Möbius: Exploring A New Modality For Poetry Generation

Mattt Thompson

April 27, 2009

Abstract

This paper describes the design and implementation of a computational poetry engine. Approaching the task of poetry generation in terms of a dichotomy between form and matter, this framework aims to more authentically model the aesthetic process of writing poetry. Incorporating a range of theoretical traditions, the system is able to produce technically valid and reasonable Haiku. The modularity of its design positions the system as a viable framework for the task of poetry generation.

1 Introduction

Haikus can be fun, / But sometimes they don't make sense. / Refrigerator

Writing poetry is hard enough for humans to do, let alone computers. Among the difficulties described by Manurung, et al., (2000) is the need to satisfy various phonetic, syntactic, and semantic constraints. Yet even with an immense amount of information to pull from, traditional approaches of natural language generation still seem to break down. Normally, NLG tasks can be divided into stages of content determination, text planning, and surface realization (Reiter, 1994). NLG systems begin with a communicative intent, some meaning to convey. We cannot assume the same for poetry, however. As Levin (1962) explains "In poetry the form of the discourse and its meaning are fused into a higher unity". When writing poetry, surface realization informs and depends upon content determination and text planning. This linear process in conventional NLG becomes circular for poetry. Sharples (1962) relates the writing process to a creative process, composed of cycles of analysis, *reflection*, and engagement, *synthesis*. Thus, we might better understand creativity as an evolutionary process driven by an introspective feedback loop.

There is also the question of what poetry is, exactly. This question has been actively debated among poets and philosophers throughout the ages, into the modern day. In his *Poetics*, Aristotle writes of the role of genre and basic elements in poetry. With the rise of poetic forms, the definition of poetry became conflated with requirements of structure. Today, poetry remains highly influenced by free verse poets like Ezra Pound, who abandoned those traditional forms in order to focus on the rhythmic aspects of language. In this paper, we adopt the minimal notion that, as a necessary condition, poetry is divided into lines, or verses (Fabb & Halle, 2008). This convention allows us to reconcile the great breadth of poetic tradition while maintaining Levin's (1962) notion of poetry as a fusing of form and meaning.

For this project, we explore a new modality in poetry generation, to see if we can create a framework that more authentically models the aesthetic process of writing poetry. Our modality was originally inspired by the classical tradition of Aristotle, who describes the dichotomy between form and matter. We extend this notion to describe a poem as the interplay between poetic form and the linguistic units within it. Poetic form, in this sense, subsumes the Chomskyan notion of grammar, by imposing various constraints on subjects and themes (semantics), structure (syntax), along with meter, prosody, and rhyme (phonology). Different poetic forms are distinguished by the variation and extent of these constraints. Matter can be understood as the linguistic units of poetry that encode particular thoughts or feelings. These units can range from individual phonetic features to larger syntactic structures and elements of discourse.

This form / matter dichotomy extends further, into how these two aspects can be modeled computationally. That is, form may be best modeled by a generative approach, which contrasts the apparent optimality problem of modeling matter. A particular poetic form imposes a set of rules that can be encoded into a formal grammar. As such, the process of transforming poetic form into linguistic rules is top-down, producing data from theory. Conversely, matter must constrain a large set of data in a bottom-up manner. A poet wishing to capture a particular thought or feeling in text has the entire expressive repertoire of language to choose from. Deciding on what to write is a process of narrowing down candidates until only the best remains.

§2 describes the implementation of the generative poetry system. Preliminary results are provided in §3. Finally, we discuss system design and details of its implementation in (§4), and conclude (§5)

2 Implementation

For our implementation, we chose to model the poetic form of Haiku. As a form with well-specified rules for meter, syntax, and semantics, it presents an accessible, yet non-trivial problem. Haiku traces its origin to 17th Century Japan, where the poet Matsuo Bashō gave rise to the popularity of *haikai*, a genre of various forms that appealed to the "poetic spirit".

Modern Haiku was given its name and familiar 5/7/5 form by Masaoka Shiki two centuries later. Haiku is characterized by its economy of words and ideas. As such, functional or syntactic words are often omitted because of the form's terse nature. Verbs normally occur in the active voice and present tense. Traditionally, Haiku also incorporate season word, or *kigo*, and a verbal caesura, or *kireji*.

2.1 Form

We formally define the structure of a Haiku to be comprised of 17 syllables distributed over metrical verses of 5/7/5. We start with this set of rules and construct a formal grammar to encode them accordingly.

`haiku`	→	`stanza5 stanza7 stanza5`
`stanza5`	→	`ADJ-2 PP-1 DET-1 N-1`
`stanza5`	→	`DET N-2 PP-1 DET-1 N-1`
`stanza7`	→	`V-2 PP-1 DET-1 ADJ-2 N-3`
`stanza7`	→	`PP-2 DET-1 N-2 V-2 N-2`

Figure 1: Sample PCFG Transformation Rules: End-state tokens encode part of speech and syllable requirements, e.g. `V-2` is a verb with 2 syllables.

Grammar

A probabilistic context-free grammar, or PCFG, is used to model syntactic constraints imposed by the Haiku poetic form. PCFG lends itself well to this task, as it can directly encode constraints on meter and part of speech, while remaining transparent and accessible for manual tuning of rules.

In the process of implementing this system, we developed two PCFG rule-sets using two very different methods. The first grammar was constructed by hand according to the form of traditional Japanese Haiku. In this grammar, the Haiku is characterized by a punctuation break in the poem, which signals a shift in tone. The contrast between the first and second halves are used to shed insight about the theme as a whole.

The second grammar took a different approach, using traditional NLP tools to computationally derive the salient structures of real Haiku. We compiled a corpus of amateur and published western-style Haiku, as well as translations of traditional Japanese Haiku. From this corpus, we used the Stanford Part-Of-Speech Tagger (Toutanova, Manning, et al., 2003) to derive typical grammatical structures for each individual line.

2.2 Matter

We by considering a large set potential candidate words to be used in the poem. For each position in the generated grammar, we determine the optimal candidate from that set by ranking on a series of constraints. Each constraint corresponds to a particular computational module. For our implementation, we created modules to impose constraints on **semantics**, **syntax**, **meter** and **phonology**, which corresponds to constraints of the grammar. Conventional constraints like markedness against duplicate words are integrated into the constraint ranking logic.

	{`flower`, `2 syl`, `noun`}	*redundant	Ident-IO meaning	*form	*syllables	Ident-IO phonetics
	[spring]				*	
☞	[petal]					
	[concrete]		*			
	[blooming]			*		
	[beautiful]			*	*	

Figure 2: Sample Optimality Theory Tableaux For Constraints on Position in Grammar

Semantics

We use the user-inputted theme word to generate a large set of potential candidates. Using word association norms, or WAN, we determine which words are semantically related. A directed graph, built on the Edinburgh Associative Thesaurus (STFC, 1994), maps cue words and their responses; edges are weighted by the strength of their semantic association. WANs have been developed over several decades by psychologists. These word associations were collected from subjects, who were given a cue word and asked to list words that immediately came to mind.

We start with the user-supplied theme word, and all other adjacent words in the graph. In order to ensure a sufficiently large and interesting set of candidates, a random walk is taken on a subset of those adjacent words. We heuristically determined a balance between candidate set cardinality, coherency, and computational branching factors through successive trials. In its current implementation, the system takes 2 steps on 2 randomly-chosen neighbors.

	Edinburgh Associative Thesaurus	WordNet
`bird`	Blond, Wing, Flight, Bardot, Table, Eagle, Starling, Pheasant, Eye, Flies, Cage	Bird Of Passage, Observe, Nester, Dickey Bird, Wildfowl, Yell, Archaeornis, Ratite Bird, Missy, Dickeybird, Craniate
`bird`	Blond, Wing, Flight, Bardot, Table, Eagle, Starling, Pheasant, Eye, Flies, Cage	Bird Of Passage, Observe, Nester, Dickey Bird, Wildfowl, Yell, Archaeornis, Ratite Bird, Missy, Dickeybird, Craniate
`feather`	Pillow, Edge, Bird, Pen, Fear, Beds, Fall, Cap, Tickle, Thin, Cuckoo	Body Covering, Scapular, Rotary Motion, Acquire, Flight Feather, Animal Material, Conjoin, Quill Feather, Rotation, Contour Feather, Spurious Wing
`soar`	Vet, Climb, High, Air, Heaven, Glider, Blood, Swift, Score, Wings, Plane	Come Up, Climb, Wallow, Ascension, Wing, Go Up, Uprise, Arise, Rise, Ascent, Move Up
`graceful`	Runner, Dance, Body, Smooth, Delightful, Lady, Delicate, Fall, Apollo, Queen, Beauty	Willowy, Lissom, Svelte, Lissome, Sylphlike, Gracile, Lithesome, Elegant, Awkward, Smooth, Lithe

Figure 3: Comparison of Word Relations for the Edinburgh Associative Thesaurus and WordNet

Initially, the task of candidate generation was left to WordNet (Miller, 2006), which provides a formalized set of systematic relations that exist between words, like synonymy, metonymy, and antonymy. Although this is incredibly useful in other NLP applications, it was not particularly good at generating poetic word choices. Rather than constraining words within a small set of well-defined relations, good poetry relates words in unexpected ways that can be subtle, difficult to classify, and difficult to derive from a corpus (Netzer, et al., 2009).

Syntax

Each position in the grammar encodes an expected part of speech. Candidates that don't match these expectations are removed.

Candidates are queried in WordNet, which maps word senses to parts of speech. Because words often have multiple senses, this query can return a number of different parts of speech. A word qualifies as a particular part-of-speech if it is included in the list of possible ones across different word senses.

Meter

Each position in the grammar encodes an expected number of syllables. As before, candidates that don't match these expectations are removed.

We use the CMU Pronouncing Dictionary (Carnegie Mellon University, 1995) to calculate the number of syllables for each candidate. The dictionary provides a correspondence between the orthographic representations of each candidate and its corresponding phonetic string. A simple calculation on stressed vowels in the phoneme string provides a syllable count for that word.

Phonology

In order to determine a best candidate, we maximize for phonological complexity (and harmony) using a Phonology module created for this project. Rather than a hard constraint like all of the previous ones, this functions more as a ranking, much like a faithfulness consideration in Optimality Theory (Prince, et al., 2004).

We refer back to the CMU Pronouncing Dictionary for a word's phonetic string. This phonetic representation is enhanced by a consideration of feature analysis. Following the tradition of Jakobson (1952), phonemes are thought of in terms of feature bundles that describe acoustic characteristics such as voicing, rounding, continuation, and sonorance. Phonological complexity is thus defined by a set of hand-weighted rules that essentially looks at how different adjacent phonemes are.

3 Results

Refer to Appendix A for a sample of selected poems generated by this system.

In our preliminary tests, we generated 10 poems for each of 10 themes:

{`bird, dance, dog, flower, joy, rain, red, sickness, spring, steel` }

Themes were selected based on salient features of the environment at the time, with an effort to include a wide range of different types of words. Each of the generated poems was checked for correctness of form (i.e. 5/7/5) and qualitatively analyzed in terms of how reasonable the poem seemed as far as imagery and convention.

Of the 100 generated poems, 91 conformed to Haiku structure, while 3 others were questionable, based on the pronunciation of certain bimoraeic syllables caused by word-final syllabic [l] (e.g. *real* and *guile* could be considered mono- or bi-syllabic). Cases where incorrect structure were produced were a results of word omission at a position where no suitable candidates were found. This is most likely to occur on positions that required a syllable count of 4 or 5.

Generated Haiku - `joy`

high euphoria
incredible lucky sex
feathered summer love

Generated Haiku - `sickness`

calm vegetable grey
a light stick better for brown
wallpaper vomit

Generated Haiku - `dog`

cannabis station
a heavy blossom smoke weed
delicate kettle

4 Discussion

4.1 Design

One of the primary design goals of this project was modularity, such that any individual component could be readily swapped out for something better suited to a particular task. Fortunately, the nature of the form / matter modality lends itself naturally to this.

At its top level, the system operates in parallel between two separate, complementary components – one for form and the other for matter. That is, the process of one does not assume to affect the other until they converge to produce the final output. Within each component, there can be any number of functional modules. For instance, the matter component contains modules relating to semantics, syntax, meter, and phonology.

In the course of developing this project, a number of modules were replaced by more effective candidates. For instance, the first semantic module generated candidates using WordNet with little success. It was replaced by an approach using word association norms, which produced more and better candidates.

4.2 Implementation

From the examples given in the Results section, we can see aspects in which the system was very successful, and some ways it could be improved. One of the strengths of the system was in consistently creating unexpected imagery. Especially in the poem generated about `sickness`, the imagery of *calm vegetable grey* and *wallpaper vomit* are visceral and familiar to the experience of being sick at home or in the hospital.

Orthographically, the output contains no capitalization or punctuation, aside from line breaks. In Haiku, punctuation is especially important in organizing thoughts and word-flow, and would be a primary concern for improving output.

Perhaps the most significant shortcoming of the system is lack of subject-verb agreement, as in the sequences *a light stick* and *blossom smoke weed*. This is primarily a shortcoming of the way we use a PCFG to generate the grammar. Rules in a PCFG are limited to their local scope, and cannot look back or ahead in a sequence. One solution would be to create a new module for Morphology that could conjugate word stems according to a Hidden Markov Model.

5 Conclusion

We presented a general, modular framework designed for poetry generation in terms of a generative process of constructing a grammar from poetic form, and the optimality process of narrowing candidate words according to constraints imposed by the form. Preliminary results demonstrated the system's ability to produce Haiku in technically valid structure. It's modularity of design allows for the system to be improved by incorporating various NLP techniques, or extended into poetry generation for other forms.

Appendix A: Example Output

cannabis station
a heavy blossom smoke weed
delicate kettle

animal pardon
nest watches the green table
sex worm wonder bird

high euphoria
incredible lucky sex
feathered summer love

light pot dog bright pink
introduce the long feline
artificial white

automatic inn
instrument dance stick made sharp
mechanical steel

calm vegetable grey
a light stick better for brown
wallpaper vomit

clean unhappy greens
america fur porridge
ugly man heavy

Appendix B: NLP Resources

CMU Pronouncing Dictionary

The CMU Pronouncing Dictionary is a public domain pronouncing dictionary created by Carnegie Mellon University. It is used as the American lexicon for the Festival Speech Synthesis System and also for the CMU Sphinx speech recogntion system. The latest release is 0.7a, which contains 133,746 entries (from 123,442 baseforms).

`http://www.speech.cs.cmu.edu/cgi-bin/cmudict`

Edinburgh Associative Thesaurus

The Edinburgh Associative Thesaurus (EAT) is a set of word association norms showing the counts of word association as collected from subjects. This is not a developed semantic network such as WordNet, but empirical association data.

`http://www.eat.rl.ac.uk/`

Stanford Part-Of-Speech Tagger

A Part-Of-Speech Tagger (POS Tagger) is a piece of software that reads text in some language and assigns parts of speech to each word (and other token), such as noun, verb, adjective, etc., although generally computational applications use more fine-grained POS tags like 'noun-plural'. This software is a Java implementation of the log-linear part-of-speech taggers.

`http://nlp.stanford.edu/software/tagger.shtml`

WordNet

WordNet is a large lexical database of English, developed under the direction of George A. Miller. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations.

`http://wordnet.princeton.edu/`

References

[1] Aristotle and H. Lawson-Tancred. *Metaphysics*. Penguin Books, London, 1998.

[2] Aristotle and K. McLeish. *Poetics*. Theatre Communications Group, New York, 1st u.s. ed edition, 1999.

[3] M. Boulton. *The Anatomy of Poetry*. Routledge Kegan Paul, London, 2nd ed., rev edition, 1982.

[4] N. Chomsky. *Syntactic Structures*, volume 4. Mouton, The Hague, 1968.

[5] N. Fabb, M. Halle, and C. Piera. *Meter in Poetry: A New Theory*. Cambridge University Press, Cambridge, UK, 2008.

[6] A. Finch and K. Varnes. *An Exaltation of Forms: Contemporary Poets Celebrate the Diversity of their Art*. University of Michigan Press, Ann Arbor, 2002.

[7] R. Jakobson. *Six Lectures on Sound and Meaning*. MIT Press, Cambridge, Mass., 1st mit press ed edition, 1978.

[8] R. Jakobson and L. R. Waugh. *The Sound Shape of Language*. Mouton de Gruyter, Berlin, 2nd ed edition, 1987.

[9] P. K. Jason. *Critical Survey of Poetry*. Salem Press, Pasadena, Calif., 2nd rev. ed. edition, 2003.

[10] S. R. Levin. Linguistic Structures in Poetry. 1962.

[11] H. Manurung and T. H. Ritchie, G. Towards A Computational Model of Poetry Generation. 2000.

[12] S. Mendick. The Associative Basis of the Creative Process. *Psychological Review*, 1969.

[13] Y. Netzer, D. Gabay, Y. Goldberg, and M. Elhadad. Gaiku : Generating Haiku with Word Associations Norms.

[14] A. Prince and P. Smolensky. *Optimality Theory: Constraint Interaction in Generative Grammar*. Blackwell Pub., Malden, MA, 2004.

[15] E. Reiter. Has a Concensus on NL Generation appeared? And is it Psycholinguistically plausible? In *Proceedings of the Seventh International Natural Language Generation Workshop*, 1994.

[16] H. Sato. *One Hundred Frogs: From Renga to Haiku to English*. Weatherhill, New York, 1st ed edition, 1983.

[17] M. Sharples. An Acount of Writing as Creative Design. In M. Levy and S. Ransdell, editors, *The Science of Writing: Theories, Methods, and Individual Differences and Applications*. 1996.

[18] N. Tosa, H. Obara, and M. Minoh. Hitch Haiku: An Interactive Supporting System for Composing Haiku Poem.

[19] M. Tsan Wong and A. Hon Wai Chun. Automatic Haiku Generation Using VSM. In *Proceedings of ACACOS'08*, April 2008.