Joanna Misztal-Radecka
Bipin Indurkhya

A BLACKBOARD SYSTEM FOR GENERATING POETRY

Abstract

We present a system to generate poems based on the information extracted from input text such as blog posts. Our design uses the blackboard architecture, in which independent specialized modules cooperate during the generation process by sharing a common workspace known as the blackboard. Each module is responsible for a particular task while generating poetry. Our implementation incorporates modules that retrieve information from the input text, generate new ideas, or select the best partial solutions. These distinct modules (experts) are implemented as diverse computational units that make use of lexical resources, grammar models, sentiment-analyzing tools, and language-processing algorithms. A control module is responsible for scheduling actions on the blackboard. We argue that the blackboard architecture is a promising way of simulating creative processes because of its flexibility and compliance with the Global Workspace Theory of mind.

The main contribution of this work is the design and prototype implementation of an extensible platform for a poetry-generating system that may be further extended by incorporating new experts as well as some existing poetry-generating systems as parts of the blackboard architecture. We claim that this design provides a powerful tool for combining many of the existing efforts in the domain of automatic poetry generation.

1. Introduction

Some researchers consider creativity to be an exclusively human domain that cannot be manifested in a computational system [26, 36]. Others (Alan Turing, for instance) acknowledge that creativity is a key feature that needs to be incorporated in a machine before we can consider it to be truly intelligent, but do not rule this out as a possibility in the future [46]. This position is also echoed by Geoffrey Jefferson in 1949 [24]:

> *Not until a machine can write a sonnet or compose a concerto because of thoughts and emotions felt, and not by the chance fall of symbols, could we agree that machine equals brain* (...).

Consequently, in recent years, many scientists have taken up this challenge, and a new research area of computational creativity has emerged [4]. There are essentially two approaches to incorporating creativity in computational systems. One approach is based on the argument that, because computer architecture is different from brain architecture, the processes by which humans generate creative artifacts may not have much relevance for designing computational creativity systems [19]. In this approach, one focuses on the nature of the created artifact: two key characteristics of these artifacts are deemed to be originality and intelligibility [23], and the goal for a computational creativity system is to generate such artifacts.

However, in the other approach, it is argued that the process of creation should also be considered in conjunction with the characteristics of the outputs [9]. Consequently, these approaches to computational creativity focus on modeling the processes by which creative artifacts are produced; for example, by incorporating the intentions of the artist [10]. Some frameworks for evaluating the creativity of computational systems have also been proposed [9, 25, 39].

We follow this latter approach in the research described here. To incorporate cognitive processes underlying creativity, we rely on the Global Workspace Theory of mind [1, 2], according to which mental processes can be modeled as a large number of autonomous specialized modules, some conscious and some subconscious, interacting together, sharing a common workspace. The blackboard architecture [11] provides a natural platform to implement this model. In this architecture, a number of independent agents interact together through a shared memory space called the blackboard to arrive at a solution to the problem. It was originally proposed for speech understanding [15] but now has been successfully applied to solve many problems in different domains [7, 8, 20].

The blackboard architecture also allows us to incorporate the assumptions of Baar's theory [1, 2] for simulating creativity, according to which the poetry-generation process is performed by a group of experts – each of which has some specific knowledge about some aspect of writing poetry, and all of them share a global workspace (the blackboard).

Here we present our ongoing research on a poetry-generating system based on the blackboard multi-agent architecture. Some parts of this work were presented earlier in [31–33].

The rest of this paper is organized as follows: In Section 2, we present the background of this work, including the existing approaches to poetry generation and emotional modeling. In the same section, we also introduce the blackboard architecture and other external resources that are incorporated in our system. Lexical resources used in our system are described in Section 3. In Section 4, we present the implemented model of our system. Different components of the system, such as the blackboard objects, poetry experts, and control component, are described in various subsections. For example, the details of the poetry-generation algorithm are presented in Section 4.4, and an example is provided to illustrate the generation process. In Section 5, we present a sample of the outputs generated by our system, some remarks on interpreting these outputs, and a summary of our evaluation results according to two different criteria. Finally, Section 6 summarizes the contribution of this work and contains some remarks about the performance of our system, suggestions for future improvements, and a possible application of the system as an interactive platform.

2. Background and motivation

2.1. Poetry-generating systems

Some approaches to generating aesthetically pleasing poems are based on a set of predefined templates. For instance, Ray Kurzweil's *Cybernetic Poet* [28] generates a language model from a set of poems input by the user, and composes new ones in the same style.

However, a more challenging task is to design a system that generates poems in an intentional way. There are some systems that take up this challenge. For example, ASPERA [17] generates poems with a forward-reasoning knowledge-based system. It uses selected fragments from a corpus to transform an input prose message into a poem. A rule-based approach is used to select the most suitable metrics according to the predefined user preferences.

Jukka Toivanen et al. [45] present a system that creates novelty by replacing existing words with new ones in existing Finnish poems. The system incorporates two separate corpora for defining the grammatical structure and semantic content of poems. In a subsequent work, Jukka Toivanen et al. [44] introduce a constraint programming technique for poetry generation. Their system consists of two modules; the first articulates aesthetic constraints and search space based on the inspiration from the user or the environment, and the other generates poems by applying these constraints in the articulated search space.

As poetry has a strong emotional component, several poetry-generation systems aim to incorporate emotional affects into the creation process. Simon Colton et al. [10] present a corpus-based poetry generator that creates poems according to *the mood of the day*, as estimated from the news of that day. However, the mood is defined so that it can only be *good* or *bad*, with no further refinement of the emotional state.

The *Stereotrope* system [47] generates emotional and witty metaphors for a given topic based on the analysis of a corpus and the insights from commonsense knowledge.

Several poetry-generation systems make use of a multi-agent architecture. For example, in MASTER [27], a *society* of agents in various emotional states influence the moods of each other with their pieces of poetry. The final poem embodies *social learning*, as the agents interact by reciting their own pieces of poetry to each other, thereby influencing others (while, at the same time, being influenced by them). The poems produced by this system are not meaningful in the usual sense, but they consist of repeated words and sounds that create the poetical form. In contrast to the blackboard approach, the agents in this system interact directly with one another, and there is no global knowledge. Moreover, all of the agents in MASTER are implemented as the same type of module whereas, in a blackboard architecture, agents can be diverse computational modules with distinct knowledge sources and roles.

The McGONAGALL system [29] provides another approach in which multiple modules cooperate together, and the final solution evolves through this interaction. In this system, evolutionary algorithms are used to generate poems that satisfy the constraints on *grammar, meaning, and poeticity*. This approach allows for the pursuit of multiple alternative solution paths in parallel, focusing on more-promising results, and coming back to former ideas (if needed).

The WASP system [17, 19] incorporates a set of specialized families of experts, considered as a *cooperative society of readers/critics/editors/writers* that cooperate during the poetry-generation process. This approach is considered more powerful than the former rule-based approach in WASP and is reminiscent of the blackboard architecture, as the agents work together on a common population of poem drafts. However, WASP does not incorporate the blackboard model directly.

Among the above-mentioned systems, we can distinguish two different approaches towards modeling each system's *personality*. In the first approach, the system's behavior is determined by some predefined parameters (e.g., in MASTER – each agent is initialized with a mood represented by the coordinates in the emotional space). Another alternative is to adapt the emotional state to some environmental factors. This approach is taken by Colton et al. [10], where the mood of the day is calculated from the sentiment value in daily news. The *Cybernetic Poet* also builds a data-driven model, but it exhibits neither creative nor emotional behaviors – the system can only replicate the style of the existing poetry.

In our system, we combine both approaches – the emotional state is acquired based on the affective information extracted from the blog text, but it also depends on the individual features of the system. The model of emotions and optimism rates that represent the bias for the affective interpretation of input (how *optimistic* the system is) gives the system an individual personality. Hence, the external input (the text of the blog) is used only as an inspiration for the theme and a stimulus for the affective state.

Another important issue in the poetry-generation system design is the way of defining the *inspiration* or the main theme for the poem. The *Cybernetic Poet* [28] takes inspiration from the existing poetry and selects the most typical words for the new poems (while avoiding plagiarism). In MASTER, the agents select initial words according to the predefined sentiment values. In a system designed by Tony Veale [47], poems are generated from user-supplied phrases. In some systems, the inspiration is derived from the input text: as in Colton et al. [10], where the mood and theme for the poem are defined based on some news texts; or in Toivanen et al. [43], where the document- specific word associations are extracted from the input text and used for the generation of the poem.

In our approach, the generated poems are also inspired by the input text (such as Internet blog posts) that carries some emotional content. First, our system analyzes the text to find the most inspiring phrase as follows: Each phrase occurring in the text is rated according to the number of diverse associations derivable from it, and the phrase with the most diverse associations is selected. Then, we compute an emotional state corresponding to the input text using an internal model of emotions with an individual optimism rate (depending on whether the emotional interpretation has some pessimistic or optimistic bias). Finally, poems are generated based on the inspirational phrase that contains words that express the emotions extracted from the input text, as well as other typicaly poetic expressions.

As the generation phase typically produces a number of outputs (many of which may not be interesting), one of the most difficult aspects of a computational creativity system is the process of selecting the best candidates. In Cybernetic Poet, a set of constraints is defined for each word, and the most suitable candidates are selected (relaxing the restrictions as necessary). In the genetic-algorithms approach [29], the next population of poems is evaluated by the fitness functions. In our system, we incorporate experts that apply specific evaluation functions to individual lines in the generated poems. As one may expect, it is difficult (and sometimes impossible) to satisfy all of the constraints simultaneously. Therefore, the experts apply the evaluation criteria sequentially, creating a hierarchy of constraints. The following evaluation experts work on the set of expressions pruned by their predecessors.

2.2. Blackboard architecture

The blackboard model [11] is often visualized by the metaphor of a group of independent human experts with diverse knowledge sitting in a room with one blackboard. When the initial problem is written on the blackboard, the experts start to add their contributions incrementally while monitoring the blackboard until they find some information that they can use for their contributions. The experts continue to add their contributions to the blackboard until the problem is solved.

The blackboard model allows for a diversity of modules to be implemented. Each expert is viewed as a black box, with the internal implementation details hidden: the only aspect that needs to be standardized is the interface between the module and the blackboard.

The information on the blackboard may be organized at different levels of abstraction and can be in different formats (as long as the appropriate modules can interface with it). At any point in time, several modules may be active, so there is a control component implemented as an independent module that manages the scheduling of modules. Based on heuristics, this component is responsible for selecting the most beneficial module to run at each step of the problem solving.

As the solution in a blackboard system is built incrementally, this architecture is effective for problems that require numerous steps or the use of many diverse sources of knowledge. The blackboard architecture also facilitates searching in a large space of possible solutions, including many paths of reasoning with incomplete or imprecise knowledge. The search strategy may be flexibly adapted to different stages of problem solving, allowing it to work on several levels of abstraction and explore multiple paths of reasoning.

One of the most important problems in blackboard systems is the selection of a solution path that should be further pursued. Hence, many approaches have been developed to choose the focus of attention. As in a standard problem-solving task, several state-space search strategies may be used when the focus is being selected [7], such as depth-first search, breadth-first search, or an incremental search given some *a priori* knowledge.

In our system, we use a combination of the above-mentioned problem-solving approaches. We use the depth-first search to determine the main theme and topic for the poem by selecting the most inspiring key phrase. As a result of this choice, our system gets *inspired* by a particular theme instead of generating a large set of poems on different topics.

For the next step of the poetry-generation process, we use the best-first search strategy: different experts produce partial solutions at various levels of abstraction, thereby generating a large set of diverse ideas that, in turn, inspires other experts for their creative outputs.

In the last step of the generation process, an incremental search is used. A set of constraints is gradually applied to the generated phrases to select the poetry lines that best satisfy the predefined evaluation criteria, and some experts ensure that certain user-customized constraints (such as rhyme and rhythm patterns) are satisfied in the subsequent lines of the poem.

Different blackboard implementations also vary in their approaches to the control mechanism [7]. Some basic techniques include variations of the agenda-based control where all of the competing actions are placed on an agenda. At each step, the activated actions are ranked, and the action with the highest score is executed. Some other approaches incorporate subgoaling techniques, a hierarchical control, or involve a separate control planning system [12].

In our system, we have adapted the original agenda-based control with some modifications proposed by Barbara Hayes-Roth [20]. The partial solutions produced by the experts are placed on the agenda and rated. We set aside a part of the

blackboard for the control data. We also incorporate a distinct set of control experts to carry out the rating tasks.

The blackboard architecture is naturally suitable for simulating cognitive processes such as creativity because of its flexibility and relation to the Global Workspace Theory [1, 2], and it has already been incorporated into some computational creativity systems. For example, Geraint Wiggins [49] proposed the Global Workspace Theory as a model for simulating creative music composition. A second example is the Slant system for story generation [34], in which the blackboard model combines separate subsystems dealing with plot, figuration, and the narrative discourse tasks during the generation process. Yet another application of the blackboard system was described by John Charnley and Simon Colton [8] for combining reasoning systems used in discovering mathematical theorems and generating conjectures.

2.3. Emotional modeling

The research on human emotions dates back to ancient times [6]. One of the first categorizations of emotional states was made by Cicero, who separated them into four categories: fear, pain, lust, and pleasure. Later studies elaborated on this; for example, by Charles Darwin [13], Paul Ekman [14] (who defined six basic emotions of happiness, sadness, fear, anger, disgust and surprise), and many others.

In recent years, a number of dimensional models have been proposed to represent emotions. For example, in the two-dimensional circumplex model (Fig. 1) proposed by James Russel [40], the horizontal dimension represents pleasure-displeasure, and the vertical dimension represents arousal-sleep. In another two-dimensional model proposed by Cynthia Whissel's [48], one set of coordinates represents evaluation (*positive-negative*) and the other activation (*passive-active*). Various words from the Dictionary of Affect in Language are placed in this space. Yet another model is Robert Plutchik's wheel of emotions [38], consisting of eight basic and eight composite emotions placed in a circle, where the similarity of emotions is represented by the radial dimension.

The dimensional models are promising tools for the computational modeling of emotions, as they provide a simple way to measure, define, and compare different affective states. They are used in AI systems to simulate emotional personalities (as proposed by Albert Heide and Gracian Trivino [21] or by Alexis Kirke and Eduardo Miranda [27]). However, they have some significant limitations, as they are largely based on the verbal representation of affects. In particular, they do not allow the definition of compound emotions, and they do not consider the situation where several emotions are being experienced at the same time [6].

In our system, we adapted the two-dimensional model of emotions proposed by James Russel, supplementing it with the emotional categories corresponding to the WordNet-Affect lexicon categories (such as anger, fear, happiness, etc.) as well as coordinates derived from the ANEW database (containing valence and arousal values of words) [5].
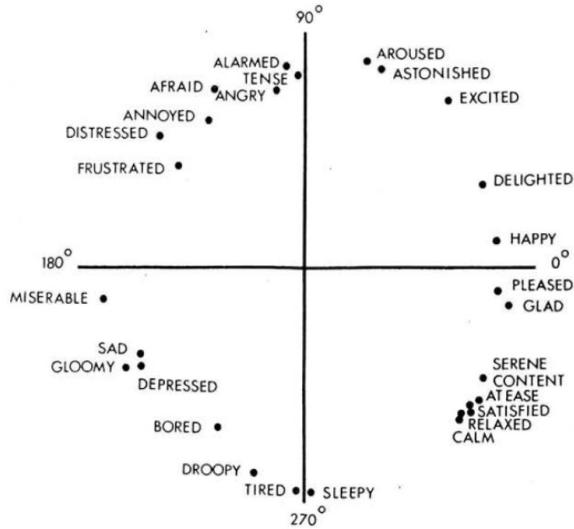
Figure 1. 2D circumplex model of emotions adapted from Russel [40].

3. Lexical resources

We use the Python NLTK library for most of the language-processing tools. For defining word associations and finding related expressions, we use the well-known WordNet [30] lexical database containing a hierarchical structure of syntactically related nouns, verbs, adjectives, and adverbs *grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept.* We also use statistical NGram models trained with the Brown Corpus [16] to define word probabilities and co-occurrences. For counting the number of syllables and finding rhyming phrases, we incorporate the CMU Pronouncing Dictionary with the phonetic notation of English words.

For sentiment analysis, we use the ANEW database [5] consisting of nearly 2500 words rated in terms of pleasure, arousal, and dominance to define the emotional value of any given text.

To extract the sentiment evaluation, we use the *Sentistrength* [42] sentiment analysis tool. This estimates the negative and positive sentiment values in short, informal texts (rating both positive and negative scores with a 1–5 scale), considering common and slang words, emoticons, and idioms. The base of the algorithm is the *sentiment word-strength list* containing terms with a 2–5 scale of positive or negative evaluation. The initial, manually prepared word sentiment list has been optimized by a training algorithm to minimize the classification error for some training texts. The system also considers a spelling-correction algorithm and a *booster word list* with terms that can increase or decrease the score of other words (such as *very, extremely*), as well as a *negating word list* with terms that invert the emotion value (*not, never*). Addition-

ally, the algorithm uses a list of emoticons commonly used in social web texts and considers some other stylistic parameters (such as interrogative and repeated letters).

We also use the WordNet-Affect lexical resource [41] to build a hierarchy of words describing emotional states that are used later to generate the affective content of poems. The lexicon contains WordNet hyponyms of the word *emotion*, which is *a subset of synsets suitable to represent the affective concepts correlated with affective words.* Thus, for example, we can derive a correlated set of words containing *forgive, merciful, excusable, affectionate, commiserate, tender* for the emotional category *compassion*.

4. System implementation

The system structure is based on the blackboard model and consists of a group of experts representing diverse sources of knowledge, the common blackboard workspace, and the control component. The system architecture is presented in Figure 2 and described in detail in the following sections. The algorithm for generation is explained in detail in the *poetry generation algorithm* subsection, along with an illustrative example of the generation process.
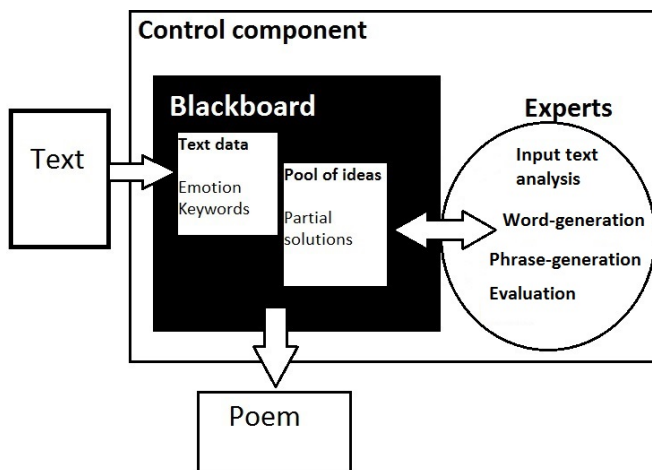

Figure 2. Blackboard architecture used in the system. A group of specialized experts work on the common blackboard workspace. The control component chooses one of the competing experts that will contribute to the final solution.

4.1. Blackboard

The blackboard is a common workspace shared by the experts containing partial solutions and other information about the problem. It is used by the experts for indirect communication. The blackboard area is divided into subareas representing distinct levels of abstraction. The main elements of the blackboard are as follows:

Text The input text used as inspiration for the poem. The experts analyze it to define the main theme and sentimental content for the poem.

Constraints The initial constraints and information about the poem. In the example, we use constraints on the number of lines, the number of syllables in each line, and the grammar constraints on tense and person to ensure grammatical consistency of the poem. These constraints are set by the user at the beginning of the process or chosen randomly by the system from a predetermined set.

Key phrases Most frequent noun phrases retrieved from the text by one of the experts. Each phrase has its *inspiration* value defined by $W * Cat$, where W is the number of words that the experts can generate from this phrase and Cat is the number of non-empty categories of words (such as the different parts of speech).

Topic The main theme for the poem selected from the key phrases is the phrase with the highest *inspiration* score. If there are more phrases with the same value, one of them is selected at random. Once the topic is set, the experts start to produce their phrases associated with it.

Emotion The emotional state for the poem defined by an expert by analyzing sentiments in the sentences (from the text) containing the *topic* phrase.

Pool of ideas A part of the blackboard that is used as a workspace for experts. It contains all of the words and partial solutions generated by them. It is also a source of inspiration, as some experts use artifacts generated by other experts to produce new artifacts. The expressions in the pool are divided into categories based on their grammatical forms and meanings.

Poem draft Current version of the poem consisting of lines that are chosen by the *selection experts* from the candidate *phrases*.

4.2. Experts

Experts are independent modules representing distinct knowledge sources that have access to the common blackboard. They are triggered by events on the blackboard; when they find something that they can use, they add new information to the blackboard. Each expert has some specific knowledge about the poetry-generation process, and they cooperate and compete with each other by playing different roles in the system. The experts work on the blackboard data at different abstraction levels. Specific experts grouped by the levels of abstraction that they utilize are presented below.

Text-analysis experts Experts that retrieve information from the initial text and add their data to the blackboard.

- Keyword expert – Extracts the most frequent noun phrases from the text.
- Emotion expert – Determines the emotional state for the poem by posting an *emotion* appropriate to the input text on the blackboard. As the whole text may be long and emotional attitude may vary within it, the sentiments are considered only for sentences containing the topic of the poem. Sentiments are calculated in terms of *valence* (positive/negative evaluation of pleasure scaled from -5 to 5) and *arousal* (passive/active scaled from -5 to 5) levels.

The valence of the text is calculated by using the SentiStrength tool, which estimates the negative and positive sentiment strength in sentences based on the *sentiment word strength list* containing words with positive and negative emotional values evaluated by human judges [42]. However, as we would like to incorporate an independent emotional intelligence in our system, we introduce another parameter called the *optimism rate*, which can be set by the user at the beginning of the algorithm or chosen randomly. The optimism rate biases the valence results (so that the perception of the input text may be more optimistic or pessimistic) as follows:

$$V = \alpha_{opt} \cdot \sum_{s \in Text} Sent_{pos} + (2 - \alpha_{opt}) \cdot \sum_{s \in Text} Sent_{neg} \qquad (1)$$

where $\sum_{s \in Text} Sent_{pos}$ and $\sum_{s \in Text} Sent_{neg}$ is the sum of positive and negative sentiments, respectively, for all sentences in the text, α_{opt} is the optimism rate of the system (between 0.7 and 1.3) that biases positive emotional value and $2 - \alpha_{opt}$ is the bias for negative one. The range (0.7, 1.3) has been selected somewhat arbitrarily so that the final state was affected by the optimism rate value but not completely dominated by it.

The arousal value is calculated with the use of ANEW. The algorithm combines the average ANEW arousal value for the words in the input text as follows:

$$A = \left(\sum_{w \in Text} A_{ANEW}(w) \right) \Big/ length(Text) \qquad (2)$$

where $A_{ANEW}(w)$ is the arousal value of the word w retrieved from the ANEW database. However, the sentiment in the text may be expressed not only through words but also by other features of the text, similar to expressing emotions with voice intonation in a spoken message. For example, the text *"That's great..."* can be perceived as less arousing than the same words written in a different way: *"That's GREAT!!!"*. Hence, the arousal calculation uses a punctuation-sensitive algorithm; i.e., some punctuation marks in the text increase the arousal value while others decrease it. The calculated arousal score is modified according to the following rules:

$$f(A) = \begin{cases} A - 1 & \text{if ``...'' in text} \\ A + 1 & \text{if ``!'' in text or word in capitals in text} \\ A + 2 & \text{if ``!!!'' in text} \end{cases} \qquad (3)$$

where A is the arousal text.

Once the valence and arousal of the text are calculated, the emotional state is defined as follows:

$$emotion = \operatorname*{arg\,min}_{x \in S} d((v_t, a_t), (v_x, a_x)), \qquad (4)$$

where *emotion* is the current emotional state, S is the set of all emotional states from the *model of emotions*, v_t and a_t are the valence and arousal of the text, v_x and a_x are the valence and arousal of the emotional state, and $d(x_1, x_2)$ is the Euclidean distance.

The model of emotions is a two-dimensional model where each emotional state is represented by a point (pair of coordinates) in the *(valence, arousal)* space (Fig. 3). The emotions used in the model are WordNet hyponyms of the word *emotion* used in the WordNet-Affect lexicon in the hierarchy of emotional categories. The *(valence, arousal)* coordinates for emotional labels in the model are taken from the ANEW database. The choice of emotional categories is based on the lexical resources that we use. It is possible to improve the model by rearranging the categories or their spatial coordinates or to use other more-complex models of emotions.
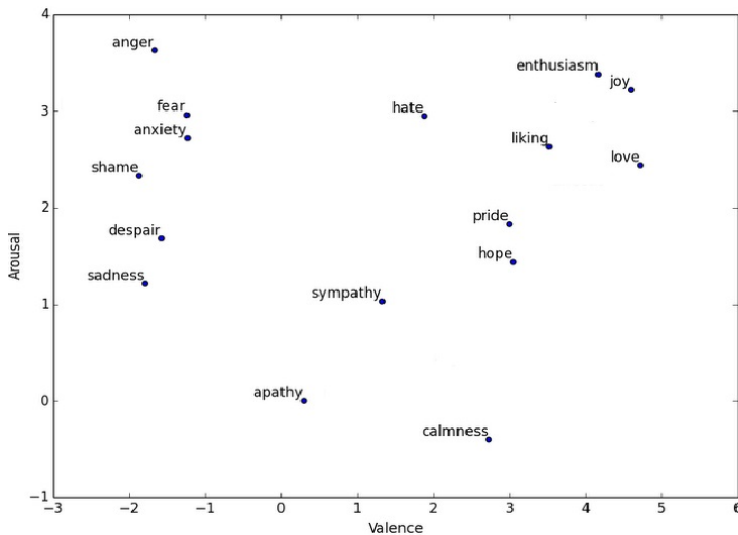

Figure 3. Emotional model used in the system. The labels are WordNet hyponyms of the word *emotion* derived from the WordNet-Affect lexicon.

Words-generation experts have lexical knowledge: they generate words associated with the topic and add them to the *pool of ideas* sections.

- The WordNet Expert generates synonyms, hypernyms, and antonyms for nouns and adjectives based on the WordNet lexical resource [30].
- The Collocation Expert generates words that are frequently used together with the given nouns and adjectives. It retrieves information from the bi-gram model of texts from the Brown Corpus, searching for adjectives that describe nouns, verbs that follow nouns, and nouns that follow adjectives.

- The Emotional-Word Expert generates words that describe the emotional state defined for the poem. The affective words are derived from WordNet Affect as the hyponyms of the given category name. For instance, if the emotional state was defined as *calmness*, the generated set of words would contain *peace, calm, tranquilly, easiness, cool, still*.

Poem-making experts produce new lines for the poem. They use partial solutions generated by other experts in the *pool of ideas* to produce new phrases. Their outputs are evaluated by the *selection experts*. They can generate a number of phrases proportional to their *importance factors*, which are set manually at the beginning of the algorithm. Some of these experts compose stylistic forms that are typical of poetry. They use Context-Free Grammar rules or statistical-language models to construct phrases.

- The Apostrophe Expert generates apostrophes[1] with the noun, its description, and hypernym; for example, *O life the heavenly being*.
- The Comparison Expert generates comparisons for adjectives using nouns that are most frequently described by them; for example, *As deep as a transformation*.
- The Epithet Expert generates expressions with a noun and its epithets or emotional adjectives; for example, *marvelous sophisticated fashion*.
- The Metaphor Expert generates metaphors by comparing the person to an object; for example, *You were like the downtalking style*.
- The Oxymoron Expert composes phrases with antonym words; for example, *good and bad*.
- The Rhetorical expert composes rhetorical questions about a noun, or a noun and its epithets; for example, *Why was the style so peculiar?*
- The Sentence Expert generates sentences according to grammar rules using all the word categories, and the emotions-describing words; for example, *She loved the peaceable New York*.
- The Exclamation Expert generates a new phrase by adding "!" exclamation mark to the phrase from the pool.
- The Repetition Expert generates a new phrase by repeating a phrase from the pool.

Selection Experts select the best solutions according to the given constraints and heuristics.

- The Inspiration Expert selects the topic for the poem from the set of key phrases according to the following formula:

$$Topic = \underset{x \in Keyphrases}{\arg\max} \; W_x \cdot Cat_x, \tag{5}$$

---

[1]An apostrophe is a figure of speech in which the poet addresses an absent person, an abstract idea, or a thing (definition from `http://grammarist.com/literature/apostrophe-poetry/`).

where W_x is the number of words that the experts can generate from this phrase and Cat_x is the number of non-empty categories to which these words belong (such as different parts of speech).

- The Syllable Expert selects phrases that have the number of syllables closest to the target number of syllables for the current line in the poem. The syllables are counted using the CMU Pronouncing Dictionary combined with the syllable-estimating algorithm used for words that are not included in the dictionary.

- The Rhyme Expert selects the best rhyming phrases by comparing the syllables in the words starting from the end of phrases and counting how many syllables correspond. It uses the CMU Pronouncing Dictionary.

4.3. Control component

The main roles of the control component are to select the focus of attention on particular levels of the generation process and to schedule the activation of experts according to the predefined constraints.

In the first phase of text analysis, key phrases are extracted and rated according to their inspiration value (meaning how many words may be associated with them) using the agenda-based control mechanism. In this phase, the focus of attention is determined: this is the key phrase that will be considered as the topic and the main theme for the final poem. We use a depth-first, event-scheduling approach to determine this choice; i.e., only one phrase is selected, and all further generation is based on this expression. This approach simulates an *inspiration* of the system: after reading the text, it decides which theme seems particularly interesting and embarks on making a poem based on this inspiration. Eliminating other solution paths significantly improves the efficiency of the system; however, some modifications could be further explored (for example, considering more than one key phrase would allow a higher diversity in generated outputs).

In the following steps of generation, the system expands the solution space by allowing the experts to produce numerous word associations, expressions, and phrases. Following the breadth-first-search approach, the experts produce several partial solutions on different levels of abstraction, resulting in a large set of diverse ideas and inspiring other experts for their creative outputs. No solutions are eliminated at this point, allowing a divergent solution space to evolve in an unrestrained way (which is the basis for creativity).

For the experts working at the level of control data, an incremental search strategy is used. A part of the control section of the blackboard contains the set of constraints for the final output. These are applied by a specialized group of experts, each of which possesses knowledge about a particular constraint. The experts work sequentially, and each of them rates the new phrases with its own evaluation function, eliminating less-suitable ones. The following control experts work on the set of expressions pruned by their predecessors, so the order in which experts are deployed

strongly affects the final form of the poem. The role of control component is to apply a knowledge-scheduling approach according to a set of predefined preferences. This approach allows for a flexible control of constraints for the output, including some customized parameters.

4.4. Poetry generation algorithm

The general algorithm for the poetry generation is as follows:

Algorithm 1: Poetry generation algorithm

Data: Input text
Result: Poem

1 Component initialization;
2 Keyword expert: Key phrase extraction;
3 Emotional expert: Emotional state definition;
4 Inspiration expert: Poem topic selection;
5 **while** *current line number < required line number* **do**
6   Word-generation experts: generate words associated with the topic;
7   Poem-making experts: construct expressions from words on the blackboard;
8   Selection experts: rate phrases according to the constraints for the current line;
9   Control component: schedule experts;

We now present an example of the generation process.

1. Component initialization

   - Blackboard is initialized with the text input by the user. The form of the poem is selected randomly from a set of templates, and the grammar constraints are defined for stylistic consistency. The selection of constraints may be performed automatically by the system or predefined by the user.
     Text

     When someone leaves you, apart from missing them, apart from the fact that the whole little world you've created together collapses, and that everything you see or do reminds you of them, the worst is the thought that they tried you out and, in the end, the whole sum of parts adds up to you got stamped REJECT by the one you love. How can you not be left with the personal confidence of a passed over British Rail sandwich?

     Constraints

     Number of syllables in lines: (8,8,8,8)
     Grammar form:

         – Person: *she*;
         – Tense: *present*;

- Poem-making experts are initialized with individual *importance factors* varying from 1 to 5, meaning how many phrases they can generate during each turn. The default values presented below were tuned empirically and may be individually modified by the user.

  Poem-making expert importance factors

    – *Apostrophe expert – 2*
    – *Comparison expert – 3*
    – *Epithet expert – 5*
    – *Metaphor expert – 2*
    – *Oxymoron expert – 2*
    – *Rhetorical expert – 3*
    – *Sentence expert – 5*
    – *Anaphora expert – 1*
    – *Epiphora expert – 1*
    – *Exclamation expert – 1*
    – *Repetition expert – 1*

- Emotional expert is initialized with a random *optimism factor* between 0.7 and 1.3. A higher value means a more-optimistic attitude.

  Optimism factor: 0.84

2. Topic selection

   The topic is chosen as the most *inspiring* key phrase from the text. To define it, all key phrases are first retrieved and evaluated with their *inspiration* score.

   - *Keyword expert* extracts key phrases as the most frequent phrases consisting of a noun and descriptive adjectives.
     Key phrases:

     [someone, end, whole little world, whole sum, British rail sandwich, parts, personal confidence, fact]

   - *Word-generation experts* estimate how many words they can produce from each key phrase. The *inspiration* for each phrase is calculated according to formula (5). The *inspiration expert* selects the *most inspiring* phrase for the topic.
     Inspirations:

       – *whole little world: 6920,*
       – *personal confidence: 3920,*
       – *whole sum: 3880,*
       – *someone: 2324,*
       – *parts: 1918,*
       – *fact: 1512.*
       – *end: 910,*

Poem topic:

Whole little world

- *Emotional expert* defines the emotional state for the poem. The sentiments are retrieved from sentences containing the topic phrase. The expert calculates *valence* and *arousal* according to (1), (2), and (3). Then the emotional state is defined as in (4).
Valence: −0.94
Arousal: 2.0
Emotional state: despair

3. Word generation
Once the topic and emotional state for the poem are fixed, the *word-generation experts* start to produce their ideas. They store their outputs in the appropriate categories in the *pool of ideas* section of the blackboard.
Pool of ideas:

- Nouns – *[macrocosm, existence, universe, cosmos, world, creation]*
- Adjectives – *[whole, little, small]*
- Verbs – *existence: [loses, makes, reflects, becomes, fails, is, belongs], world: [centered, admired, rendered, ...], cosmos: [took], universe: [is, had, are, was, were], creation: [is, does, prevents]*
- Epithets – *world: [little, contemporary, real, previous, ...] , existence: [happy, celestial, historical, ...], cosmos: [Jewish], universe: [interdependent, entire, ...], creation: [own, inventive, artistic, ...]*
- Comparisons – *whole: [lines, block, incident, country, ...], ...*
- Hypernyms – *existence: [state], world: [natural object], creation: [activity], ...*
- Antonyms – *whole: [fractional], little:[big]*
- Emotional words – *[pessimistic, cynical, resignation, discourage, hopeless, ...]*

4. Phrase generation
As the words appear in the *pool of ideas*, the *poem-making experts* start to produce phrases for new lines according to the grammar constraints. They add their outputs to the *phrases* section.
Phrases:

- Epithet Expert: *corporate existence, great world*
- Apostrophe Expert: *oh world the little natural object*
- Sentence Expert: *the creation prevents abjectly, she likes the hopeless, she loves the pessimistic cosmos*
- Comparison Expert: *as whole as a story, whole like a convocation*
- Metaphor Expert: *she is like the human existence*
- Exclamation Expert: *as whole as a story!*
- Rhetorical Expert: *why is the existence so nonfunctional?*

- Oxymoron Expert: *whole but fractional*
- ...

5. Line phrase selection

When all experts finish their generation, the phrases that best fulfill the line constraints are selected by *selection experts*. Then, the *control component* makes the final selection, judging by the experts' frequencies in former lines. The same algorithm is repeated for each line of the poem.

Generating line 4.

> *Poem:*
>
>> line 1: *what is the Jewish cosmos?* (Rhetorical Expert)
>> line 2: *o existence the daily state* (Apostrophe Expert)
>> line 3: *perceptual physical world* (Epithet Expert)
>
> *Target syllables number:* 8
> *Syllable expert* – best phrases candidates:
>
>> *happy corporate existence* (Epithet Expert) : 8,
>> *she sees the pessimistic world* (Sentence Expert): 8,
>> ...
>
> *Control component* – selecting less active experts in former lines generation:
>
>> Epithet Expert: 1 line,
>> Sentence Expert: 0 lines,
>> ...
>
> *Line phrase selection:*
>
>> *she sees the pessimistic world* (Sentence Expert)

5. Results and evaluation

Below, we present some example outputs generated by our system inspired by various input texts. We add some remarks on the interpretation of the generated poems, which are further analyzed in the *Evaluation* section.

5.1. Results

1. Compassionate poem about life
   Inspired by the text:

   > *With the holiday craziness yesterday, and having to work, I didn't get to finish posting all of my thankfulness pictures. So you might see them pop up over the next few days. This morning I am thankful for the adult men in my life. My dad and Mr P. I am fortunate to have both*

> *of them in my life to encourage me, support me, take care of me, and*
> *love the kids with all of their hearts*[2].

Topic:

> *Life*

Emotion:

> *compassion*

Poem:

> *O life the personal beingness*
> *You are like the simple life!*
> *Musical sacrificial life*
> *You are like the general life*
> *You see the excusable life*
> *Emotional musical life*
> *O life the heavenly being*

Remarks:

The topic *life* provided a wide range of epithets associated with the main phrase, which creates the poetic style. Another observation is that all of the lines correspond to the title phrase *life*; however, there is no association among the lines, and so the final poem seems to lack continuity.

2. Angry poem about the end

Inspired by the text:

> *I remember being endlessly entertained by the adventures of my toys!*
> *Some days they died repeated, violent deaths, other days they traveled*
> *to space or discussed my swim lessons and how I absolutely should be*
> *allowed in the deep end of the pool, especially since I was such a talented*
> *doggy-paddler*[3].

Topic:

> *Deep end*

Emotion:

> *anger*

Poem:

> *I knew the undisrupted end*
> *I was like the various end*
> *As deep as a transformation*
> *O end the left extremity*
> *Objective undisrupted end*

---

[2] https://phrogmom.wordpress.com
[3] http://hyperboleandahalf.blogspot.com

> *I hated the choleric end*
> *O end the dead extremity*

Remarks:
The emotional state for the poem is *anger*, which corresponds to some negative expressions in the text (*died, violent deaths, deep end*). The mood is expressed in the poem by words *choleric* and *hated*.

3. Anxious poem about a theater
   Inspired by the text:

   > *The show with KFrog out in Temecula California was incredible. It was in a huge theater at the Pechanga Casino. The show was for KFrog Cares which raises money for Loma Linda University Children's Hospital. It was us with Katie Arminger and Jana Kramer was supposed to headline, but she got sick. So that meant we were the headliners! And that crowd could not have been more gracious, rowdy, or amazing! We had the BEST show. And we REALLY needed that. It was crazy because we sold out of ALL of the merch we brought! T-shirts and Eps flew out the door! That was an amazing feeling[4].*

   Topic:

   > *Huge theater*

   Emotion:

   > *anxiety*

   Poem:

   > *Great big theater*
   > *Vast like a majority*
   > *Why is the theater so entire?*
   > *She needs the insecurity*

   Remarks:
   This poem was generated giving priority to the rhyming measure. The emotional state of the poem (anxiety) is different from the sentiments in the input text.

4. Fearful poem about the way
   Inspired by the text:

   > *Lately everyone has been wondering "Are Jenelle and Gary going to get back together?!" NO! He is living his life and I'm living mine. We are both happy with our lives the way they are at the moment, I know for me at least I'm EXTREMELY happy. Gary might of been tweeting things because he might of been jealous in a way that I was dating Courtland but he agrees to stop today[5].*

---

[4]http://www.nataliestovall.com
[5]http://starcasm.net

Topic:

> *Way*

Emotion:

> *fear*

Poem:

> *O mode the symbolic property*
> *Quickest moderate way*
> *She was like the mode*
> *She seemed hysterical because the way left*

Remarks:

In this poem, we can observe that the system does not handle ambiguous words well. The *way* is once interpreted as *property* or *mode*, but the algorithm does not consider the context of the phrase in the text. However, the poetry may allow for some less strict interpretations of meaning, as the ambiguity can be used as an intentional poetical operation.

5. Calm poem about New York

Inspired by the text:

> *Daniel and I felt so, so very fortunate to go here. The timing could not have been more perfect for us as our last few days in New York had been quite hectic. We had been trying to wrap our heads around our lives without any more fertility treatments*[6].

Topic:

> *New York*

Emotion:

> *calmness*

Poem:

> *she loved the peaceable New York*
> *new like a revision*
> *she was like the york*
> *she needed the easy house of york!*

Remarks:

This example illustrates how the cooperation of experts within a blackboard system may lead to creative solutions. The well-known name *New York* has been broken up into words and processed separately. The analysis on various abstraction levels provides an original and uncommon interpretation of the topic phrase.

---

[6]`http:\\storyofmylifetheblog.blogspot.com.es/`

5.2. System evaluation

The evaluation of a computational creativity system is a difficult and ill-defined problem. Some methods for evaluating computational creativity as well as their results for particular systems (and in the domain of poetry generation) are presented and compared by Anna Jordanous [25].

In order to make a comparative evaluation, we used Ritchie's empirical criteria [39] to evaluate the outputs and compared our results with those presented by Pablo Gervas and Francisco Pereira et al. [18, 37] for the WASP poetry generator. We also evaluated the creative behavior of our system by incorporating the FACE model [9].

Alison Pease and Simon Colton [35] argued that the *Turing test* for poetry (as applied by Ray Kurzweil [28]) is not a reliable measure of its creativity. They note that the current versions of such tests consider only the human-style creativity, and the evaluation is prone to subjective biases and is therefore inappropriate for evaluating creativity of a computational system.

However, we argue that the reader's perception is a factor that should be considered during the poetry-creation process [22]. Hence, we use the results of an experiment performed on a group of students to evaluate the aesthetics and quality according to Ritchie's empirical criteria.

5.2.1. Empirical criteria

We use Ritchie's *empirical criteria* [39] on a set of 35 poems produced by the system to measure their creativity. According to these criteria, we first measure the *typicality* by verifying if the generated outputs can be classified as poems. Here, we incorporated Manurung's definition [29] of triple criteria for poetic artifacts; namely, meaningfulness, grammaticality, and individual poeticity.

We measured the *quality* of the generated poems by evaluating their aesthetic measures as provided by a group of student volunteers. According to Ritchie's definition, the *novelty* of artifacts is a measure of their similarity to the representatives of the same genre from the *inspiration set*. Since we do not use any poems as the training corpora in our algorithm, this measure is not applicable to our work. Hence, criteria 9–18 are not suitable for the evaluation of our system.

Below, we explain the evaluation process used in detail to measure two of Ritchie's criteria:

- Typicality: According to Ruli Manurung [29], in order for an artifact to be classified as a poem, it must satisfy the three constraints of meaningfulness, grammaticality, and poeticity. We evaluated the grammar and meaningfulness individually by verifying that each line has a proper grammatical structure and is meaningful. We scored each poem on a scale [0–2] for meaningfulness and grammar (0 for incorrect, 2 for completely correct). Then, we evaluated its poeticity considering rhymes (number of rhyming lines), syllables (number of syllables according to the template), poetical expressions (diversity of poetical forms within the poem), and emotionality (emotional expressions within the poem). We scored each poetical

criterion [0–2] individually, and took the average score across all the criteria for each poem as the overall measure of its poeticity. Then, we calculated the final value of *typicality* as the average across poeticity, grammar (considering number of ungrammatical expressions), and meaningfulness (judging if the lines make some sense), and normalized the score to (0,1) interval.

- Quality: We calculated the quality of the generated poems as the average subjective aesthetic measure assigned by a group of volunteers as described below and normalized the score to $(0, 1)$ interval.

To evaluate the aesthetic quality of the generated poems, we performed an experiment with a group of 15 participants. The participants received a list of 35 poems generated by the system without any information on how the poems were created. The participants were then asked to score each poem for its aesthetic quality on a 1–5 scale (1 meaning a poor quality poem and 5 meaning a very good poem). The histogram for all of the poem scores is presented in Figure 4.
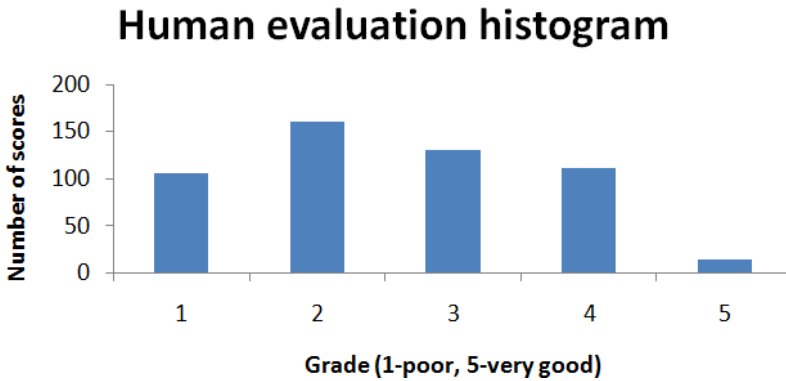

Figure 4. Histogram of reader scores during evaluation of poems.

The *threshold* is a score indicating the acceptable value of typicality or quality of items. According to Graeme Ritchie [39], in order to perform a comparable evaluation, the *threshold* value must be suitably defined. The threshold selection is not a trivial task and may strongly influence the final result of the evaluation.

The evaluation was performed on a set of 35 poems generated by the system. The experiments were made with varying threshold values for the typicality and quality measurements, corresponding to the *high, medium, and low* values defined by Pablo Gervas [18]. We considered Ritchie's criteria 1–8 [39] that refer to typicality and quality measures, as the others were not applicable (as explained above in Section 5.2.1). The results are presented in Table 1. In Table 2, we present the evaluation results for medium thresholds for our system and for the WASP system as presented by Pable Gervas and Francisco Pereira et al. [18, 37].

The results show that the average typicality of the poems with a medium threshold is relatively low (0.58). In other words, some outputs do not meet the criteria

required for poetry. However, the quality measures for atypical outputs show that, even though some of them cannot be considered as typical representatives of their genre (poetry), a few of them are still considered interesting. According to Margaret Boden [3], such examples have a higher potential for creativity. Nonetheless, the evaluation needs to be repeated by a larger group of independent judges on a bigger set of generated poems to make a better approximation.

As shown in Table 1, the overall evaluation strongly depends on the selection of threshold values. In this work, we chose the threshold values to be equal to the evaluation of WASP so that we can perform a reliable comparison. However, it might be good to readjust some of the thresholds to better correspond to the creative domain. For instance, threshold values could be adapted so that less-typical high-quality outputs receive a higher score in the overall evaluation. We observed that none of the outputs received average scores above the high threshold (0.7).

The comparison with WASP indicates that our system generates less-satisfactory results, according to the typicality criterion. As most of WASP's artifacts fall into the typical category, there are no good atypical outputs for a given threshold. As observed by Francisco Pereira et al. [37], this may be a result of misunderstanding the specification of evaluation criteria by the human judges. In our evaluation, we asked the volunteers to judge the quality of the poems according to their aesthetic preferences, while the correctness of the poems was assigned to the typicality measure (according to Manurung's definition). Since the measurements of typicality in both systems were performed in different ways, the results are not entirely comparable. Moreover, the quality of outputs deemed as aesthetic is also prone to subjective views, as the evaluation was performed by different judges.

Table 1

Ritchie's empirical criteria for the evaluation of outputs, with varying *typicality* and *quality* threshold measures: high (0.7), medium (0.5) and low (0.3). Scores are in range [0–1].

Thresholds	Low	Medium	High
Average typicality	0.58	0.58	0.58
Typical results/results	0.89	0.69	0.2
Average quality	0.48	0.48	0.48
Good results/results	0.83	0.57	0
Good typical results/typical results	0.87	0.62	0
Good atypical results/results	0.06	0.14	0
Good atypical results/atypical results	0.67	0.5	0
Good atypical results/good typical results	0.07	0.33	0

Analysis of the human-evaluation results revealed that readers usually pay more attention to the form of poems than to their linguistic diversity (e.g. rhyming poems receive higher scores). The participants also gave higher scores to poems containing well-known expressions than those having more innovative word combinations.

<div align="center">

Table 2

Comparing results of our system with WASP for medium thresholds values (0.5).

</div>

System	Blackboard system	WASP
Average typicality	0.58	0.71
Typical results/results	0.69	0.88
Average quality	0.48	0.47
Good results/results	0.57	0.5
Good typical results/typical results	0.62	0.57
Good atypical results/results	0.14	0
Good atypical results/atypical results	0.5	0
Good atypical results/good typical results	0.33	0

5.2.2. FACE model evaluation

As the main focus of computational creativity systems is to produce artifacts in an intentional way, the process of generation should also be considered an important parameter for evaluation. FACE [9] provides one such model of evaluation, which we applied to our system. A *creative act* that satisfies all FACE criteria is denoted by a tuple $\langle F^g, A^g, C^g, E^g \rangle$, where the *C (concept)* means the system taking input and producing outputs denoted by *E (expressions)*, the *A (aesthetic measure)* is the fitness function evaluating the (concept, expression) pairs with real-number values, and the *F (framing information)* is the linguistic comment explaining the context or motivation of the outputs.

- Concept and concept expression
  In our case, the *concept* is the blackboard architecture with a group of experts cooperating to compose the poem. The motivation to use the blackboard architecture (as presented in the *background* section) is the Global Workspace theory, which compares the brain functioning to a group of independent modules sharing a public workspace. Our system takes a piece of text as an input and produces the *concept expressions* in the form of poems. The evaluation of generated outputs is explained in the *results* subsection. In this approach, we could also consider each expert as an independent *concept* producing its own *expressions* as partial solutions for the problem.
- Aesthetic measure
  The *aesthetic measure* in the system is composed of heuristic functions that evaluate candidates for new lines in the poem. Each pair of experts (*concept*) – phrase (*expression*) is evaluated according to the stylistic constraints on syllables measure, rhyme, or emotional content, resulting in a value (a real number). Another measure is used for the topic selection – each key phrase is evaluated according to its *inspiration* value as in (5). However, the process lacks an evaluation of the poem as a whole – there is no overall aesthetic measurement. As we remarked in Section 5, even though all of the lines of the generated poems relate to one topic (the main theme), they do not always make overall sense, because the verses are

not always conceptually connected to each other. This aspect might be improved by the use of some chaining techniques to produce a sequence of related phrases and by increasing the score for the meaning in the evaluation criteria.

- Framing information

   The *framing information* in our system is to be found in the name of the emotional state defined according to the model of emotions (4). This parameter provides some information about the context of the poem but does not contain any extra explanation of why particular lines have been selected.

As described above, the generation process performs the *generative acts* of the form $\langle C^g, E^g \rangle$, meaning that the system performs the creative task by processing the textual input with contribution from various experts via the blackboard and generates a poem (an expression of a concept). F^g is provided only by the emotional state description, which partially satisfies the *framing information* criterion, while the aesthetics A^g is defined at a lower level of granularity (at the level of individual lines), but without considering the overall structure of the poem.

We would like to make the following observation in comparing our results with the Full-FACE poetry generating system [10], which generates poems (expressions) based on the templates (concepts) created by the system, chooses the metrics (aesthetics), and generates the commentary (framing information). The *aesthetic* measure provided in the Full-FACE system is applied to the metrics, such as rhyme, meter, and stress (all of which are also relevant for our system). Hence, we claim that A^g is applicable to our system as well, but both our system and Full-FACE lack an overall aesthetics measure for the whole poem.

A possible way to improve our system in this regard is to incorporate some experts that evaluate the poem as a whole and apply constraints at the holistic level (not only for particular lines); e.g. based on its semantic consistency. Another improvement would be to incorporate automatic generation of poem templates instead of a random selection from a predefined set. The rhythm and stress could be adjusted according to the emotional state defined for the poem.

6. Conclusions and future research

We proposed a blackboard system for generating poetry inspired by a given piece of input text. Generated poems contain phrases that express emotions extracted from the input text and other phrases typical for poetry. The blackboard architecture used in the system supports flexibility in generating a variety of solutions at different abstraction levels, leading to creativity and the simulation of inspirational and innovative processes in the mind. Similar to the Baars' Theater of Consciousness, the experts act out their roles on the scene of the blackboard, and their cooperative play leads to the final solution. The architecture is also extensible with new linguistic resources and stylistic constraints. The evaluation of the system's outputs according to the *empirical criteria* proposed by Graeme Ritchie [39] indicates that, even though some of the generated poems do not satisfy typicality, others qualify as high-

quality artifacts. According to Margaret Boden [3], such atypical high-valued outputs are a promising indication for the creativity potential of the system. However, the evaluation needs to be repeated by an expert group of objective judges to confirm the results.

As a continuation of this work, we would like to incorporate other poetry- and language-generating tools as new experts in the system. We would like to include some modules responsible for creating idea-chains and conceptual associations to provide a consistency of meaning within the poem as well as incorporate some metaphor-generating tools for increasing the diversity of expressions. The system can also be extended with additional evaluation experts that use more complex heuristics for selecting parts of the poem and perform an overall evaluation of the poems.

Another interesting extension of this project would be to develop an open-source platform based on the blackboard architecture that the users can extend by adding new experts to the poetry-generating process. Such a system could allow researchers from different domains to collaborate on the design of a complex poetry-generating system. It would also be possible to make an interactive platform to allow human experts to participate in the generation process and add their phrases to the blackboard. The program could visualize the generation process by displaying some of the data appearing on the blackboard during the work by experts. This could provide us with a window to observe the human creative process in action and deepen our understanding of human creativity.

We believe that the blackboard architecture has the potential to contribute to the field of computational creativity and provide a way to combine many of the existing approaches in this domain. We hope that our experience with the system presented here provides evidence to support this claim.

References

[1] Baars B.J.: In the Theater of Consciousness: The Workspace of the Mind. *Journal of Consciousness Studies*, vol. 4, 1997.

[2] Baars B.J.: The global brainweb: An update on global workspace theory. *Science and Consciousness Review*, 2003.

[3] Boden M.A.: *The creative mind*. London: Abacus, 1992.

[4] Boden M.A.: Computer Models of Creativity. *AI Magazine*, vol. 30(3), pp. 23–34, 2009, `http://www.aaai.org/ojs/index.php/aimagazine/article/view/2254`.

[5] Bradley M.M., Lang P.J.: Affective Norms for English Words(ANEW): Affective ratings of words and instruction manual. Tech. rep., NIMH Center for the Study of Emotion and Attention University of Florida, 2010.

[6] Cambria E., Livingstone A., Hussain A.: *The Hourglass of Emotions*. Springer, 2012.

 [7] Carver N., Lesser V.: The Evolution of Blackboard Control Architectures. Tech.
     rep., CMPISCI, 1992.

 [8] Charnley J., Colton S.: *A Global Workspace Framework for Combining Reasoning
     Systems*, pp. 261–265. Springer, 2008.

 [9] Colton S., Charnley J., Pease A.: Computational Creativity Theory: The FACE
     and IDEA Descriptive Models. In: *Proceedings of the International Conference
     on Computational Creativity*, 2011.

[10] Colton S., Goodwin J., Veale T.: Full-FACE Poetry Generation. In: *Proceedings
     of the International Conference on Computational Creativity*, 2012.

[11] Corkill D.D.: Blackboard Systems. *AI Expert*, vol. 6, 1991.

[12] Corkill D.D., Lesser V.R., Hudlicka E.: Unifying Data-Directed and Goal-
     Directed Control: An Example and Experiments. *AAAI-82*, 1982.

[13] Darwin C.: *The expression of the emotions in man and animals / by Charles
     Darwin*. D. Appleton and Co., New York, 1916.

[14] Ekman P.: *Basic Emotions*. John Wiley & Sons, Ltd., 1999.

[15] Erman L., Hayes-Roth F., Lesser V., Reddy D.: The Hearsay-II Speech-
     Understanding System: Integrating Knowledge to Resolve Uncertainty. *ACM
     Computing Surveys*, 1980.

[16] Francis W.N., Kucera H.: Brown Corpus Manual. Tech. rep., Brown University,
     1979.

[17] Gervás P.: An expert system for the composition of formal Spanish poetry.
     Knowledge-Based Systems, 2001.

[18] Gervás P.: Exploring Quantitative Evaluations of the Creativity of Automatic
     Poets. In: *2nd workshop on creative systems, approaches to creativity in artificial
     intelligence and cognitive science, ECAI 2002*, Lyon, France, 2002.

[19] Gervás P.: Engineering Linguistic Creativity: Bird Flight and Jet Planes. In:
     *NAACL HLT 2010 Second Workshop on Computational Approaches to Linguistic
     Creativity*, 2010.

[20] Hayes-Roth B.: A blackboard architecture for control. *Artificial Intelligence*,
     1985.

[21] van der Heide A., Trivino G.: Simulating emotional personality in human com-
     puter interfaces. In: *Proceedings of IEEE International Conference on the Fuzzy
     Systems (FUZZ)*, 2010.

[22] Indurkhya B.: Whence is creativity? In: *Proceedings of the Third In-
     ternational Conference on Computational Creativity*, pp. 62–66, Dublin, Ire-
     land, 2012, `http://computationalcreativity.net/iccc2012/wp-content/
     uploads/2012/05/062-Indurkhya.pdf`.

[23] Indurkhya B.: On the Role of Computers in Creativity-Support Systems. In:
     T. Besold, M. Schorlemmer, A. Smaill, eds., *Computational Creativity Research:
     Towards Creative Machines*, pp. 109–125, Paris: Atlantis Press, 2015.

[24] Jefferson G.: The Mind of Mechanical Man. *British Medical Journal*, vol. 4616(1), pp. 1105–1110, 1949.

[25] Jordanous A.: *Evaluating computational creativity: a standardised procedure for evaluating creative systems and its application*. Ph.D. thesis, University of Sussex, 2012.

[26] Keith S.R.: *Explaining creativity: The Science of human innovation*. Oxford University Press, 2012, `http://bangor.eblib.com/patron/FullRecord.aspx?p=886582`.

[27] Kirke A., Miranda E.: Emotional and Multi-agent Systems in Computer-aided Writing and Poetry. In: *Proceedings of the Artificial Intelligence and Poetry Symposium*, 2013.

[28] Kurzweil R.: *A (Kind of) Turing Test*. The MIT Press, 1992.

[29] Manurung R., Ritchie G., Thompson H.: Using Genetic Algorithms to Create Meaningful Poetic Text. *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 24, pp. 43–64.

[30] Miller G.A.: WordNet: A Lexical Database for English. *Communications of the ACM*, 1995.

[31] Misztal J., Indurkhya B.: A Blackboard Architecture for automatic poetry generation. In: *Proceedings of the Workshop Computational Creativity, Concept Invention, and General Intelligence at 21st European Conference on Artificial Intelligence*, 2014.

[32] Misztal J., Indurkhya B.: A computational approach to re-interpretation Generation of emphatic poems inspired by Internet Blogs. In: *Proceedings of AAAI Fall Symposium on Modeling Changing Perspectives: Re-conceptualizing sensory-motor experiences*, 2014.

[33] Misztal J., Indurkhya B.: Poetry generation system with an emotional personality. In: *Proceedings of the 5th International Conference on Computational Creativity*, 2014.

[34] Montfort N., Pérez y Pérez R., Harrell D.F., Campana A.: Slant: A Blackboard System to Generate Plot, Figuration, and Narrative Discourse Aspects of Stories. In: *Proceedings of the Fourth International Conference on Computational Creativity*, 2013.

[35] Pease A., Colton S.: On Impact and Evaluation in Computational Creativity: A Discussion of the Turing Test and an Alternative Proposal. In: *Proceedings of the AISB symposium on AI and Philosophy*, 2011.

[36] Penrose R.: *The Emperor's New Mind: Concerning Computers, Minds, and the Laws of Physics*. Oxford University Press, Inc., New York, 1989, ISBN 0-19-851973-7.

[37] Pereira F.C., Mendes M., Gervás P., Cardoso A.: Experiments With Assessment of Creative Systems: An Application of Ritchie's Criteria. In: *Computational Creativity Workshop 2005*, International Joint Conference on Artificial Intelligence 2005, University of Edinburgh, Edinburgh, Scotland, 2005.

[38] Plutchik R.: The nature of emotions. *American Scientist*, vol. 89(4), pp. 344–350, 2001.

[39] Ritchie G.: Some Empirical Criteria for Attributing Creativity to a Computer Program. *Minds and Machines*, vol. 17(1), pp. 67–99, 2007.

[40] Russel J.A.: A circumplex model of affect. *Journal of personality and social psychology*, 1980.

[41] Strapparava C., Valitutti A.: WordNet-Affect: an Affective Extension of Word-Net. In: *Proceedings of the 4th International Conference on Language Resources and Evaluation*, 2004.

[42] Thelwall M., Buckley K., Paltoglou G., Cai D.: Sentiment Strength Detection in Short Informal Text. *Journal of the American Society for Information Science and Technology*, 2010.

[43] Toivanen J.M., Gross O., Toivonen H.: The Officer Is Taller Than You, Who Race Yourself! Using Document Specific Word Associations in Poetry Generation. In: *Proceedings of the 5th International Conference on Computational Creativity*, 2014.

[44] Toivanen J.M., Jarvisalo M., Toivonen H.: Harnessing Constraint Programming for Poetry Composition. In: *Proceedings of the International Conference on Computational Creativity*, 2013.

[45] Toivanen J.M., Toivonen H., Valitutti A., Gross O.: Corpus-Based Generation of Content and Form in Poetry. In: *Proceedings of the International Conference on Computational Creativity*, 2012.

[46] Turing A.: Computing Machinery and Intelligence. *Mind*, 1950.

[47] Veale T.: Less Rhyme, More Reason: Knowledge-based Poetry Generation with Feeling, Insight and Wit. In: *Proceedings of the International Conference on Computational Creativity*, 2013.

[48] Whissel C.: The dictionary of affect in language. *Emotion: Theory, Research, and Experience*, 1989.

[49] Wiggins G.A.: The Mind's Chorus: Creativity Before Consciousness. *Cognitive Computation*, 2012.

Affiliations

Joanna Misztal-Radecka
    Jagiellonian University, Krakow, Poland, `asiamsztl@gmail.com`

Bipin Indurkhya
    Jagiellonian University, Krakow, Poland, `bipin@agh.edu.pl`