Date of acceptance Grade

Instructor

Introduction to Computational Creativity: Course Project and Take-Home Exam

Janne Nevalainen
(011733227)

Helsinki October 15, 2015

UNIVERSITY OF HELSINKI
Department of Computer Science

1 Project Overview (max 1 page)

The system creates poems. That is, sequences of words that are in some way pleasing for humans. The system asks user for 2 nouns, which will be used as subjects in poems. System then by using 5-grams from Google, transforms the search space to contain adjectives that are related to those nouns. ('as X as .* '+word). e.g. "as red as an apple", would relate red to apple and add red to the search space for adjectives for apple. Weights are also counted by how often words appear.

When adjectives are chosen by weights, system selects new nouns relating to those adjectives. "as red as an Y"

These are then used for creating poem, that has insight into original words given by user. Structure for poems are few templates given by creator. CMU pronouncing dictionary is used for finding words that rhyme when pronounced.

The goal of the system, to create something that is pleasing for humans, can not be done in isolation, but is highly tied to and defined by mankind and its culture. The choice of using 5-grams to get info from, will give system information about this culture. Also the system likely has more information about culture, through relations between words, than a single human. Second lines of the poems use nouns that are similar to original nouns in some context, but perhaps not in others.

2 Project Collaboration (max 0.3 pages)

Components used in the system.

Google's 5-grams
`http://storage.googleapis.com/books/ngrams/books/datasetsv2.html`

The CMU Pronouncing Dictionary
Pronunciation dictionary for North American English. `http://www.speech.cs.cmu.edu/cgi-bin/cmudict`

Relating to previous, couple functions for checking rhyming.

3 Running Instructions (max 0.3 pages)

Download file, `http://vismantic.hiit.fi/files/as_x.txt`
(or run getfile.sh) 200 MB

Run with iPython notebook 2.7:
"ipython notebook poem.ipynb"

or with Python 2.7:
"python poem.py"
In iPython notebook version it is easier to repeat creating poems.

Environment used:
Python 2.7.10, Anaconda 2.3.0 (64-bit)
NLTK 3.0.3
Other packages: re
random
numpy

Should there be need to install parts of NLTK, run nltk.download() from python and choose from list provided by installer. (possibly the CMU dictionary corpus)

4 Selected Project Results (max 2 pages)

Examples of poems created by the system.

```
(input: hour, life)
Hour will be little and hour is recently. Life was big and life was truly.
So late and so much as reign will be. So large and so large as young will be.

(input: apple, building)
Apple is now round and apple is so red. Building will be important and building i
As red and as good as cherry was. So fresh and so round as morning is.


Tree is so much and tree was high. River is romantic and river was always ly.
So rapid and so quickly as flight is. As rigid and as stiff as law is now.


Woman is now tender and woman was always tall. Home was always thin and home will
As gentle and as soft as kitten is. So much and so early as nation is so.


Cat is so lightly and cat is tall. Dog will be cold and dog was always watchful.
As softly and as active as bird will be. As much and as faithful as atlantic is.


Love will be strong and love is important. War is so early and war will be perman
As strong and as great as horse was. As old and as big as creation was.
```

5 Creativity as Search

a. Description of the system as search (max 0.6 pages) U - Universe contains all possible concepts. That is all sequences of words, all poems. Also all ideas of what is poetic.

R- Rules. When given nouns, system will transform the search space in R, on allowed words, as set of those adjectives that relate to given nouns. Also it adds to R a set of nouns, relating to the set of adjectives.

The structure of the poems is predefined as few templates. As such the system heavily restricts the allowed structure of poems. This limits the set of possible produced poems to small subset of U. It also means the products, though from smaller subset, are from a subset that is more likely to be poem like.

Adjectives and noun that are chosen are evaluated according to if they relate to nouns given. But this is done mainly when transforming the R for current situation.
E - evaluation
Rhyming is rare piece in this system that is evaluated. An adjective from set in R is chosen and it is evaluated if it rhymes with another adjective. If not, it is discarded and tried again. Even this evaluation is for a piece to be used in poem, not for the ready product.

T - method T(R,E) for searching U w.r.t E,R This System could be considered to have predefined structure for the poem (few of them) given by creator. So mostly the words to be used are searched for.

b. Transformationality of the system (max 0.4 pages) [Is your system transformationally creative/creative in the metalevel? If yes, how? If not, describe one way of extending the system to be transformational.]

When given nouns for poem, system transforms the search space for adjectives to use in poem, to include only those, that are related to given nouns.

Same goes for second line of poem. The set of nouns to use, are those, which are related to adjectives in the previous set.

So parts of search space are transformed according to the social interaction of the system and world. Evaluation part does not change.

One way to extend transformationality more, would be to model structural rules of the poem (e.g. placing of noun, adjectives, adverbs etc within poem, number of words per line). Have different combinations of these. Then let users evaluate (rank) 2 or more of these at a time. Then system could transform towards those poem structures that got higher evaluation.

6 Creative Autonomy (max 0.3 pages)

[Does your system have creative autonomy? Why or why not?]
The system changes its search space for words, according to external input. It will though discard the new search space after current words to be used in poem are discarded. It gives weights for the words it includes in search space for given input. This valuation of importance of words, it does self. It could possibly be considered a change of standards in small scale. When it gets new nouns, then it changes to new standards, where different adjectives are now highly valued. The structures for the poems are not transformed at all, but are predefined by creator.

The system does not have neither internal nor external evaluation for its final products. As such, it has nothing to transform towards. It's products were evaluated and transformed to produce poems by creator when the system was created.

The system is not autonomous in relation to final products. It does not evaluate its final creations and it does not change its standards more than temporarily.

7 Evaluation

a. Inspiring set (max 0.3 pages) [What is the inspiring set of your system? What can you tell about your system's (creative) behavior with respect to the inspiring set?]

The System does not have any poems as an inspiring set. In that sense the creations are not similar to any inspiring set.

Sets for adjectives and nouns to be used in poems are indirectly defined by Google 5-grams. For most words there are lot enough alternatives, that the result is not very similar to source. For some words there might be few alternatives, but this can be seen as feature of such words.

b. FACE (max 0.3 pages) [Which of the FACE ground level generative acts does your system perform? Which of the acts did you perform?]

Fg - System aims to create poems. Performed by me.
Cg - Concept. For given word, relating words are chosen by using 5-grams. This act was performed by me.

Eg - Expression System self finds adjectives and nouns to be used in poem, according to given nouns. So system creates the expression of above concept. Poems as expressions for the concepts for choosing words and combining them with structure to create a poem, are produced by the system.

Cg - Parts of the system were done as human/computer interaction. With iterative process the system produced poems and I as creator evaluated them, then tuned the concept. Mostly structures for poems and filling words were done this way.

8 Markov chains and genetic algorithms (max 0.5 pages)

[Compare Markov chains and genetic algorithms as (potentially) creative mechanisms, from the viewpoint of Boden's definition of creativity.]

Boden's 3 types of creativity: Combinatorial, exploratory and transformational.

Combinatorial: Genetic Algorithms combine different parts of solutions with each other. As such it is possible to bring solution working in one area, that would also work in a some ways different area.

Markov Chains on lower level are mostly generators. They generate material from their inspiring set, but the produced material has high similarity to the inspiring set. Markov Chain is weakly combinatorial. Through middle parts 2 ideas can get close to each other, but in very limited ways.

Exploratory and transformational: GA can transform the space to allow new structures. In the area of poems, for example, GA could produce different sequences of parts of speech - POS, to be used as structures for poems. And through mutation these structures can transform to new ones.

Markov Chain does not explore. Its search spaces is quite limited and connected to inspiring set. Also MC does not transform self or change its standards.

A higher level Markov Chain to produce POS tags for example and lower level MC's to generate content to these, would be a bit closer Boden's definition of creativity.