Haiku Generation
A Transformer Based Approach
With Lots Of Control

Giacomo Miceli
haiku@jamez.it

Abstract

Haiku generation has attracted the interest of the NLP community for decades. After researching different ways of imbuing language models with a poetic sense, we introduce *Haikoo*, a transformer-based model that outperforms previous state-of-the-art neural network-based haiku poetry generators. *Haikoo* consists mainly of two pieces: a GPT-2 model, which we fine-tuned on haiku poems, and a Plug and Play Language Models, which we employ to control the generated results to a further extent than the classic prompt approach. We found that GPT-2 learns successfully to generalize many of the qualities of haiku poetry while retaining enough flexibility to compose poems on entities never seen on the training data. PPLM on the other hand acts as a rudder, moving the generated output towards a specific concept or word. *Haikoo* helps to search the space of haiku poems satisfying poetic constraints, producing results that make sense to most readers, and that range from lyrical to hilarious.

No prompt, no PPLM-BoW
summer solstice fragrance of lavender blossoms from the purple hills
No prompt, PPLM-BoW: "poetry"
what is poetry a mockingbird sings a song from a foreign tongue
Prompt: "natural language", PPLM-BoW: "understanding"
natural language a hint of understanding the way an infant talks

Table 1: Output samples of *Haikoo*.

1 Introduction

Haiku is a type of short poem originally from Japan with an exceptionally well-defined structure. Traditional compositions consist of three phrases following a 5, 7, 5 *on* pattern (a type of Japanese phoneme often translated as "syllable"), typically contain a *kigo*, or seasonal reference, and often they have a *kireji*, or caesura also known as metrical pause or break in a verse. Modern and especially English language haikus do not follow the 5-7-5 pattern very strictly, but usually adhere to a short-long-short form of around 10/12 words in total.

The work of a poet consists of describing the human experience and associated feelings using a very specific written style. *Haikoo* relies on language models trained on an Internet-corpus, corresponding to a level "WS2" using the World Scope definition provided by (Bisk et al., 2020), meaning that the model has no experience in "extralinguistic events", particularly pertaining to the domain of perception and social interaction. As such, we are not concerned that our method will replace poets anytime soon.

Our goal is instead twofold. On the one hand, we hope to make haiku poetry even more popular. By releasing this model, we wish to see it integrated into places where it can raise awareness of this ancient form of poetry that is so befitting of the Twitter generation. On the other hand, we recognize that writing good poetry is an arduous task and that *Haikoo* can be a useful tool even for the amateur poet, helping to produce interesting poems on a given subject.

Table 1 shows some of our model's capability. Refer to Appendices A for further examples.

2 Related work

A thorough investigation on the topic of poetry generation should start at the roots of creativity itself. Boden (1998) distinguishes between three different kinds of creativity: combinational, exploratory, and transformational. He argues that exploratory creativity is most commonly found in computer models, and indeed that's the case for most poetry-generation tools including *Haikoo*. But it's the combinational type of creativity that most resonated with our research, motivating the use of a two-fold way of controlling the output, something that implicitly allows for comparisons and interesting mixes to arise.

Netzer et al. (2009) showed haiku poem generation using Word Association Norms, a hand-made corpus collecting cue words and free associations for any given term. While this approach falls short of consistently producing haiku-related features, we found their analysis on the complexity of evaluating poetry very inspiring.

In Tosa et al. (2009) we see the first strong attempt to drive the output of a generated poem using phrases based on user input. While using just a handcrafted database to assemble poems, it remains one of the highest-cited papers on the topic.

In Xian-chao et al. (2017) we find one of the earliest comparisons of different machine learning approaches, focusing on RNN, RNN-LSTM, RCNN, and SeqGAN for their experiments. Their sensible usage of Perplexities to measure their results prompted us to use that method as one of the two metrics to compare *Haikoo* with a baseline. The second method we used can be partially credited to Rzepka and Araki (2015) and consists of polling human experts to evaluate a set of haikus based on certain dimensions.

Wang et al. (2021) is the only work in our research to discuss poetry generation based on a Transformer model, although their focus on Limerick poem generation introduces further complexity in the form of a rigid rhyming scheme and requires a remarkable feat of engineering to produce viable results.

One aspect that emerged from several related works (particularly the ones focusing on English-based haiku generation) is the acceptance of a less "constrained" definition of what constitutes the metrical subdivision of a haiku. A definition by author Kerouac (1971) proposes "that the Western Haiku simply say a lot in three short lines in any Western Language". Although *Haikoo* exhibits a consistent approximation of the 5-7-5 syllabic structure, we decided to not impose any strict check on the length of the output.

Project Gutenberg
It is raining so long that moss and lichens just grow upon the heart.
Tempslibres
stuck in traffic dreaming of the monsoon in a far-away land
Ballas
heat spike a blister beetle probes the cactus flower

Table 2: Samples from the three datasets.

3 Data

Experiments relied on a corpus of 14,000 haiku poems written in English by experienced and amateur poets alike. We used three main sources combined. The largest and arguably the best is Project Gutenberg, providing roughly 5,500 samples from published poets. The second in terms of size is Tempslibres, an online international haiku community that provided 5,200 haiku from both expert and novel writers. The remainder of the dataset comes from an online collection (Ballas) which contains poems with a large variance in quality, coming from smaller sets such as `haikupoet.com`. Table 2 provides samples from each collection, while table 3 shows a POS analysis on the three datasets, displaying a reasonable uniformity between them.

A fourth dataset we considered in early stages is twaiku, a collection of haikus posted on Twitter. The quality of those is rather poor, in part as a result of the lower-barrier to publishing and more facetious approach to poetry, which ends up skewing the dataset with "poems" that respect the metric, but are often missing kigo and kireji and frequently contain profanity. This dataset was not included in our final corpus.

	Guttenberg	Tempslibres	Ballas
Vocabulary size	1440	2274	1916
Nouns (%)	28.9	31.6	32
Determiners (%)	10.2	13.9	14.2
Prep. & Conj. (%)	11	11.9	11.3
Adjectives (%)	8.8	9.6	9.6

Table 3: Quantitative summaries of datasets.

4 Methods

While researching related work, the two most convincing neural-based haiku generation approaches used Generative Adversarial Networks, an architecture first described in Goodfellow et al. (2014). Encouraged by the results of Hong (2019), we use a vanilla LeakGAN model as a baseline. Relying on TextGAN (Lam, 2019) for architectural implementation, we trained it for 10 epochs. Our training set comprised 90% of the poems from each data source, leaving 10% for the test set. Training required over 7 hours on a 1080 Ti GPU, at which point any gain on self-BLEU (one of the internal metrics of TextGAN) seemed to have tapered-off.

GPT-2 (Radford et al., 2018) gives access to powerful world knowledge that can be harvested to generate text in many styles. For open-ended text generation tasks, GPT-2 benefits from fine-tuning with domain-specific data to improve the relevance and quality of the predicted text. Using the tools provided by Wolf et al. (2019) we fine-tuned the model, following the same ratio of training/test split as with the LeakGAN model. We loaded the default configuration of a pretrained GPT-2 model comprising 12-layer, 768-hidden, 12-heads, and 117M parameters. After training for 4 epochs, which remarkably took only 5 minutes on our configuration, and using AdamW as optimizer, our basic model was ready.

While the language model can be conditioned with a prompt by specifying the first word(s) of a poem, we looked into other methods to further control the output. At first, we tried the approach described in Krause et al. (2020), but we quickly found that GeDi needs to be specifically pre-trained on each topic, and did not yield zero-shot results on most topics. We also realized that in some cases a poet might not be interested in the general "texture" of a composition, but in a specific word. All of that led us to the Plug and Play Language Models presented by Dathathri et al. (2019). PPLM provides a decoding scheme to guide the generation process using bag-of-words or classifiers, without re-training the model and instead relying on gradient descent in the latent space of the original Transformer model. The bag-of-words approach, in particular, allows for highly focused control over the output.

To maximize the odds that a certain topic or concept will be covered by the generated haiku, we provide utilities to automatically generate bag-of-words from one word, using WordNet and word2vec to find related tokens. But the flexibility of PPML allows one to specify even a single word *as* the bag-of-words and the model will, most of the time, generate meaningful text with it. PPLM required minimal adaptation to work with our fine-tuned GPT-2 and most of the tweaking required is on the stepsize value used in the normalization of the gradients. We find that the stepsize should be inversely proportional to the frequency of the selected word(s) in a given language, for instance using `stepsize=0.03` for a common word, or `stepsize=0.15` for an uncommon word.

5 Results

We performed two distinct analyses to compare the baseline with *Haikoo*. First, we measured Bigram Perplexity on a sample output of both models and found that our proposed model fares considerably better than the GAN approach, as seen in table 4. We did not take that for granted, considering that by accessing a larger vocabulary than what constituted our corpus of haiku, the GPT-2 based model could have potentially thrown off the perplexity evaluation.

Perplexity by itself doesn't say much on matters of poetry. Elements that a good scoring system would take into account are: topic classification, meter analysis, rhyme detection, diction analysis, and metaphor detection (Kesarwani, 2018). Because such an automated scoring system does not exist yet, we used Amazon Mechanical Turk to rate the output of our models.

Model	Perplexity
Baseline	2079
Proposed	**1185**

Table 4: Perplexities for the models.

	Baseline mean	sd
Sensicality	3.33	0.14
Wisdom	3.57	0.13
Overall quality	3.44	0.14
	Proposed mean	**sd**
Sensicality	**4.02**	0.11
Wisdom	**4.07**	0.09
Overall quality	**3.96**	0.13

Table 5: AMT evaluation.

AMT offers the possibility of specifying additional qualifications that Workers must meet to work on a given task. We use "Blogger" as a proxy for familiarity with the written word and, more hopefully, a general appreciation for poetry. Poems generated by both baseline and *Haikoo* were evaluated over 200 times on 3 metrics, asking Workers to rate on a scale of 1 to 5: the "Sensicality" of the poem, or how much sense it made; the Wisdom or conversely the stupidity of a poem; and lastly a general Overall Quality assessment. Table 5 shows the results of AMT evaluation. On each metric, our model outperforms the baseline with a particularly high margin in the metric of "Sensicality".

6 Conclusion

Haikoo is a model that can be trained quickly, runs efficiently on common architectures, and generates compelling poems that amuse and inspire. It can be used as a fully-autonomous system, or it can be guided flexibly in two different ways.

Some of the possible future embodiments of this work include: an accessible interactive web-page; a Twitter-bot; a pipeline to generate poems based on images, using neural image caption generation; or conversely, a pipeline to generate *Haiga*, a style of Japanese painting that incorporates the aesthetics of haiku poetry, using *Haikoo* as a seed for the text-to-image generation.

We hope our work can help to spark a multidisciplinary conversation about the role of neural-based techniques in the literary arts and more generally, among creative specialists.

References

Sam Ballas. Generating poetry with poetrnn.

Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, Nicolas Pinto, and Joseph Turian. 2020. Experience grounds language.

Margaret A. Boden. 1998. Creativity and artificial intelligence. *Artificial Intelligence*, 103(1):347–356. Artificial Intelligence 40 years later.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2019. Plug and play language models: A simple approach to controlled text generation. *CoRR*, abs/1912.02164.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial networks.

Lakshay Gopalka Ziwei Lin Ran Jing Hanwen Hong. 2019. Generating haiku using generative adversarial network.

Jack Kerouac. 1971. *Scattered Poems*. City Lights Books, San Francisco.

Vaibhav Kesarwani. 2018. Automatic poetry classification using natural language processing.

Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, Richard Socher, and Nazneen Fatema Rajani. 2020. Gedi: Generative discriminator guided sequence generation.

William Lam. 2019. Textgan-pytorch. `https://github.com/williamSYSU/TextGAN-PyTorch`.

Y. Netzer, D. Gabay, Y. Goldberg, and Michael Elhadad. 2009. Gaiku : Generating haiku with word associations norms.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2018. Language models are unsupervised multitask learners.

Rafal Rzepka and Kenji Araki. 2015. Haiku generator that reads blogs and illustrates them with sounds and images.

Naoko Tosa, Hideto Obara, and Michihiko Minoh. 2009. Hitch haiku: An interactive supporting system for composing haiku poem. In *Entertainment Computing - ICEC 2008*, pages 209–216, Berlin, Heidelberg. Springer Berlin Heidelberg.

Jianyou Wang, Xiaoxuan Zhang, Yuren Zhou, Christopher Suh, and Cynthia Rudin. 2021. There once was a really bad poet, it was automated but you didn't know it.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Wu Xian-chao, K. Momo, Ito Kazushige, and C. Zhan. 2017. Haiku generation using deep neural networks.

A Additional samples from *Haikoo* (cherry-picked from batches of 10)

No prompt | No BoW
first fireflies
in the night the old dog and I
fight fire with fire
~

a little Buddha
reflected in the puddles–
summer rain festival
~

winter solstice
a cloud of stars rises
from the desert valley

Prompt: "big ship stuck" | BoW: "canal"
big ship stuck in the mud
the heat gets hotter today
in the canal
~

big ship stuck
in the middle of nowhere
wind gusts
~

big ship stuck
the sound of the waves
through my ear canal

Prompt: "a tree" | BoW: climate change terms
a tree without leaves
the smell of ozone
and warming air
~

a tree stump shadow
in the winter greenhouse
a warm breeze

No prompt | BoW: "escalation"
the old man's eyes
still on the battlefields map
an escalation
~

No prompt | BoW: volcanology terms
the old pond
in a giant steaming bubble
the lava flows freely
~

a new year's day
the smell of sulfur around me
redemption of the soul

No prompt | BoW: Fantasy terms
first raindrops
a child draws the dragon's eye
the scent of middle-earth
~

unicorn garden
i've never dreamed of becoming
the game's hero

Prompt: "dry" | BoW: "sanctuary"
dry creek
a white buffalo's silhouette
by the animal sanctuary

Prompt: "pandemic" | BoW: "vaccine"
pandemic heat –
a man with a feverish cough
syrup arrives
~

pandemic craze
another airborne disorder
called antivaccine
~

pandemic fever -
the only way to be aseptic
is to be dead

Prompt: "consciousness" | No BoW
consciousness to smell
the sun-drenched forest
that never was

No prompt | BoW: apohenia terms
patterns everywhere
i search for new ways
in the morning light
~

seeking patterns
searching for the right word
at the right time