Corpus-Based Generation of Content and Form in Poetry

Jukka M. Toivanen, Hannu Toivonen, Alessandro Valitutti and Oskar Gross
Department Of Computer Science and
Helsinki Institute for Information Technology, HIIT
University of Helsinki, Finland

Abstract

We employ a corpus-based approach to generate content and form in poetry. The main idea is to use two different corpora, on one hand, to provide semantic content for new poems, and on the other hand, to generate a specific grammatical and poetic structure. The approach uses text mining methods, morphological analysis, and morphological synthesis to produce poetry in Finnish. We present some promising results obtained via the combination of these methods and preliminary evaluation results of poetry generated by the system.

Introduction

Computational poetry is a challenging research area of computer science, at the cross section of computational linguistics and artificial intelligence. Since poetry is one of the most expressive ways to use verbal language, computational generation of texts recognizable as good poems is difficult. Unlike other types of texts, both content and form contribute to the expressivity and the aesthetical value of a poem. The extent to which the two aspects are interrelated in poetry is a matter of debate (Kell 1965).

In this paper we address the issues of generating content and form using corpus-based approaches. We present a poetry generator in which the processing of content and form is performed through access to two separate corpora, with minimal manual specification of linguistic or semantic knowledge.

In order to automatically obtain world knowledge necessary for building the content, we use text mining on a *background corpus*. We construct a word association network based on word co-occurrences in the corpus and then use this network to control the topic and semantic coherence of poetry when we generate it.

Many issues with the form, especially the grammar, we solve by using a *grammar corpus*. Instead of using an explicit, generative specification of the grammar, we take random instances of actual use of language from the grammar corpus and copy their grammatical structure to the generated poetry. We do this by substituting most words in the example text by ones that are related to the given topic in the word association network.

Our current focus is on testing these corpus-based principles and their capability to produce novel poetry of good quality on a given topic. At this stage of research, we have not yet considered rhyme, rhythm or other phonetic features of the form. These will be added in the future, as will more elaborate mechanisms of controlling the content.

As a result of the corpus-based design, the input to the current poetry generator consists of the background and the grammar corpora, and the topic of the poem. In the intended use case, the topic is directly controlled by the user, but we allow the grammar corpus to influence the content, too. Control over form is indirectly over the choice of the two corpora. The only directly language-dependent component in the system is an off-the-shelf module for morphological analysis and synthesis.

The current version of our poetry generation system works in Finnish. Its rich morphology adds another characteristic to the current implementation. However, we believe that the flexible corpora-based design will be useful in transferring the ideas to other languages, as well as in developing applications that can adapt to new styles and contents. A possible application could be a news service in the web, with a poem of the day automatically generated from recent news and possibly triggering, in the mind of the reader, new views to the events of the world.

After briefly reviewing related work in the next section, we will describe the corpus-based approach in more detail. Then, we will give some examples of generated poetry, with rough English translations. We have carried out an empirical evaluation of the generated poetry with twenty subjects, with encouraging results. We will describe this evaluation and its results, and will then conclude by discussing the proposed approach and the planned future work.

Related Work

The high complexity of creative language usage poses substantial challenges for poetry generation. Nevertheless, several interesting research systems have been developed for the task (Manurung, Ritchie, and Thompson 2000; Gervás 2001; Manurung 2003; Diaz-Agudo, Gervás, and González-Calero 2002; Wong and Chun 2008; Netzer et al. 2009). These systems vary a lot in their approaches, and many different computational and statistical methods are often combined in order to handle the linguistic complexity and creativity aspects. State of the art in lexical substitution but not in poetical context is presented, for instance, by Guerini et al. (2011). We next review some representative poetry generation systems.

ASPERA (Gervás 2001) employs a case-based reasoning approach. It generates poetry out of a given input text via a composition of poetic fragments that are retrieved from a case-base of existing poems. In the system case-base each poetry fragment is annotated with a prose string that expresses the meaning of the fragment in question. This prose string is then used as the retrieval key for each fragment. Finally, the system combines the fragments by using additional metrical rules. In contrast, our "case-base" is a plain text corpus without annotations. Additionally, our method can benefit from the interaction of two distinct corpora for content and form.

The work of Manurung et al. (2000) draws on rich linguistic knowledge (semantics, grammar) to generate a metrically constrained poetry out of a given topic via a grammar-driven formulation. This approach requires strong formalisms for syntax, semantics, and phonetics, and there is a strong unity between the content and form. Thus, this system is quite different from our approach. The GRIOT system on its part (Harrell 2005) is able to produce narrative poetry about a given theme. It models the theory of conceptual blending (Fauconnier and Turner 2002) from which an algorithm based on algebraic semantics was implemented. In particular, the approach employs "semantics based interaction". This system allows the user to affect the computational narrative and produce new meanings.

The above mentioned systems have rather complex structures involving many different interacting components. Simpler approaches have also been used to generate poetry. In particular, Markov chains (n-grams) have been widely used as the basis of poetry generation systems as they provide a clear and simple way to model some syntactic and semantic characteristics of language (Langkilde and Knight 1998). However, the characteristics are local in nature, and therefore standard use of Markov chains tends to result in poor sentence and poem structures. Furthemore, form and content are learned from a single corpus and cannot be easily separated.

Methods

We next present our approach to poetry generation. In the basic scenario, a topic is given by the user, and the proposed method then aims to give as output a novel and non-trivial poem in grammatically good form, and with coherent content related to the given topic.

A design principle of the method is that explicit specifications are kept at minimum, and existing corpora are used to reduce human effort in modeling the grammar and semantics. Further, we try to keep language-dependency of the methods small.

The poetry generator is based on the following principles.

1. *Content:* The topics and semantic coherence of generated poetry are controlled by using a simple word association network. The network is automatically constructed from a so-called background corpus, a large body of text used as a source of common-sense knowledge. More specifically, the semantic relatedness of word pairs is extracted from their co-occurrence frequency in the corpus. In the experiments of this paper, the background corpus is Finnish Wikipedia.

2. *Form (grammatical):* The grammar, including the syntax and morphology of the generated poetry, is obtained in an instance-based manner from a given grammar corpus. Instead of explicitly representing a generative grammar of the output language, we copy a concrete instance from an existing sentence or poem but replace the contents. In our experiments, the corpus consists mainly of old Finnish poetry.

3. *Form (phonetic):* Rhythm, rhyme, and other phonetic features can, in principle, be controlled when substituting words in the original text by new ones. This part has not been implemented yet but will be considered in future work.

The current poetry generation procedure can now be outlined as follows:

- A topic is given (or randomly chosen) for the new poem. The topic is specified by a single word.
- Other words associated with the topic are extracted from the background graph (see below).
- A piece of text of the desired length is selected randomly from the grammar corpus.
- Words in the text are analyzed morphologically for their part of speech, singular/plural, case, verb tense, clitics etc.
- Words in the text are substituted independently, one by one, by words associated with the topic. The substitutes are transformed to similar morphological forms with the original words. The original word is left intact, however, if there are no words associated with the topic that can be transformed to the correct morphological form.
- After all words have been considered, the novelty of the poem is measured by the percentage of replaced words. If the poem is sufficiently novel it is output. Otherwise the process can be re-tried with a different piece of text.

For the experiments of this paper, we require that at least one half of the words were replaced. This seems sufficient to make readers perceive the new topic as the semantic core of the poem.

We next describe in some more detail the background graph construction process as well as the morphological tools used.

Background Graph

A background graph is a network of common-sense associations between words. These associations are extracted from a corpus of documents, motivated by the observation that (frequent) co-occurrence of words tends to imply some semantic relatedness between them (Miller 1995).

The background graph is constructed from the given background corpus using the log-likelihood ratio test (LLR). The log-likelihood ratio, as applied here for measuring associations between words, is based on a multinomial model of word co-occurrences (see, e.g., Dunning (1993) for more information).

The multinomial model for a given pair $\{x, y\}$ of words has four parameters $p_{11}, p_{12}, p_{21}, p_{22}$ corresponding to the probability of their co-occurrence as in the contingency table below.

	x	$\neg x$	Σ
y	p_{11}	p_{12}	$p(y;C)$
$\neg y$	p_{21}	p_{22}	$1-p(y;C)$
Σ	$p(x;C)$	$1-p(x;C)$	1

Here, $p(x;C)$ and $p(y;C)$ are the marginal probabilities of word x or word y occurring in a sentence in corpus C, respectively.

The test is based on the likelihoods of two such multinomial models, a null model and an alternative model. For both models, the parameters are obtained from relative frequencies in corpus C. The difference is that the null model assumes independence of words x and y (i.e., by assigning $p_{11} = p(x;C)p(y;C)$ etc.), whereas the alternative model is the maximum likelihood model which assigns all four parameters from their observed frequencies (i.e., in general $p_{11} \neq p(x;C)p(y;C)$).

The log-likelihood ratio test is then defined as

$$LLR(x,y) = -2\sum_{i=1}^{2}\sum_{j=1}^{2} k_{ij}\log(p_{ij}^{null}/p_{ij}), \tag{1}$$

where k_{ij} is the respective number of occurrences. It can be seen as a measure of how much the observed joint distribution of words x and y differs from their distribution under the null hypothesis of independence, i.e., how strong the association between them is. More complex models, such as LSA, pLSA or LDA could be used just as well.

Finally, edges in the background graph are constructed to connect any two words x, y that are associated with $LLR(x,y)$ greater than an empirically chosen threshold. To find words that are likely semantically related to the given topic, first-level neighbours (i.e., words association with the topic word) are extracted from the background graph. If this set is not large enough (ten words or more in the experiments of this paper), we add randomly selected second-level neighrbours (i.e., words associated to any of the first-level neighbors).

In the future, we plan to use edge weights to control the selection of substitutes, and possibly to perform more complex graph algorithms on the background graph to identify and choose content words.

Morphological Analysis and Synthesis

Morphological analysis is essential and non-trivial for morphologically rich languages such as Finnish or Hungarian. In these languages, much of the language's syntactic and semantic information is carried by morphemes joined to the root words. For instance, the Finnish word "juoksentelisinkohan" (I wonder if I would run around) is formed out of the root word "juosta" (run). Hence, morphological analysis provides valuable information of the syntax and to some degree of the semantics. In our current system, morphological analysis and synthesis are carried out using Omorfi[1], a morphological analyzer and generator of Finnish language based on finite state automata methodology (Lindén, Silfverberg, and Pirinen 2009).

With the help of Omorfi we can thus generate substitutes that have similar morphological forms with the original words. For instance, assume that the topic of the poetry is "ageing" and we want so substitute "juoksentelisinkohan" by a word based on "muistaa" (remember). Omorfi can now generate "muistelisinkohan" (I wonder if I would think back) as a morphologically matching word.

Examples

We next give some example poems generated by the current system with the original example texts used to provide structure for these poems. We also give their rough English translations, even though we suspect that poetical aesthetics somewhat change in translation. The substituted words are indicated by *italics*.

The first example poem is generated around the topic "(children's) play". We first give the Finnish poem with the template used to construct it (on the right) and then the English translation of both the generated and original poems.

Kuinka hän leikki *silloin*	kuinka hän leikki kerran
uskaliaassa, uskaliaassa //	suuressa vihreässä //
kuiskeessa	puistossa
vaaleiden puiden alla.	ihanien puiden alla.
Hän oli *kuullut* huvikseen,	Hän oli katsellut huvikseen,
kuinka hänen *kuiskeensa*	kuinka hänen hymynsä
kanteli helkkeinä tuuloseen.	putosi kukkina maahan,

Original by Uuno Kailas: Satu meistä kaikista, 1925

How she played *then*	how she played once
in a *daring, daring whispering*	in a big green park
under the *pale* trees.	under the lovely trees.
She had *heard* for fun	She had watched for fun
how her *whispering*	how her smile
drifted as jingle to the wind.	falled down as flowers,

The next poem is generated with "hand" as the topic. The template used is shown below the generated poem and thereafter the translations, respectively.

Vaaleassa kourassa
sopusuhtaisessa kourassa ovat *nuput* niin kalpeita
kuvassasi lepää *lapsikulta* jumala.

Alakuloisessa metsässä
Hämärässä metsässä ovat kukat niin kalpeita
Varjossa lepää sairas jumala

Original by Edith Södergran: Metsän hämärä, 1929

In a *pale fist*
in a *well-balanced fist*, the *buds* are so pale
in *your image* lies a *dear child* god.

In a gloomy forest
In a dim forest flowers are so pale
In the shadow lies a sick god

The final example poem has "snow" as its topic.

Elot sai karkelojen teitä,	Aallot kulki tuulten teitä,
lumi ajan *kotia*,	aurinko ajan latua,
hiljaa *soi kodit autiot*,	hiljaa hiihti päivät pitkät,
hiljaa *sai armaat karkelot* -	hiljaa hiipi pitkät yöt -
laiho sai lumien riemut.	päivä kutoi kuiden työt,

Original by Eino Leino: Alkusointu, 1896

[1] URL: http://gna.org/projects/omorfi

Lives got the frolic ways,
snow the home of time,
softly *chimed abandoned homes*,
softly *got frolics beloved -*
ripening crop got the snows' joys.

Waves fared the wind's ways,
sun the track of time,
slowly skied for long days,
slowly crept for long nights -
day wove the deeds of moons

Subjectively judging, the generated poems show quite a wide range of grammatical structures, and they are grammatically well formed. The cohesion of the contents can also be regarded as fairly high. However, the quality of generated poetry varies a lot. Results from an objective evaluation are presented in the next section.

Evaluation

Evaluation of creative language use is difficult. Previous suggestions for judging the quality of automatically generated poetry include passing the Turing test or acceptance for publishing in some established venue. Because the intended audience of poetry consists of people, the most pragmatic way of evaluating computer poetry is by an empirical validation by human subjects. In many computer poetry studies both human written and computationally produced poetry have been evaluated for qualities like aesthetic appreciation and grammaticality.

In this study we evaluated poetry using a panel of twenty randomly selected subjects (typically university students). Each subject independently evaluated a set of 22 poems, of which one half were human-written poems from the grammar corpus and the other half computer-generated ones with at least half of the words replaced. The poems were presented in a random order, and the subjects were not explicitly informed that some of the poems were computer-generated.

Each subject evaluated each text (poem) separately. The first question to answer was if the subject considered the piece of text to be a poem or not, with a binary yes/no answer. Then each text was evaluated qualitatively along six dimensions: (1) How typical is the text as a poem? (2) How understandable is it? (3) How good is the language? (4) Does the text evoke mental images? (5) Does the text evoke emotions? (6) How much does the subject like the text? These dimensions were evaluated on the scale from one (very poor) to five (very good). (The interesting question of how the amount of substituted words affects the subjective experience of topic, novelty and quality is left for future research.)

Evaluation results averaged over the subjects and poems are shown in Figures 1 and 2. Human-written poems were considered to be poems in 90.4% of the time and computer-generated poems 81.5% of the time (Figure 1). Intervals containing 66.7% of the poems show that there was more variation in the human-written poetry than in the computer generated poetry. Overall, these are promising results, even though statistically the difference between human-written and computer generated poetry is significant (p-value with Wilcoxon rank-sum test is 0.02).

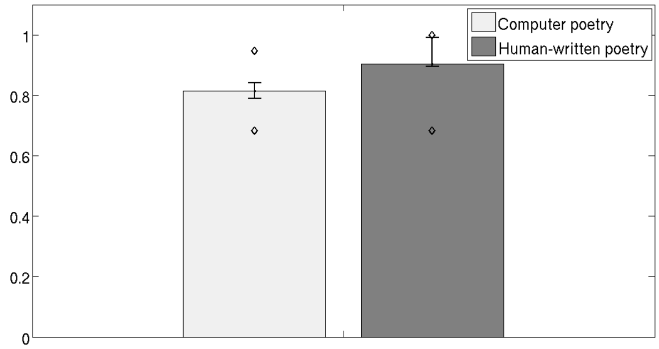

Figure 1: Relative amounts of texts (computer-generated and human-written poetry) subjectively considered to be poems, averaged over all subjects. The whiskers indicate an interval of 66.7% of poems around the median. Points indicate the best and worst poems in the both groups.

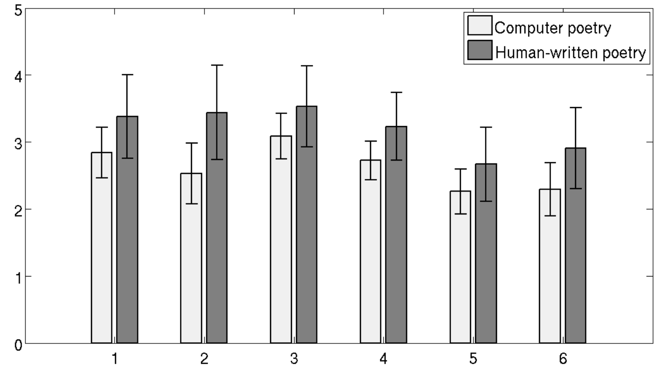

Figure 2: Subjective evaluation of computer-generated and human-written poetry along six dimensions: (1) typicality as a poem, (2) understandability, (3) quality of language, (4) mental images, (5) emotions, and (6) liking (see text for details). Results are averaged over all subjects and poems. The whiskers indicate one standard deviation above and below the mean.

The evaluated qualities have a similar pattern (Figure 2): The average difference between human-written and computer-generated poetry is not large, and in many cases there is a lot of overlap in the ranges of scores, indicating that a good amount of (best) computer-generated poems were as good as (worst) human-written ones. Statistically, however, the differences are highly significant (all p-values below 0.001). The biggest drop in quality was in understandability (dimension 2). However, somewhat controversially, the language remained relatively good (dimension 3). An interesting observation is that some of the generated poems were rated to be quite untypical but their language quality and pleasantness were judged to be relatively high.

Discussion

We have proposed a flexible poetry generation system which is potentially able to produce poetry out of a wide variety of different topics and in different styles. The flexibility is achieved by automating the processes of acquiring and applying world knowledge and grammatical knowledge. We use two separate corpora: background corpus for mining lexical associations, and grammar corpus for providing grammatical and structural patterns for the basis of new poetry. We have implemented the system for Finnish, a morphologically rich language. We carried out a preliminary evaluation on the produced poetry, with promising results.

It may be questioned whether the current approach exhibits creative behaviour, and whether the system is able to produce poetry that is interesting and novel with respect to the text that is used as the basis of new poetry. First, the generated poems are usually very different from the original texts (our subjective view, to be evaluated objectively in the future). Second, some of the generated texts were rated to be quite untypical, even though recognized as poems. The pleasantness and language quality of these poems were still judged to be relatively high. According to these observations we think that at least some of the system's output can be considered to be creative. Thus, the system could be argued to automatically piggyback on linguistic conventions and previously written poetry to produce novel and reasonably high quality poems.

Our aim is to develop methods that can be applied to other languages with minimal effort. In our current system, morphological analysis and synthesis are clearly the most strongly language-specific components. They are fairly well isolated and could, in principle, be replaced by similar components for some other language. However, it may prove to be problematic to apply the presented approach to more isolating languages (i.e., with a low morpheme-per-word ratio), such as English. In agglutinative languages (with higher morpheme-per-word ratio), such as Finnish, a wide variety of grammatical relations are realized by the use of affixation and the word order is usually quite free. We currently consider implementing the system for other languages, in order to identify and test principles that could carry over to some other languages.

So far, we have not considered controlling rhythm, rhyme, alliteration or other phonetic aspects. We plan to use constraint programming methods in the lexical substitution step for this purpose. At the same time, we doubt this will be always sufficient in practice since the space of suitable substitutes can be severely constrained by grammar and semantics. Another interesting technical idea is to use n-gram language models for computational assessment of the coherence of produced poetry.

We consider the approach described in this paper to be a plausible building block of more skillful poetry generation systems. The next steps we plan to take, in addition to considering phonetic aspects, includes trying to control the emotions that the poetry exhibits or evokes. We are also interested in producing computer applications of adaptive or instant poetry.

Acknowledgements: This work has been supported by the Algorithmic Data Analysis (Algodan) Centre of Excellence of the Academy of Finland.

References

Diaz-Agudo, B.; Gervás, P.; and González-Calero, P. A. 2002. Poetry generation in COLIBRI. In *ECCBR 2002, Advances in Case Based Reasoning*, 73–102.

Dunning, T. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational linguistics* 19(1):61–74.

Fauconnier, G., and Turner, M. 2002. *The Way We Think: Conceptual Blending and the Mind's Hidden Complexities*. New York: Basic Books.

Gervás, P. 2001. An expert system for the composition of formal spanish poetry. *Journal of Knowledge-Based Systems* 14(3–4):181–188.

Guerini, M.; Strapparava, C.; and Stock, O. 2011. Slanting existing text with valentino. In Pu, P.; Pazzani, M. J.; André, E.; and Riecken, D., eds., *Proceedings of the 2011 International Conference on Intelligent User Interfaces*, 439–440. ACM.

Harrell, D. F. 2005. Shades of computational evocation and meaning: The GRIOT system and improvisational poetry generation. In *In Proceedings, Sixth Digital Arts and Culture Conference*, 133–143.

Kell, R. 1965. Content and form in poetry. *British Journal of Aesthetics* 5(4):382–385.

Langkilde, I., and Knight, K. 1998. The practical value of n-grams in generation. In *Proceedings of the International Natural Language Generation Workshop*, 248–255.

Lindén, K.; Silfverberg, M.; and Pirinen, T. 2009. HFST tools for morphology - an efficient open-source package for construction of morphological analyzers. In *Proceedings of the Workshop on Systems and Frameworks for Computational Morphology*, 28–47.

Manurung, H. M.; Ritchie, G.; and Thompson, H. 2000. Towards a computational model of poetry generation. In *Proceedings of AISB Symposium on Creative and Cultural Aspects and Applications of AI and Cognitive Science*, 79–86.

Manurung, H. 2003. *An evolutionary algorithm approach to poetry generation*. Ph.D. Dissertation, University of Edinburgh, Edinburgh, United Kingdom.

Miller, G. A. 1995. Wordnet: a lexical database for english. *Communications of the ACM* 38(11):39–41.

Netzer, Y.; Gabay, D.; Goldberg, Y.; and Elhadad, M. 2009. Gaiku : Generating haiku with word associations norms. In *Proceedings of NAACL Workshop on Computational Approaches to Linguistic Creativity*, 32–39.

Wong, M. T., and Chun, A. H. W. 2008. Automatic haiku generation using VSM. In *Proceedings of ACACOS*, 318–323.