generated poetry (e.g., whether or not it is beneficial to provide the evaluators with a definition of poetry).

References

Roger Bakeman, Duncan McArthur, Vicenç Quera, and Byron F. Robinson. 1997. Detecting sequential patterns and determining their reliability with fallible observers. *Psychological Methods*, 2(4):357–370.

Md. Kalim Amzad Chy, Md. Abdur Rahman, Abdul Kadar Muhammad Masum, Shayhan Ameen Chowdhury, Md. Golam Robiul Alam, and Md. Shahidul Islam Khan. 2020. Bengali poem generation using deep learning approach. In *Intelligent Computing Paradigm and Cutting-edge Technologies*, pages 148–157, Cham. Springer International Publishing.

Angela Fan, Mike Lewis, and Yann N. Dauphin. 2018. Hierarchical neural story generation. *CoRR*, abs/1805.04833.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378–382.

Giorgio Franceschelli and Mirco Musolesi. 2021. "creativity and machine learning: A survey". *CoRR*, abs/2104.02726.

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

Hugo Gonçalo Oliveira. 2017. A survey on intelligent poetry generation: Languages, features, techniques, reutilisation and evaluation. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 11–20, Santiago de Compostela, Spain. Association for Computational Linguistics.

Jack Hopkins and Douwe Kiela. 2017. Automatically Generating Rhythmic Verse with Neural Networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 168–178, Vancouver, Canada. Association for Computational Linguistics.

Ankit Kumar. 2021. Bidirectional lstm networks for poetry generation in hindi. *International Journal of Innovative Science and Research Technology*, 6(8):885–888.

J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Jey Han Lau, Trevor Cohn, Timothy Baldwin, Julian Brooke, and Adam Hammond. 2018. 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, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1948–1958. Association for Computational Linguistics.

Ruli Manurung, Graeme Ritchie, and Henry Thompson. 2012. Using genetic algorithms to create meaningful poetic text. *Journal of Experimental & Theoretical Artificial Intelligence*, 24(1):43–64.

Ivona Milanova, Ksenija Sarvanoska, Viktor Srbinoski, and Hristijan Gjoreski. 2019. Automatic Text Generation in Macedonian Using Recurrent Neural Networks. In *ICT Innovations 2019. Big Data Processing and Mining*, pages 1–12, Cham. Springer International Publishing.

Shakeeb A. M. Mukhtar and Pushkar S. Joglekar. 2021. Urdu & Hindi Poetry Generation using Neural Networks.

Christopher Strachey. 1954. The "Thinking" Machine. *Encounter*, page 25–31.

Zhe Wang, Wei He, Hua Wu, Haiyang Wu, Wei Li, Haifeng Wang, and Enhong Chen. 2016. Chinese Poetry Generation with Planning based Neural Network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1051–1060, Osaka, Japan. The COLING 2016 Organizing Committee.

Stanley Xie, Ruchir Rastogi, and Max Chang. 2017. Deep Poetry : Word-Level and Character-Level Language Models for Shakespearean Sonnet Generation. Technical report, Stanford University.

Rui Yan. 2016. I, Poet: Automatic Poetry Composition through Recurrent Neural Networks with Iterative Polishing Schema. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 2238–2244. AAAI Press.

Rui Yan, Cheng-Te Li, Xiaohua Hu, and Ming Zhang. 2016. Chinese Couplet Generation with Neural Network Structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2347–2357, Berlin, Germany. Association for Computational Linguistics.

Xiaoyuan Yi, Ruoyu Li, and Maosong Sun. 2017. Generating chinese classical poems with rnn encoder-decoder. In *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, pages 211–223, Cham. Springer International Publishing.

Eun-Soon You, Soohwan Kang, and O Su-Yeon. 2020. Automated korean poetry generation using lstm autoencoder. In *CEUR Workshop Proceedings*, volume 2653, pages 3–10.

Xingxing Zhang and Mirella Lapata. 2014. Chinese Poetry Generation with Recurrent Neural Networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 670–680, Doha, Qatar. Association for Computational Linguistics.

Andrea Zugarini, Stefano Melacci, and Marco Maggini. 2019. Neural poetry: Learning to generate poems using syllables. In *Artificial Neural Networks and Machine Learning – ICANN 2019: Text and Time Series*, pages 313–325, Cham. Springer International Publishing.

Appendix A Survey instructions

In this evaluation, you will be presented with 18 short texts. Your task is to use your intuition to determine the answer to the question ”Is this poetry?” for each of the texts. Each text is cut off at 100 words/punctuation marks.

Appendix B Discarded poems

the and , and . and , , the , , . , the and . , the , the the . , . of the , and of , the of the and , the , , of the and of the the the the of and , the , the , and the , , the and of , . , of . , , the the . , , , the , . the . . the the the the the . the of . of of and . , the . , , of . and of ,

Poem 1: Discarded poem, generated by model *m7lstm*.

the and the of of , , . . the the of of the the . , , the the , and and . the and , . the of . and , , , , and , , , the , the and and , the and . , of the of the . . , and and , . and and , the of of , , , , the of . and , the . , the . . , . of . . the the , the . the and the . of . the and and .

Poem 2: Discareded poem, generated by model *m7gru*.

Appendix C Poems in survey, in order of appearance

the world in the air . the wind is a thousand , in the air , and i see the way . i have been in the air to my own hands , and i know it . but i have been lost in my heart to be in your eyes and your heart and i will you know . the wind is in my hand , the wind , and it is the light , and the sky , i have been a little little woman , a woman , the old woman was a little girl , i am in

Poem 3: Poem generated by model *m5gru*.

the world has no way of the time of the night of the night . the sun . the sky is not to be a little , and i was a child , and the man who is it , the last of you . you will not be . but you are it is the same . the day is a little thing , a man and the same , i know the way , but the last of the sun has been in a dream , the one of the sun , i have not to the same , a

Poem 4: Poem generated by model *m1lstm*.

the sun is , and the way the world is not , and i am no more in a long . the world is a man , and the sun of the world . i know it was in the world of the world , and i am not . i know you , the man , the way of my heart and you can see the world . i am not . i will not know the world , and the first , a moment , in the night of the world , and in the air , the sun ,

Poem 5: Poem generated by model *m2lstm*.

the world is a man who was a little way . the first man is the same . but i have a man in the way . the wind , my mother said to the the the night , i was not afraid . i am no one , the same , i know the time . but i have been the way , and my heart was the same . i am a man , i am a man in my heart . my father said , i know . i will be . i was , but the old world

Poem 6: Poem generated by model *m6gru*.

the end recited: the same to the same one . i will not see it , that i know , i have been in the world , but the same . i know that i had to say , and i have been the world to the other side to be , the way of the world , i know , i have been the time of the end of a dream , to the world of the sea , the way of the world of my heart . you were a thousand times i have a good , for a man

Poem 7: Poem generated by model *m5lstm*.

the first of the world . the king of the world is the element , and the proudest insistence , the whirligig of the earth . the world was the first time . it was a way in the air . i saw my father . i was the old woman , the first day . it was a time . it seemed to be a little one . it is not . . . . and the only time i am , the first , and the lord , the splendid , that , and the

Poem 8: Poem generated by model *m4gru*.

the first time we are not to know , i am a long , and i have been in the sun and i have a good . but i am the same . you were in the house . i am not the way to the world , a little thing , and the other in the sun . i have a little , i know .) i was not the way to my own house , a little , in a field of the water , and the night , the wind , i am a long , i have

Poem 9: Poem generated by model *m1gru*.

the weather is hot on the back of my watch which is down at finkelstein's who is gifted with 3 balls but no heart , but you've got to understand when the bull goes down on the whore , the heart is laid aside for something else , and let's not over-rate the obvious decency for in a crap game you may be cutting down some wobbly king of 6 kids and a hemorrhoid butt on his last unemployment check , and who is to say the rose is greater than the thorn ? not i , henry , and

Poem 10: Poem by Charles Bukowski, cut off at 100 tokens.

the room where it was a man who had been , and he had no one . i have not seen the world to the ground , in that old world . . . i am in a new york city dreamscape , i had a child of the dead , and i have seen my eyes . the last time to the hearer's of the day . and when you were born . i have been a woman in the air . i will have been the same . i am a woman . i was a little woman with her

Poem 11: Poem generated by model *m4lstm*.

the air . the night . i am a man , i know , to say you have you to do you , and the time i have been in your pocket . you have to be the way . the sun is a little bit of the sky . a little bit of my heart . it was the world , and i will see my own . i have seen a woman with a little , i have to be the end of a man . a woman is the way to be , or a man . a man

Poem 12: Poem generated by model *m3lstm*.

the hinge of the year the great gates opening and then slowly slowly closing on us . i always imagine those gates hanging over the ocean fiery over the stone grey waters of evening . we cast what we must change about ourselves onto the waters flowing to the sea . the sins , errors , bad habits , whatever you call them , dissolve . when i was little i cried out i ! i ! i ! i want , i want . older , i feel less important , a worker bee in the hive of history

Poem 13: Poem by Marge Piercy, cut off at 100 tokens.

the way . we have been the problem , a time . the water is a small . a little child . i am not afraid . the world is a new century , a little girl and the little . and they have been a man . they are not a man , and he is the world of a man and who would have to be a woman , and the old men are not not not the way . a woman , who was not a boy and a man , he says . he is his wife ,

Poem 14: Poem generated by *m8gru*.

the world , and all the men of the world . the sky was not a man to do . the sun is my life , and the sun was the world of my heart , a woman , a woman . we have been a boy , the sky is the world to do with the world , the old ones are the dead ? i was born . i am not the first of my life , the sun and a white washstand . i know it , the world . i have no more , i have been a

Poem 15: Poem generated by model *m8lstm*.

the evening empty as a convex coconut split down the seam: not that it can be filled . the evening empty as a gourd that twists on an iron thread: the rough skin of the sphere . . . . . . not that there was a spoken word to recall the moment of seeing the short span when the clocks ceased to revolve and hands met in jest or benediction time of the vortex into which hibiscus and almond trees strayed and windows made of aluminum . the stars are suddenly remote candescent petals night throws above the yard

Poem 16: Poem by Mark McMorris, cut off at 100 tokens.

the sun was the first day , the sky , a man , the man and the sky , he is the world , the sun , the hand of a tree and the night , the sun , and a woman who is the only thing to be , but i have seen my own . the moon of your face and my heart , a woman in a room , the air is the night of the sea , i am a child , i know you were a woman . i know . the old woman , i am

Poem 17: Poem generated by model *m3gru*.

the end of this poem of my own love , the world is a man , a few of the same , i have to be a little , that is the time i have been the same home . i am not the first , that you have to be a little thing to the way . the sun is the other . you are a way to the end of the world , and the first thing i have been the same , i have been . i am not , i have a little , but i could tell

Poem 18: Poem generated by model *m6lstm*.

the same . i was the same of the time , the little girl was a man . i am not the same thing . i will be a little . you were the thing . i was not the thing . the sun is a man , the first , a woman who would be , the way , the world , a little and the man is to the world . and the sun , the dark of his own , a man , a little and a woman , and the way of a man and a woman is

Poem 19: Poem generated by model *m2gru*.

the car had a cover over it and it was over the wheels and it hurt my ass and i couldn't sleep . it seems i should move , go forward now i was wandering through the jungle anywhere on earth but i was a woman in bed in new york and how many people have died in wild places dreaming you were still in bed would you know . travel well i said to my dog when she went on her journey thinking of a cheap movie i've thought this was an urn turning this was on water this

Poem 20: Poem by Eileen Myles, cut off at 100 tokens.