Kelly Flood May 12, 2020
KAF2189
Literature in the Age of Artificial Intelligence with Prof. Dennis Yi Tenen

Final Project

Kuhlmann's "Love-Kiss XLI": A Legacy of Combinatorial Art

Quirinus Kuhlmann's "Love-Kiss XLI," and the combinatorial method behind it, are emblematic of the historical trend of combination and combinatorial art, a long legacy of other forms of permutation and combination. As Gerard Gillespie describes of Kuhlmann, "If his program, too, incorporates instruction in the principles of grammar and eloquence, this involves not sterile rote learning, but training in the actual generation of a surprising abundance of words and forms from the hidden potential of language, that is, the creative act of adding to language by deriving variety from the essentials."[1] Kuhlmann's method behind "Love-Kiss XLI" is based on his assertion that poetry is a way to unlock wisdom and discover meaning in the universe. His "Love-Kiss" sonnets are combinatorial by nature and experimental in that they explore different permutations to find truths which were previously only known to God. In a letter to Athanasius Kircher, Kuhlmann states, "No Science is closer to Wisdom than divine Poetry; none has been more unknown up until now, at the same time as none has been more widely known… as the maker of poems, which is the proper function of poetry, it surpasses all human knowledge."[2]

The method of creating these sonnets involves combining words and phrases according to a specific format which is derived from the proteus verse, which is the poetic equivalent of word-combinations; "by transposing the words the proteus verse gives rise to a great number of 'new' poems."[3] The sonnet contains "some 13! potential permutations, permutations that would

---

[1] Gerald Gillespie, "Primal Utterance: Observations on Kuhlmann's Correspondence with Kircher, in View of Leibniz's Theories" in *Garden and Labyrinth of Time: Studies in Renaissance and Baroque Literature* (New York: 1988): 94.
[2] *Ibid*.
[3] Gottfried Wilhelm Leibniz, *Dissertatio de Arte Combinatoria. In qua Ex Arithmeticae*… (Leipzig: 1666): 86 from Jan C. Westerhoff, "Poeta Calculans: Harsdörffer, Leibniz, and the 'Mathesis Universalis'" (1999): 456.

contain the possibility of saying and knowing everything there is to know in the sciences."[4] Inevitably, word-combinations will sometimes yield incomprehensible outcomes or outcomes which do not lend any new insights or unlock any new meanings for us. But if the method is repeated over and over again, creating a large number of these sonnets à la "Love-Kiss XLI," it will eventually give some outcome which does reveal something new to us.

Kuhlmann's methods are very much a legacy of previous combinatorial arts, beginning with Ramón Llull's *Ars Brevis* in 1308. Llull's Art describes a method which situations any subject or user of the method in relation to all other objects in the world through "systems systematizing systems."[5] The components of the Art - including the alphabet, figures, definitions, rules, table, etc. - which are combined to explain the intellect so that it may be applied to new ideas; the Art uses logic and pure combination to discover all possible syllogisms on any given subject.[6] It is a simplified combinatorial method which can be learned and practiced by any user, although Kuhlmann cautioned against replicating too simple a method, because it would enable 'parroting without understanding.'[7]

Gottfried Leibniz made the next great leap in combinatorial art with his "On the Universal Characteristic" in 1679. Similar to Llull's goal of creating a method which allows us to better understand the universe, Leibniz aimed to create a method which could find and demonstrate axioms, in essence general truths; however, he was specifically interested in benefiting the Church by demonstrating truths about God.[8] Leibniz's method on the universal characteristic rested on his position that numbers can explain the universe; ideas and words can be reduced to numbers and used to assess the quality of arguments and essentially demonstrate their

---

[4] Chirsopher D. Johnson, "N 2, or a Late Renaissance Poetics of Enmeration," (2012): 1131.
[5] Ramón Llull, "Ars Brevis" in *Selected Works of Ramón Llull (1232-1316)*, vol. 1 (Princeton: 1985).
[6] *Ibid*.
[7] Gillespie, "Primal Utterance": 93
[8] Gottfried Wilhelm Leibniz, "On the Universal Characteristic" in *Philosophical Papers and Letters* (Dordrecht, Boston: 1989): 221-229.

truthfulness. His method allows for a kind of synthesis based on combination, whereby his universal characteristic can lead to the discovery of new truths in the universe. Leibniz's contribution is also significant for "Love-Kiss XLI," because he formulated the proteus verse which Kuhlmann used to structure his sonnets; Leibniz defined a proteus verse as "a couplet or a longer poem which is constructed in such a way that the order of words can be changed without destroying its meter or rhyme."[9]

Kuhlmann shares similar goals to Kircher, Llull, and Leibniz in that he wants to understand the universe by discovering new truths, and he was interested in "an art of arts and a method of methods" to achieve this goal.[10] While he learns from Llull's and Leibniz's accomplishments in the domain, ultimately his method is an alternative rather than strict adaptation of their methods. As already described, Kuhlmann's method is based on poetry as a science of wisdom and particularly combinatorial poetry, in the context of "Love-Kiss XLI" and partially learned from Leibniz's proteus verse, as a means of uncovering any and every possible thing there is to be known.

Introducing the Poem Generator: Methods and Insights

To recreate Kuhlmann's combinatorial poetic method developed in "Love-Kiss XLI," I've written a Python code in Spyder which can generate poems in the style of Kuhlmann's sonnets. To start writing the code (see below), I first sought to understand the structure of the sonnet. Each sonnet consists of three verses, each comprising four lines. Each line consists of thirteen parts: a two-syllable prepositional phrase (such as "From Night"), eleven parts each comprising a one-syllable noun, and a three-syllable noun combination (such as "Fire and Plagues" or

[9] Leibniz, *Dissertatio de Arte Combinatoria*: 86 from Westerhoff, "Poeta Calculans": 456.
[10] Gillespie, "Primal Utterance": 90.

"Morning Rays").[11] There are also some stylistic features, primarily the capitalization of each word and the separation of each part by "/".[12]

I began by collating into pandas DataFrames all of the words from various text files - including lists of all one-syllable nouns[13], one-syllable prepositions, and three-syllable noun combinations[14] - that the code would need. I first used a list of *all* one-syllable nouns in the English language, but I found the use of obscure (and offensive) words limited the output sonnet's readability; instead, I made a new DataFrame merging words from the list of one-syllable nouns and from the list of the most common words in the English language (excluding swear words).[15]

Next, I defined functions - each building upon the last - which would be used to construct the complete sonnet. The first such function picked a random word from one of the aforementioned DataFrames, capitalized it, and returned it as an output. Next, I began functions to construct the parts of each line of a verse. One function makes the beginning, picking a random preposition and a random one-syllable noun and returning a prepositional phrase. The second makes a list of eleven random one-syllable nouns and joins them together, separated by "/", returning eleven parts each with one-syllable nouns. The third chooses or makes noun phrases to create the three-syllable ending of a line. The fourth makes lists of the beginning, middle, and end parts and joins them together, separated by "/", to form one complete line. The stanza and sonnet functions work similarly. The former makes four lines, puts them in lists, and joins them to create one verse, while the latter makes three verses, puts them in lists, and joins them to create an entire sonnet. Finally, the Kuhlmann sonnet function prints a readable sonnet

---

[11] Quirinus Kuhlmann and Richard Sieburth, "Love-Kiss XLI," *Poetry* (2009): 13.
[12] *Ibid*.
[13] Ashley Bovan, "Word Lists for Writers."
[14] This text file was partially derived from EUinterpreter, "Noun-Noun Collocations," *Vocabulary*.
[15] "Google-10000-English," *GitHub*, Github, Inc. (2019).

in the style of Kuhlmann's "Love-Kiss XLI." Examples of sonnets produced by this code may be found below.

Conclusion: Lessons to Uncover

This prototype is a primitive version of the method behind Kuhlmann's sonnet. Firstly, it is not as purely combinatorial as perhaps Kuhlmann and other combinatorial artists aspire to in their methods. I took some liberties to keep the code simple and to ensure the output poems were more accessible and readable to any audience, particularly by constraining the possible words that could be selected for insertion into each line of the sonnet. The output poems also contain some repetition between lines. If the poem generator were to be as aspirational as envisioned by Kuhlmann, Llull, and Leibniz, the code should be much more exhaustive and be allowed to generate the most completely combinatorial poems possible. The tradeoff would be a large number of nonsensical output poems with very obscure words, but it would also widen the potential for generating unique poems which do uncover new truths or ideas, as Kuhlmann aimed for. Whether the current output poems, such as those examples below, reveal revolutionary truths about the universe is debatable. Secondly, the output poem does not perfectly follow the structure and rules of the sonnets in Kuhlmann's "Love-Kiss XLI." The lines of each output poem from this code are meant to form proteus verses, however they are imperfect proteus verses, since - in most cases - the ordering of the words *cannot* be changed without destroying the meter or rhyme. Additionally, the poem generator does not generate rhyming poems, so almost all output poems will not rhyme, nor will they have perfect proteus verses.

The poem generator, ultimately, reveals that the aspirations of combinatorial artists like Kuhlmann - namely, discovering universal truths through combination and permutation - may be unreachable. All of their methods intend to be simple - able to be learned, practiced, and

replicated by others. While the simplicity of Kuhlmann's method made it feasible to write a poem generator code even with limited coding experience, the method ultimately cannot reveal to its users universal truths. Kuhlmann, criticizing a method suggested by Kircher, warns, "For without the box the boy can answer nothing, and with the box he understands nothing but words. He produces whatever he hears, without understanding, like a parrot." Similarly, this poem generator can produce poems which mimic Kuhlmann's "Love-Kiss XLI," but the output poems are largely banal and uninspiring. They are words on a screen, and they don't seem to uncover any new ideas. But while the ultimate goal for these combinatorial artists may as well be unattainable, these goals for uncovering truths have motivated a pursuit by many to build upon their efforts, consistently pushing the frontier of mathematics and philosophy as they intersect.

Kelly Flood

Appendix

GitHub Repository: https://github.com/floodkelly/Kuhlmann-Love-Kiss-XLI-poem-generator

Example Sonnet 1

Where Eggs / Kiss / Wells / Freeze / Deals / Mom / Said / Unit / Desk / Dish / Pee / Sims / Size and Cure
In Bolt / Few / Feed / Smile / Caps / Match / Bye / Bloom / Things / Dock / Trap / Elect / Rubber Boot
With Know / Til / Wealth / Verse / Bent / Twist / Refer / Hearts / Def / Blend / Sports / Raise / Shoot and Arts
Where Tools / Reach / Rail / Super / Verse / Pos / Ski / Aim / Plot / Sell / Rule / Cube / Building Block

To Deal / Suit / Flush / Tube / Taste / Tries / Here / Hub / Fails / Hints / Jazz / Knows / Pencil Lead
Near Rush / Chair / Yoga / Spell / Tales / Swing / Skirt / Stripes / Zen / Alias / Drove / Zen / Lily Pad
In Meet / Wear / Fault / Ace / Tags / Cards / Ties / Curve / Over / Fame / File / Blame / Wells and Bag
In Metal / Mind / Sells / Hole / Tar / Pie / Burn / Came / Print / Seat / Groove / Shine / Travel Plan

Out Noise / Tires / Fit / Buzz / Twin / Month / Mom / Tar / Dig / Bikes / Fit / Plan / Doc and Skills
By Boats / Role / Praise / Duck / Grab / Knit / Skill / Tips / Wives / Spots / Sixth / Drinks / Butter Knife
In Drag / Ridge / Jeep / Dash / Volt / Cart / Lung / Jeep / Feed / Spine / Shake / Mars / Men and Cove
The Days / Bold / Threads / Rules / Bras / Taste / Def / Yes / None / Thumb / Fat / Yea / Gypsy Moth

Kelly Flood

Example Sonnet 2

Past Switch / Bold / Tears / Then / Doubt / Lid / Flows / Clone / Dice / Volt / Fails / Bit / Picture Frame
Down Draw / Polls / Sauce / Ate / Boot / Scores / Said / Unit / Neck / Zip / Yacht / Self / Knows and Drove
Near Sleeps / Send / Seeks / Coach / James / Like / Reel / Fun / Mask / Spoke / Tech / Stores / Crown Jewel
On Wants / Course / Tags / Smell / Tears / Queens / Minds / Wheat / Sink / Bags / Gauge / Tub /
Post Office

Come Paint / You / Things / Yoga / Wow / Seats / Fool / Sum / Mouse / Texts / Shame / Skill / Cutting Board
Where Praise / Zen / Tears / Will / Fleece / Fell / Cakes / Bet / Shift / Blah / Feel / Pix / Chi and Tool
The Help / Knives / Tube / Pay / Routes / Trip / Deer / Height / Buzz / Tons / Hits / Men / Bound and Boats
Come Stem / Spray / Sixth / Zip / Belt / Mice / Flows / Tie / Tops / Dome / Path / Raise / Cliff Dwelling

By Clause / Mood / Realm / Biz / Trick / Gnu / Leaves / Task / Bikes / Way / Curves / Falls / Chance Event
With Ride / Logs / Lunch / Arms / Aid / Check / Strike / Chat / Crude / Want / Dig / Cakes / Log and Luck
Near Mile / Bold / Chairs / Deal / Jazz / Cheers / Tool / Dice / Sleeps / Fork / Spell / Farms / Crown Jewel
To Burns / Cake / Peers / Shine / Kill / Might / Tan / Cart / Cop / Schemes / Twins / Rec / Burns and Spies

Kelly Flood

Example Sonnet 3

By Crude / Cheats / Mind / Name / Fact / Aid / Metal / Spears / Sprint / Fits / Hats / Pad / Rail and Kills
Like Theme / Are / Odds / Bits / Eyes / Metals / Like / Tours / Sports / Spec / Heel / Suit / Salad Green
The Nuke / Lots / Paste / Funds / Lives / Cheers / Crack / Clips / Wait / Grave / Dose / Docs / Salad Green
Off Gain / Pays / Urge / Doors / Belts / Sap / Lots / Height / Tools / Disk / Worst / Wed / Dome and Gift

On Yeast / Harm / Harm / Fit / Know / Mouse / Foam / Smile / James / Foam / Squad / Tries / Ban and Quote
With Craps / Noise / Boot / Pas / Miles / Gnu / Drag / Hate / Hole / Bikes / Aid / Tubes / Bird Feeder
Lest Bound / Ranch / Tools / Worst / Heel / Aids / Shift / Wheels / Lens / Hole / Sap / Yale / Sound Effect
Out Boy / Quiz / Tours / Feat / Fraud / Bound / Weeks / Div / Gig / Growth / Score / Flu / Lily Pad

With Reach / Grab / Modes / Blues / Myth / Dream / Weeks / Way / Wheat / Myth / Spell / Wear / Boat and Sake
To Stress / Glance / Tags / Taste / Modes / Tries / Facts / Mice / Fresh / Heights / Week / Mouse
/ Chin Music
Past Boy / Craps / Men / Choir / Must / Fame / Drag / Thread / Units / Bit / Join / Crew / Bird Watcher
Off Kill / Rap / Deals / Sends / Docs / Poll / Doors / Mate / Wound / Course / Tags / Arts / Tea Towel

Kelly Flood

Bibliography

Bovan, Ashley. “Word Lists for Writers.” www.ashley-bovan.co.uk/words/partsofspeech.html.

EUinterpreter. “Noun-Noun Collocations.” *Vocabulary*, www.vocabulary.com/lists/201114.

Gillespie, Gerald. 1988. “Primal Utterance: Observations on Kuhlmann’s Correspondence with Kircher, in View of Leibniz’s Theories” in *Garden and Labyrinth of Time: Studies in Renaissance and Baroque Literature*. New York: P. Lang. 89-114.

“Google-10000-English.” *GitHub*, GitHub, Inc., 19 July 2019, https://github.com/first20hours/google-10000-english.

Johnson, Christopher D. 2012. “N 2, or a Late Renaissance Poetics of Enumeration.” *MLN* 127, no. 5: 1096-1143. www.jstor.org/stable/43611283.

Kuhlmann, Quirinus and Sieburth, Richard. 2009. “Love-Kiss XLI.” *Poetry*, vol. 194, no. 1: 13-16. www.jstor.org/stable/25706521.

Leibniz, Gottfried Wilhelm. 1666. *Dissertatio de Arte Combinatoria. In qua Ex Arithmeticae fundamentis Complicationum ac Transpositionum Doctrina novis praeceptis exstruitur*… Leipzig.

Leibniz, Gottfried Wilhelm. 1989. “On the Universal Characteristic” in *Philosophical Papers and Letters*. Dordrecht, Boston: D. Reidel Publishing Company. 221-229.

Llull, Ramón. 1985. “Ars Brevis” in *Selected Works of Ramón Llull (1232-1316*), vol. 1. Princeton: Princeton University Press. 569-646.

Westerhoff, Jan C. 1999. “Poeta Calculans: Harsdörffer, Leibniz, and the ‘Mathesis Universalis.’” *Journal of the History of Ideas* 60, no. 3: 449-467. www.jstor.org/stable/3654013.