# Difficulties and Challenges in Automatic Poem Generation: Five Years of Research at UCM

Pablo Gervás, Raquel Hervás, Jason R Robinson

Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática – Universidad Complutense de Madrid
c/ Profesor José García Santesmases, s/n, Madrid 28040
pgervas@sip.ucm.es,raquelhb@fdi.ucm.es
srjrobin@uga.edu

**Abstract.**
This paper reviews work related to automatic poetry generation at Universidad Complutense de Madrid (UCM) over the past five years. It outlines our initial motivation, the difficulties we met, the challenges we identified, the research work that we have done over the past few years, and the lines of work we currently see open before us.

## 1 Introduction

For five years now, the team of researches collaborating at Universidad Complutense de Madrid (UCM) under the NIL[1] initiative has been working on various attempts to model several of the elementary human abilities that collaborate in the process that enables a poet to generate acceptable verses. Initially, this line of work drew surprised exclamations from supervising bodies, funding agencies, and academic colleagues. In general terms, it was seen more as some kind of circus trick – a machine that writes poems – than as a serious research endeavour. One must confess that the kind of results we were obtaining at the time where not particularly impressive. The difficulties and challenges involved in automating the task of composing poetry were not evident to us at the start, and we had gaily embarked upon this work without realising the amount of effort that would be required before acceptable results could be obtained. In fact, if we had been aware of all the obstacles that lay along this path, we would probably have refrained from setting off on this journey. Our first efforts were poorly rewarded in terms of actual poetic results. But as a result of those early experiences, we discovered the need to decompose the task of writing a poem into a complex process involving several smaller tasks or abilities, so that each one might be modelled correctly before attempting to combine them in the composition of actual verse. The sad aspect of this discovery was that we identified so many subtasks that it was clearly impossible for the initial team to tackle them all. However, we set to work on the ones that seemed more straightforward, and we made progress by banging our heads against these supposedly easier subtasks. Slowly, more and more researchers joined the team, and this allowed us to progressively enlarge the set of subtasks under study. We are still far from having programs that are capable of generating 'good' poems. But the scientific community has had to accept that research on some of the issues that we took on with a view to generating poems has lead to results that have practical applications in real life beyond poetry.

[1] Natural Interaction based on Language, http://nil.fdi.ucm.es

This paper outlines our initial motivation, the difficulties we met, the challenges we identified, the research work that we have done over the past few years, and the lines of work we currently see open before us.

# 2. Why Attempt to Enable Machines to Write Poetry?

Two traits with which humans are exceptionally apt are abstraction and creativity. It is likely that these traits equipped us with a powerful tool for survival early on in our development. Primordial humans probably possessed the ability to recognize completely unfamiliar and new dangers through abstraction as well as the ability to imagine and project possible outcomes under different circumstances through creativity.

These survival abilities are well exercised in the process of creating poetry of any format. That is because the creation of poetry is a system of abstraction and recognition of patterns, and then the imagination, manipulation and projection of new patterns within an environment (in most cases, language, but also images, sounds, genetic material, or any other environment a poet chooses); and the appreciation of a poetry is the recognition --intuitively or not-- of the art: the process of abstraction, manipulation and projections of the patterns within the given environment.

Dissecting the ghost in the poem is a fun challenge for many computer scientists. Can we teach a computer to identify patterns that humans perceive intuitively? Can we teach a computer to manipulate and generate the patterns that humans appreciate?

We have already established that computers are well suited for the abstraction, manipulation and projections of patterns in other fields such as in the cases of music and the graphic arts. Computers are used to manipulate much of the music and graphical art that we currently consume. Can the same thing be done with computers and language? This paper will strive to outline our current efforts and some of the challenges that we have faced in teaching computers to identify, manipulate and generate patterns in language, using poetry as the principal backdrop.

From an engineering point of view, modelling the automatic generation of poetry involves two of the fundamental abilities of human beings: creativity and linguistic manipulation. These can be the most mysterious and the most fascinating abilities that humans have, and probably the most difficult ones to model successfully.

These efforts link up with a line of research that has been progressively drawing attention at scientific meetings over the past ten years: that of computational creativity. This research concerns itself with the development and study of programs that behave in a manner susceptible of comparison with similar activities that, when they are carried out by humans, are generally considered creative.

# 3 The Challenges We Identified in Automatic Poem Generation

Having set out to address the task of automatically generating poems, our first challenge was to identify a clear goal to drive the process. To address this, we considered what goals might drive a person to write a poem. Without going into complex details, there are various possible attitudes a person might have when setting pen to paper. Some poets may have a clear aim of producing a poem that can subsequently be considered a 'valuable object'. There are also poets who may sit down with no clear aim in mind, just drawn by a primitive creative impulse. In a lighter vein, some poets may simply sit down hoping to communicate a specific message, undertaking the additional restrictions of doing so in a poetical way. Finally, at the most humble end of the scale, stand the people with a certain craft at putting words together according to metrical rules, but possibly with little talent for obtaining aesthetically pleasing results: poets who write commemorative poems on demand, friends who pen down a stanza to celebrate someone's birthday, lovers that try their hands at sonnets to convey a message as old as the world.

Thinking of these different approaches from an engineer's point of view, two questions come to mind. First, what are the differences between them, and second, which one of these might be easiest to model as a first step in a possible chain of research towards achieving them all.

To answer the first question in depth would need deep investigation into psychological, literary, and linguistic matters. However, as engineers we can postulate an initial simplified approach that - rather than shed light on the possible inner workings of human poetic ability - may allow us to isolate smaller chunks of the problem that might be easier to model than the whole, and yet provide us with initial stepping stones in the path towards understanding the rest of the process. Every kind of creative activity involves a balancing act between a creative impulse - capable of generating new material, but not necessarily very adept at determining its aesthetic qualities - and an aesthetic sensibility - the ability of appraising a given product critically, independently of whether we are capable of producing an equivalent one ourselves. Under this simplification, casual poets tend to have more of the former than the latter, unproductive critics more of the latter than the former, and true poets have a lot of both. A similar analysis might apply to creative endeavours such as music or painting. However, even in our very simplified first shot at modelling poetry, an additional ingredient must be taken into account. Although non-figurative painting is a rage with some people, and music with no identifiable melody can draw praise in certain quarters, poetry without meaning - however imprecise - tends to bring out sceptical frowns or acid comments[2]. We must therefore add to our model some kind of linguistically conveyed message that - to choose the least restrictive interpretation - is somehow associated - 'meant', 'insinuated', 'suggested'... - with a given poem.

This simple model of poetry generation is much too sketchy to be of use in understanding human poetic composition, but it goes a long way towards understanding how one may address automatic composition. Random sentence generators - with or without an underlying grammar - may be considered to have 'creative impulse', but convey no message and have no aesthetic sensibility. Systems capable of validating their output from a metrical point of view show a hint of aesthetic sensibility, but they are clearly missing out on most of the aspects that humans use to evaluate poetry. Where generators draw on corpora of previous texts to guide the generation of new material, they are somehow borrowing the aesthetic judgments encapsulated in the corpus to apply them to the new material, but they are still incapable of actually judging their output. With respect to the linguistic message, our model still leaves an issue

[2] This happens in general terms to human poets, but it is especially true when automatic poets are involved. This suggests there may be differences in the way people judge human and machine poets. This idea must be highlighted, but will not be further pursued in this paper.

unresolved: is it reasonable if we accept that some message may spring to mind as someone reads a poem, or should we require that the poet consciously introduce a particular message in his work? Because this issue is not resolved for human generated poetry, we also will not tackle it at all for computer generated poetry[3].

In the last decade, artificial intelligence has made slow progress into the task of generating texts from conceptual non-linguistic information [RD00]. But current systems tend to generate clumsy stilted texts, and they are very far from the level of literary competence that even the less gifted human poets can achieve. This has lead most attempts at modelling the composition of poetry to steer away from the question of conveying a particular message. For all practical purposes, it is more cost effective to concentrate on the rest of the picture, and let the reader put together some possible interpretation.

With respect to aesthetic sensibility, many systems have identified the fact that humans value much more in a poem than simple syllable counts, and they have shifted away from actual metrical evaluation of their outputs - much as human poets have done over the past two hundred years or so.

Regarding the second question, it seems clear that the easiest approach to model from an engineering point of view is that of a person who has a particular message to convey, and who wants it realised as a poem in some way that other people will recognise with little effort, with little concern for the quality of the poem.

Some of those ideas were already present when the first author of this paper started to build WASP (the Wishful Automatic Spanish Poet) [Gervas00]. Others arose slowly as a result of its implementation or evaluation. WASP started off as an application that composed formal poetry in Spanish in a semiautomatic interactive fashion. Its core was a forward reasoning rule-based system that obtained from the user basic style parameters and an intended message; applied a knowledge-based preprocessor to select the most appropriate metric structure for the user's wishes; and, by intelligent adaptation of selected examples from a corpus of verses, carried out a prose-to-poetry translation of the given message. In the composition process, WASP combined natural language generation and artificial intelligence techniques to apply a set of construction heuristics obtained from formal literature on Spanish poetry. In its own way, WASP had a kind of creative impulse, it worked upon an intended message, and it had a certain internal model of what principles the outcome should obey which might have qualified as a simplified model of aesthetic sensibility. And yet, its results were always judged as poor by the human evaluators (criticism that surely no other human poets have ever felt).

A number of issues came to mind when we first began wondering what important features of human poetry composition might have been lacking in our initial prototypes. What follows in the rest of this paper is not meant as an exhaustive list, but rather as a compilation of the particular set of questions that over the years that have passed since those early attempts, which we have considered worthwhile to explore in further - more precisely focused - research attempts.

One of the most difficult aspects to model is that of aesthetic evaluation of text, even if addressed only at the most simple phonetic levels. How does one identify when a sentence 'sounds good'? There are obvious initial criteria such as syllable counts,

[3] Nonetheless, by stretching this issue to its less stringent extreme, we may be condoning all kinds of automatically generated garbage: the human mind is very good at finding meaning in chaos, and it will painfully parse almost anything into something that resembles sense. In some cases, the harder it has been to artificially create a possible interpretation - or the higher the disparity between all the possible interpretations -, the more valuable a poem can be.

position of stressed syllables, rhyme and alliteration. But most of these, however precise they may sound when read in literary manuals, do not translate well into precise generative rules. It keeps surprising us how often attempts at defining them end up as a series of examples.

The ability of human poets to handle metaphor is a cornerstone of our literary tradition. When is a metaphor suitable? Which domains are good targets for a metaphor? How to choose how much to use from the tenor domain and how much from the vehicle domain?

Poems usually tell a story. Although narrative intelligence appeared in the nineties as a research line of its own within cognitive science and artificial intelligence, few results have appeared, and none that might help to address the fundamental questions of what makes a good plot line for a poem.

Almost every poem we can remember arises particular emotions in us when we read it, or even just remember it. Any serious attempt at modelling the composition of poetry should have a reasonable model of emotions - both those that the poet is portraying in this text and those that the text might evoke in the reader.

# 4 Ongoing Research Efforts at UCM to Address Those Challenges

Having identified the challenges outlined above, the research team set to work on some of them individually.

## 4.1 Understanding Phonetics

Robinson's work [Robinson06] concentrated primarily in the abstraction and identification of phonic elements and patterns in poetry written in Spanish and the graphical representation of these elements to human researchers.

A valid phonemic (or phonetic) transcription is necessary for any algorithm that will be capable of measuring phonic qualities of an utterance in most languages. In Spanish it is true that a phonemic transcription is not as complicated for identifying alliteration or rhyme as it would be for English due to the high correlation between phonemes and graphemes in Spanish; however, in order to evaluate rhyme and rhythm, the role of the phonemic transcription becomes more critical.

The main challenges for implementing an automatic Spanish phonemic transcription are: differences between spoken and written word, loan words, and differences between the numerous Spanish dialects in the world. In Spanish the differences between written and spoken word are general minimal, but as far as the other challenges are concerned, the Colors of Poetry software makes some approximations and follows prescriptive grammatical rules [Llorach03, Hammond01, RAE99] for a starting point. In this case, the starting point is a valid transcription for a very large majority of words in Spanish.

Once the valid transcription is generated, a poem can be evaluated using most of the same traditional criteria that masters of poetics and metrics have used for thousands of years. Robinson's work evaluated the similarities between the automatic, computer generated analyses of poems and professional analyses of poems. With few exceptions, the analyses were convincingly similar in metrics, syllabification, intonation and sometimes even interpretation.

It is this ability to automatically --and competently-- evaluate poems with software that allows software to evaluate automatically whether a line of poetry is more "valid" than another.

## 4.2 Using Phonetic Knowledge to Drive Poem Generation

Having the means to evaluate poems programmatically may constitute as a first approximation at having a machine model of aesthetic sensibility, at least in what concerns the phonetic level. In order to obtain a first prototype corresponding to our two basic elements in poetry composition some kind of model of creative impulse was needed. A simple first approach was attempted by applying the work described in the previous section in a preliminary approximation towards an evolutionary generator of alliterative text [HRG07]. The goal was to take a simple text and a preferred phoneme for the software to generate similar sentences with greater alliteration using synonyms. The evolutionary algorithm (with the aid of a phonemic transcriber, Microsoft Word and Google) tried to produce an alternative sentence while trying to preserve the initial meaning and coherence. A bigram language model and the evaluation of the phonetic analysis are used to assess the fitness of the sentences.

The process of computational evaluation of the sentence quality is a difficulty that we must improve upon. In our algorithm we worked with two variables, alliteration versus coherence. These two variables carried equal weight in our experiments; however, it is doubtful that a large group of people would agree that these two measures carry equal value. Most people prefer a completely mundane sentence that makes sense to line of alliterative drivel. Our future work will strive to find an acceptable balance between these extremes, and then re-apply this balance to the evolutionary algorithm.

From the point of view of the alliteration, there are other options that must also be explored. In this work we have considered that an alliteration is based in the repetition of a single phoneme, but it can also be the repetition of syllables or other textual aspects. Not to mention that we can also include more complex mixtures of alliterative multiple phonemes, vowels and/or consonants.

Once the aforementioned weaknesses have been attended to, we will then apply what we have learned to more patterns in language other than alliteration, such as rhythm and rhyme.

## 4.3 Managing Vocabulary

A useful poetry generation tool for humans and computers is a thesaurus. What many people probably already realize is that Microsoft Word has a decent thesaurus, some people may even realize that it works in many different languages. However, what many people probably do not realize is that Microsoft Word has a decent thesaurus with a programmatical interface that any software application can access. Microsoft has invested considerable time and money into the linguistic capabilities of their products. No matter what stance the readers may have towards the company and its products, we wanted to illustrate the tremendous benefit that one can realize by reusing what is already available, as opposed to reinventing the wheel. Furthermore, the ability to lookup synonyms and antonyms for almost any given word in almost any given language seemed beneficial enough to justify the use of this common commercial product.

Our implementation is an Online Agent that allows various software applications to connect, specify a word and language, and then request a list of synonyms or antonyms. The online agent then forwards the request to the Microsoft Word software installed locally, and returns the result to the online client.

Another similar implementation relies on WordNet [Miller95], a lexical database quite extensively used in the research world, to provide similar functionality, but only in English. Most available linguistic resources are not suitable to use in generation directly due to their lack of mapping between concepts and words. WordNet is an on-line lexical reference system whose design is inspired by current psycholinguistic theories of human lexical memory. The most ambitious feature of WordNet is its attempt to organize lexical information in terms of word meanings, rather than word forms. English nouns, verbs and adjectives are organized into synonyms sets, each of them representing one underlying lexical concept. These synonyms sets - or *synsets* - are linked by semantic relations like synonymy[4] or hyponymy[5].

This type of thesaurus-like modules provide interesting an interesting way that models the human ability to store and manipulate words when putting together messages in text. But it is not until they are connected to a module which uses this ability to generate texts that they can be considered models of human word usage in text composition. This has been attempted in prototypes such as the alliterator described above, or the module for lexical choice in text generation described in PRINCE project [HPGC06a, HPGC06b, PHGC06]. In that work, WordNet is used to enrich an existing generator - for the PRINCE system for building textual realizations of conceptual renderings of fairy tales - with lexical choice, where the decision between lexical alternatives that represent the same content is taken. When a word is needed for a concept during the lexical choice, the system looks for synonyms and hypernyms available for this word in WordNet.

## 4.4 Comparison, Analogy and Metaphor

Metaphor and analogy are two cognitive mechanisms that have been recognized as underlying the reasoning across different domains. Because of this, they play an indomitable role in creativity, thus calling our attention as a potential resource for the PRINCE project [HPGC06a, HPGC06b, PHGC06]. Metaphors (and Analogies) fundamentally result from an interaction between two domains (the vehicle and the tenor, in Metaphor literature). This interaction can be simplified as an isomorphic alignment (or mapping) between the concept graphs that represent the two domains. For the PRINCE project, we are exploring the structure mappings with two particular realization templates in mind: “X is the Y of Z” sentences; “X is as Y as Z” sentences. For example, “Freud is the father of Psychoanalysis” results from the mappings Freud ↔ father applied to the domains Psychoanalysis and family structure, respectively. One

[4] Two expressions are synonymous if the substitution of one for the other never changes the truth value of a sentence in which the substitution is made. Words in different syntactic categories cannot be synonyms because they are not interchangeable, which makes it necessary to partition WordNet into nouns, verbs, adjectives and adverbs.

[5] A concept represented by the synset {x,x,...}is said to be a hyponym of the concept represented by the synset {y,y,...} if a native speaker accepts sentences constructed from such frames as “An x is a (kind of) y”. Since there is normally a single superordinate, it generates a hierarchical semantic structure, in wich a hyponym is said to be below its hypernym or superordinate.

can find this template present in many more examples (e.g. “Brugges is the Venice of Belgium”, “the Lion is the king of the jungle”, “the eyes are the mirror of the soul”, etc.). Our goal is therefore to apply this template (using a structure alignment algorithm) in order to get potentially creative text realizations. Thus, we always need two domain concept maps, one for the context at hand (i.e. partially describing the story that is being generated), another for the vehicle domain (the one from which to draw the analogical perspective). This in itself raises challenges (which domains to use? when? how to select a good mapping?) for which we have some ideas that we will summarise in the discussion section.

For now, we have tested with a single vehicle: the Greek deities domain, extracted from WordNet. It was obtained by isolating the subgraph representing the Greek deity taxonomy, enriched with a simple (algorithmical) extraction of relations from their glosses (to get knowledge such as “Aphrodite is the goddess of beauty”, “Zeus is father of Aphrodite”, “Aeolus is the god of wind”, etc.). Whenever needed, our algorithm is able to map a part of the story under construction to this vehicle domain (thus providing expressions like “The princess was the Aphrodite of Royalty” or referencing king as the “The Zeus of Royalty”).

Although less subtle than Analogy or Metaphor, the Simile is another figure of speech based on cross domain similarity that we believe can be explored computationally in a setup such as that of PRINCE. Again, we are looking for a cross-domain mapping, but now with less worries regarding structure alignment: we can focus on two individual concepts that share the same distinctive property, thus avoiding the look for surrounding consistency. For example, if Adonis is said to be handsome, one can easily map our knight (which is said to be handsome in the Knowledge Base) and generate the sentence “The knight was as handsome as Adonis”. Again, this can only be made possible recurring to a rich knowledge base. These queries look both in the taxonomy as well as in the gloss contents. For example, if we ask for something that is “male” and has the word “handsome” in its gloss, we get the noun “Adonis”. Naturally, this yields very interesting outcomes (e.g. “The princess was as pretty as a Rosebud”, “The king was as stern as a Dutch Uncle”), but often ends in empty sets.

## 4.5 Putting Together an Underlying Plot for a Poem

The problem of how to generate the brief structure that may be required to act as the backbone of an intended poem is related to another open line of research [GDPH05,PG06], that of automatically generating basic stories: short texts that only narrate the main events of the plot. KIIDS (Knowledge-Intensive Interactive Digital Storytelling) is a system that deals with the problems of story representation and generation in many contexts, especially in interactive environments.

Basically, KIIDS generates stories step by step, adding narrative episodes and simulation elements when they are needed to continue the storytelling process. The core of the KIIDS system is the imaginative memory that creates the next narrative episode of a story reusing old episodes found in the case base, taking into account the current state of the narration and using explicit knowledge about narrative and world simulation. ProtoPropp is the name of the application that implements KIIDS for generating folk tales.

The system operates with a representation in Description Logics - a well established formalism used for representing knowledge on the Semantic Web -, combining stored fabulas - a narratological term for the set of main events that happen in the plot - with the narrative knowledge implemented in a domain-specific ontology. The domain of application is the traditional folk tale, using the well-known morphology of Vladimir Propp as narratological background.

### 4.6 Dealing with Emotions

There are other important factors involved in the intuitive way a reader sizes up a story on first approaching it. A valuable proposition in terms of measuring these factors is provided by Pérez y Pérez [PS01]. The Mexica storytelling system considers the tension of the stories that it generates, measured in terms of how the characters in the story suffer changes in their emotional reactions to other characters and their perception of threats to their life or health. A story is considered valuable relative to how often tension rises and falls throughout its duration.

Although this representation of tension is crude, it provides a very good initial approximation to an issue that certainly needs to be addressed by storytelling systems.

Researchers of the NIL group are beginning to address these issues in [FG06a, FG06b] an approach to automated marking up of texts with emotional labels. The approach considers in parallel two possible representations of emotions: as emotional categories and emotional dimensions. For each representation, a corpus of example texts previously annotated by human evaluators is mined for an initial assignment of emotional features to words. This results in a List of Emotional Words (LEW) which becomes a useful resource for later automated mark up. The proposed algorithm for automated mark up of text mirrors closely the steps taken during feature extraction, employing for the actual assignment of emotional features a combination of the LEW resource and WordNet for knowledge-based expansion of words not occurring in either. The algorithm for automated mark up has been tested experimentally against equivalent judgements provided by human evaluators, and it stands up reasonably well by comparison.

## 5. Workable Solutions and Future Expectations

The lines of work outlined above correspond to ongoing research efforts on topics that we believe should be taken into account on any model of poetry generation that aims to provide a reasonably complete picture of the various ingredients that humans put into play when composing poems. However, the topics on this list are considered necessary but not sufficient: there are plenty of other topics that would need to be modelled as well before a more complete picture can be drawn. The sections below outline a number of additional aspects of poetry generation that might fall under this second heading. For each one of them, we outline the problem briefly and - where possible - we suggest current technological solutions that may help to build exploratory prototypes.

### 5.1 Who Are We Writing For: User Modelling

Do human poets always have somebody particular in mind when they write a poem? Do they write for themselves? For posterity? None of these questions has a clear answer. And yet, it is easy to see that having some model of the reader - what he knows, what he feels, what he expects, what he may sympathise with... - can play a role in the process of composition. It seems reasonable to expect that poem generation systems with this ability should perform better than those without it.

User modelling is a technology that has been in use in artificial intelligence and information access for several years. It relies on building explicit or implicit models of the user that the system can apply to guide its decisions in dealing with the user. Researchers at NIL have used this technology in fields as diverse as personalised information access [DG04a] and interactive narrative [PG04]. At present, we are working on incorporating information about user emotions into our existing user models. Once this is achieved, it may be worth integrating emotional models of the intended reader into the process of composing poetry.

## 5.2 Adding Specialized Knowledge: Semantic Web Ontologies

An important obstacle in trying to emulate people's ability to convey complex meanings by means of language is the fact that most of what is conveyed by the sentences used by a person is not actually explicit in those sentences, but rather inferred logically from what he does say. This is very much so when poetry is concerned, where powerful imagery is transmitted not by the words themselves but by their meaning and the associations that it triggers in the reader's mind. This process is very difficult to model in a computer system. In fact, the task of modelling it has been a major research goal for artificial intelligence over the last fifty years or so.

One of the latest technologies proposed to address this problem is the use of ontologies - considered as formulations of conceptual taxonomies in formal languages that a computer might find easy to manipulate - for enriching web pages, in the hope that this will make it possible for machines to 'understand' the content of web pages and thus be able to provide more useful responses -  locate appropriate airline tickets, carry out monetary transactions in an autonomous manner... - to users without human intermediaries. A popular formal language for writing out these formulations is OWL (Ontology Web Language) [BHHHMPS], a logical language in the family of Description Logics [BCMNP03]. The combination of resources like Ontologies and an inference formalism such as Description Logics has proved very useful for generating semantically correct texts. However, the possibilities of applying such combinations to obtain results in practical situations is restricted by the availability of ontological resources for the domains under consideration.

NIL researchers developed in [PGD04] an OWL ontology based on Propp’s Morphology of the Folk Tale oriented towards automatic story generation. This basic ontology holds concepts about three basic domains: interactive goal-directed experiences, narrations and simple simulations.

## 5.3 Modelling Parallel Composition of Several Versions of One Poem: Evolutionary Algorithms

One aspect that machine models of poem writing tend to bypass is the fact that human poets usually build their poems as a succession of drafts that they progressively improve, sometimes having several versions of the draft co-existing as possible candidates, with final decisions made on them only once they either fit in with the rest of the composition or they get pushed out by a different alternative that does.

This kind of approach to poem composition matches reasonably well the general operating procedure of artificial intelligence techniques based on evolutionary algorithms. These techniques rely on building a population of candidates solutions, successively applying a set of mutator and crossover operators to model the way in which such a population might change over time due to evolution, and after each cycle of evolution testing which of the candidate solutions in the population are better suited to pass on to the next cycle.

### 5.4 Integrating the Different Individual Solutions: A Tribe of Agents

The arguments presented so far outline specific techniques that may be of use in modelling particular abilities that play a role in human poetry composition. For each one, the research efforts described attempt to model these abilities, but in most cases they are still far from successful. They correspond to ongoing work and it is still too soon to judge their suitability as models for the equivalent human abilities. However, it is clear that even successful models were obtained for each particular ability, the way in which humans resort to these abilities within the context of the composition of a given poem requires some way for combining these abilities, deciding when to apply each one, and resolving possible conflicts between them.

Conscious of this problem, we are exploring technological solutions that may provide the means for negotiating the interactions between the various abilities in a practical way. Multi-agent solutions are a popular approach within artificial intelligence for implementing systems where several modules must cooperate to solve a problem, in such a way that the final behaviour of the systems is not predefined but emerges from the interaction between the various modules that cooperate, known as agents. We believe that solutions of this kind may be useful in modelling the way the various prototypes we are developing might interact to produce a poem.

An example of this hypothesis at work is given in the PRINCE system for building text realisations of basic stories [HPGC06a, HPGC06b, PHGC06]. PRINCE had a WordNet Agent (for handling those queries to the database), a candidate reference agent (which gives sets of candidate references for a concept to whoever asks for them), a proxy agent - the Facilitator agent that deals with requests/communications between different agents -, and two Analogy related agents. The TextGenerator agent is the one that deals with the NLG generation process, and can be considered as a wrapper for the original PRINCE module. This is the agent that sets off the flow of control information of the whole process. In a first step it initialises the mapper agent with the whole context of the tale that is rendering into text, producing the mapping between the domains involved. After that, the TextGenerator agent follows the usual pipeline control flow of PRINCE, interacting with the RefSet agent when needing the information of the different concepts in the tale. A detailed example of an application of the architecture working on a literary domain is presented in [GDPH05].

## 6. Conclusions

This paper outlines ongoing work at Universidad Complutense de Madrid on poetry generation and related subjects within the field of artificial intelligence. The analysis presented and the topics discussed are not in the least intended as an exhaustive survey of the field, or a prescriptive of what might be done, but rather as a simple description of the way the group of researchers at NIL have chosen to break up the process of poetry generation into subtasks and to address those of them that seemed more computationally tractable. It is presented here in the hope that it may be of use to other researchers in the field. The authors want to thank all past and present members of the NIL research group whose work is cited or described here.

# 7. References

[BCMNP03] Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D. and Patel-Schneider, P. F., The Description Logic Handbook. Theory, Implementation and Applications, Cambridge University Press, UK, 2003.

[BHHHMPS] Bechhofer, S., van Harmelen, F. and Hendler, J., Horrocks, I., and McGuinness, D. L., Patel-Schneider, P. F. and Stein, A., OWL Web Ontology Language Reference, W3C. http://www.w3.org/TR/2004/REC-owl-ref-20040210/ [Last access: 16/1/2006]

[DG04a] Díaz, A. & Gervás, P., 2004. “Dynamic user modeling in a system for personalization of web contents”. Current Topics in Artificial Intelligence, CAEPIA-TTIA 2003, LNAI 3040, pp. 281-290.

[FG06a] Francisco, V., Gervás, P.: "Exploring the Compositionality of Emotions in Text: Word Emotions, Sentence Emotions and Automated Tagging", AAAI-06 Workshop on Computational Aesthetics: Artificial Intelligence Approaches to Beauty and Happiness, 2006.

[FG06b] Francisco, V., Gervás, P.: "Automated Mark Up of Affective Information in English Texts" TSD'06 (International Conference on Text, Speech and Dialogue), LNAI, Springer, 2006.

[Gervas00] Gervás, P.: “An Expert System for the Composition of Formal Spanish Poetry”. Journal of Knowledge-Based Systems, Volume 14, Issue 3-4, June 2001, Elsevier Science, pp 181-188

[GDPH05] Gervás, P., Díaz-Agudo, B., Peinado, F., and Hervás, R.: ‘Story plot generation based on CBR’, Journal of Knowledge-Based Systems, 18(4-5), 235–242, (2005).

[Hammond01] Hammond, Robert M. The sounds of Spanish: Analysis and Application (with Special Reference to American English). Somerville: Cascadilla Press, 2001.

[HPGC06a] Hervás, R., Pereira, F.C., Gervás, P., Cardoso, A.: "Cross-Domain Analogy in Automated Text Generation". ECAI-06 Workshop on Computational Creativity, August 2006.

[HPGC06b] Hervás, R., Pereira, F.C., Gervás, P., Cardoso, A.: "A Text Generation System that Uses Simple Rhetorical Figures" Procesamiento de Lenguaje Natural, nº 37, 2006, pp.199-206.

[HRG07] Hervás, R., Robinson, J., Gervás, P.: Evolutionary Assistance in Alliteration and Allelic Drivel. In M. Giacobini et al. (Eds.): EvoWorkshops 2007, LNCS 4448, pp. 537–546, 2007.

[Llorach03] Llorach, E. Alarcos. Gramática de la lengua española. Editorial Espasa Calpe, S.A., 2003.

[Miller95] Miller, G. A. 1995. WordNet: a lexical database for English. Commun. ACM 38(11):39–41.

[PG04] Peinado, F., Gervás, P.: "Transferring Game Mastering Laws to Interactive Digital Storytelling". In Göbel, S., et al (Eds.): Technologies for Interactive Digital Storytelling and Entertainment (TIDSE'04). 24-26 June, Darmstadt, Germany. LNCS, 3105, 48-54, Springer, 2004.

[PG06] Peinado, F., Gervás, P.: "Evaluation of Automatic Generation of Basic Stories". New Generation Computing 24, 3: Special issue: Computational Creativity, 289-302. ISSN: 0288-3635. 2006.

[PGD04] Peinado, F., Gervás, P., Díaz-Agudo, B.: "A Description Logic Ontology for Fairy Tale Generation". In Veale T., et al (Eds.): 4th International Conference on Language Resources and Evaluation, Proc. of the Workshop on Language Resources for Linguistic Creativity, LREC'04, 56-61. 29th May, Lisbon, Portugal. LREA, 2004.

[PHGC06] Pereira, F.C., Hervás, R., Gervás, P., Cardoso, A.: "A Multiagent Text Generator with Simple Rhetorical Habilities". AAAI-06 Workshop on Computational Aesthetics: AI Approaches to Beauty and Happiness, July 2006.

[PS01] Pérez y Pérez, R. and Sharples, M., "MEXICA: a Computer Model of a Cognitive Account of Creative Writing", Journal of Experimental and Theoretical Artificial Intelligence 13, 2, 2001.

[RAE99] Real Academia Española. Ortografía de la lengua española. 1999. Oct. 30, 2006.

[RD00] Reiter, E., Dale, R.: Building Natural Language Generation Systems. Cambridge University Press (2000)

[Robinson06] Robinson, Jason. Colors of Poetry: Computational Deconstruction. Georgia: University of Georgia, 2006.

[Veale95] Veale, T. 1995. Metaphor, Memory and Meaning: Symbolic and Connectionist Issues in Metaphor Interpretation. PhD Thesis, Dublin City University.