

Poetry Framework: Recognize Generate and UNDERSTAND Poetry

Greg Cobb Sonal Danak Arvind Krishnaa Jagannathan Michelle Scott

Computational Creativity/ Modeling and Design Deliverable 1

Problem Characterization

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” in the Merriam-Webster online dictionary. The use of rhyme and rhythm separates poetry from traditional prose.

It is possible that great poets like William Shakespeare and Robert Frost did not spend a lot of time focusing on the rhyme or structure that they planned to use. Instead the words might have come to them in a flow [1]. However, unlike such great poets who have produced large bodies of work, the novice poet, or a poetry enthusiast may not get this natural flow. We hope to come up with a tool which can assist such people in generating quality verse; the tool needs to be able to “understand” what a poem really is and be able to discern poetry from prose. Although the tool may not transform a novice poet to a professional overnight, our objective is to create a computational tool which assists human creativity, blocking out any unnecessary concerns (such as rhyme scheme or meter in the case of poetry). In addition, we hope that the tool itself would evolve “creatively”, and be able to generate poetry on any given topic.

“Poetry is what we are going to create
And this is something to which we relate
Creativity just flows in our veins
It is the fuel that works our brains
And thus this project will be great!!!”

During our exploration of computer generated poetry, we found several existing systems which try to eliminate the need for a human poet! Such systems pose the question, “Why can’t a computer generate poetry all on its own?”

Among the many we encountered, there are some systems which generate poetry based on sentence patterns [2], some which produce a particular type of poetry (such as a simple haiku) [3] or on a specific topic from a finite set of topics [4]. There are also generators which use Tweets as a source of data to generate poetry (which are often jarring) [5]. Poetry from such systems is rarely comprehensible or meaningful, and does not compare favorably with something that an amateur poet might come up.

Our hope is to create an “integrated” poetry framework, which does all of the above and at the same time produce poetry which is qualitatively better than the ones from these poetry generators.

Possible Solutions

“Give me something I can read
And this to our system I will feed
And it will tell me if you can write
And if you have enough foresight
To create a poem that our system needs!!!”

There are many ways that our team could approach the domain of poetry, as well as each of our individual goals. With respect to poetry generation, we will be exploring several freely available online data sources, for instance, Wikipedia. We could feed the system a Wikipedia article, and then the system would write a poem about the topic of the article. The Wikipedia article could be a source of either just the topic of the poem, or the words to be used in the poem, or both (along with others such as news articles/ social media feeds)

Similarly we could also use a large collection of Wikipedia articles as the source from which our system finds “hidden” poetry. Just based on the laws of probability, strings of words in such a large piece of prose should match the structure of a poem. We could probably limit our scope to finding just a few particular types of poems in such prose (such as a Haiku); this would allow us to codify a set of rules and hopefully will make our system more efficient.

Our third goal is to be able to aid humans in writing poetry. Composing poetry requires the writer to follow well established lattices for poetry. If a composer wished to write a sonnet for example, he or she would likely follow one of a handful of established rhyme schemes and choose iambic pentameter as their poem’s metre. This latticework of rules may prove to be useful constraints for creating beautiful and moving poetry, but they can also be annoying to new poets and distract the writer from the content of the poem. At the same time, the rules of a type of poem can be just as powerful when broken than when followed. For instance, if a poem establishes and follows a certain rhyme scheme across multiple stanzas and then suddenly breaks the rhyme scheme it can evoke powerful emotions like confusion and disaccord in the reader.

To assist in this creative process, we plan to create templates for some of the categories of poems, which will encapsulate the unique aspects of that category (for instance a haiku has 3 lines each having 5, 7 and 5 syllables respectively). We will also try to provide words (from a lexical database such as WordNet [6]) which can fit into these templates, from which the poet can then pick and choose the ones he/she would like to be in the poem.

Preferred Solutions

“Is this a poem?’ you may ask
I will say that’s not a tough task
I have with me this project code
That’s on the development road
And once it’s done, in glory we will bask! ”

As we explored the various poetry project ideas we noticed that all of our ideas relied on a system that understands what it means to be a poem. Thus our primary objective is to create a core framework that is capable of *understanding and recognizing both the elements of poetry and poems*. For instance, the system will know what words rhyme with each other and use this information to understand the various rhyme schemes of poems. The system will also know the syllables of words and use this to understand what the metre of a poem is. With just the core concepts of metre, syllable count, and rhyme scheme most types of poetry can be recognized.

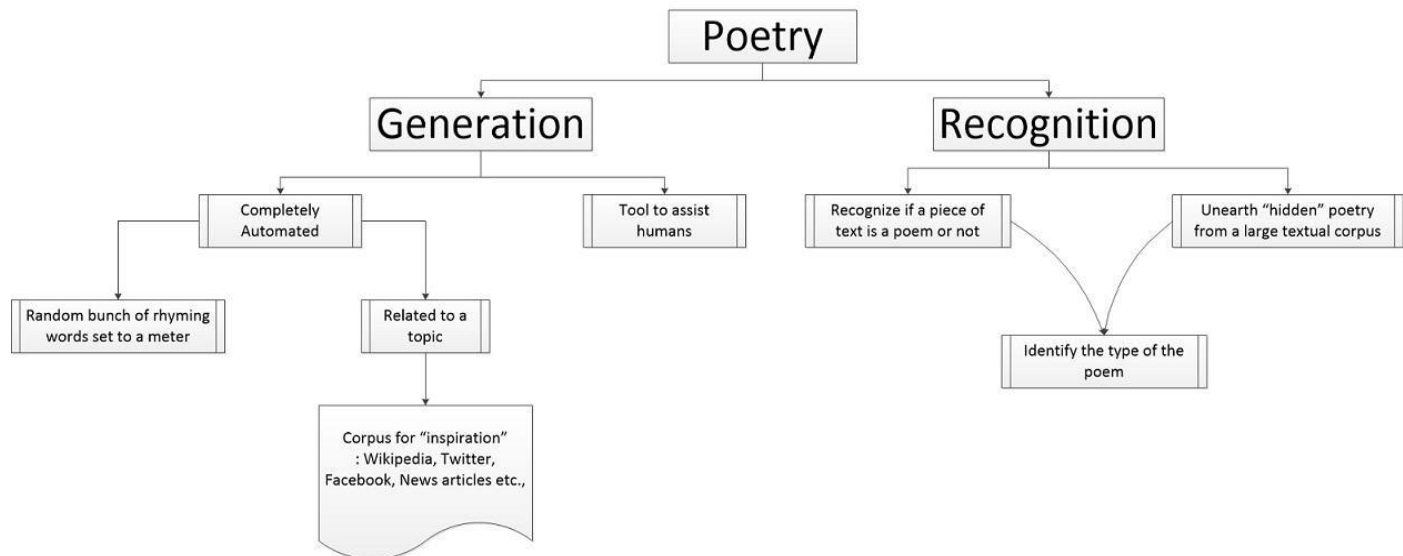
This poetry framework can then be used to run a series of experiments including but not limited to the ideas presented in the previous section. For example, if the system is able to recognize a sonnet by its metre and rhyme scheme, then it simply has to search through a body of prose for a block of text that satisfies the requirements to achieve our “hidden” poetry idea. The same core recognizer could also easily be used to recognize the characteristics of human generated poetry and machine generated poetry, thus proving to be a core component of our poetry writing tool and machine generated poetry project idea respectively.

The process of building and using the framework for our experiments will surely reveal more problems, questions, and experiments in the domain of poetry. When a computational model is not sufficient to explain a human’s thought process, we assume that it involves a creative “leap” [7]. Through our tool, we will attempt to quantify this leap and the revelations during the process of constructing such a system will help us understand poetry and the creative forces that drive its existence. It will help us understand what the core elements of a poem are and how humans craft and understand poetry. This is an opportunity for us to study meta-cognition - the intricacies of the processes involved in crafting a good poem.

One of our lofty goals is to be able to find out if a computer program can be deemed creative. We will try to explore how far we can model a machine to understand a concept as abstract as poetry and evaluate the poems this system generates by performing a simple Turing test - hand out poems written by our in-house poet [*] as well as the program to human evaluators and see if they are able to tell which poem was written by the human and which poem by the computer program. We also plan to conduct a mini user survey, in which we hope to get constructive criticism from an amateur poet who will be using our poetry assistant and using this feedback evaluate the effectiveness of our tool.

The Big Picture - An Abstract Overview of our proposed tool

“Cannot write a poem? Our tool will facilitate
In various ways our tool your brain will stimulate
To help you read poetry and recognize it too
You can be a poet, yes that’s true!
And these are the modules we hope to create!”



References

1. Csikszentmihalyi, Mihaly. "The Flow of Creativity." *Creativity: Flow and the Psychology of Discovery and Invention*. New York: Harper Collins Publishers, 1996. 110. Print.
2. Wonderful Poetry Machine. Retrieved from <http://wonderfulpoetrymachine.com/>
3. Haiku Generator. Retrieved from <http://www.generatorland.com/glgenerator.aspx?id=113>
4. Poem Generator. Retrieved from <http://thinkzone.wlonk.com/PoemGen/PoemGen.htm>
5. Eisenstein, Jacob. Pentametrone: auto-poetry from Twitter. Retrieved from <http://gtnlpclass.wordpress.com/2012/04/30/pentametrone-auto-poetry-from-twitter/>
6. Princeton University "About WordNet." WordNet. Princeton University. 2010. Retrieved from <http://wordnet.princeton.edu>
7. Sternberg, R. J. (2006). The nature of creativity. *Creativity Research Journal*, 18(1), 87-98.

[*] Our in-house poet is Sonal Danak; all verses appearing in this document are her original limericks.