Un modelo computacional para la generación automática de poesía formal en castellano

Pablo Gervás Gómez-Navarro
Departamento de Inteligencia Artificial,
Universidad Europea de Madrid,
Villaviciosa de Odón, Madrid 28670

Resumen Se plantea la definición de un esquema computacional básico para modelar el proceso de generación de poesía formal en castellano. A partir de una discusión de las limitaciones de sistemas anteriores, se propone un esquema que solventa los problemas principales manteniendo las especificaciones elementales.

1 *Introducción.*

El trabajo existente en generación automática de texto está orientado sobre todo a la obtención de textos cuyo significado se ajuste en la mayor medida posible a un mensaje predeterminado que se toma como entrada (boletines meteorológicos [3], informes técnicos [6]). La labor de generación radica en encontrar estructuras lingüisticas correctas que representen el mensaje a transmitir. Se ha trabajado también en la generación aleatoria de oraciones [9] con vistas a evaluar gramáticas, parsers, o sistemas de PLN [2, 12, 8]. En todos estos casos se han ido identificando fuentes de problemas y las investigaciones realizadas han dado lugar a importantes mejoras en los mecanismos de generación. En pocos casos se ha estudiado el problema de la generación desde el punto de vista de una evaluación estética de la forma de los textos generados. Sin embargo está claro que en multitud de ocasiones el lenguaje que las personas utilizan, generan, o reciben está diseñado prestando una atención especial a la forma. En algunos casos (titulares de periódicos, anuncios publicitarios, propaganda política...) se presta incluso más atención a la forma que al mensaje, o al menos se tratan los dos ingredientes con la misma importancia. Recientemente ha surgido interés sobre estos temas [11, 1, 7]. El caso extremo de este conjunto de fenómenos son los textos poéticos, y dentro de ese bloque general, aquellos textos poéticos que se atienen a formas estróficas y a estructuras métricas tradicionales. El presente artículo plantea un esquema computacional que permite formalizar los procesos de generación de texto en aquellos casos en que se prima la forma del mensaje sobre su contenido, prestando especial atención a aquellos aspectos formales que están reglamentados por la métrica tradicional castellana [13]. No se pretende conseguir un generador automático de poemas que compita con poetas reales, ni reducir el proceso de creación literaria a un conjunto de reglas algorítmicas. Se trata de estudiar aquella parte de los procesos involucrados que sí que es susceptible de cierta mecanización, con vistas a aclarar sus posibles aplicaciones como método auxiliar para humanizar el aspecto general de los textos generados automáticamente.

2 *Aspectos susceptibles de formalización*

En esta sección se plantean los aspectos de la métrica tradicional que pueden formalizarse de manera algorítmica. Se puede encontrar una presentación más detallada en [13].

Por un lado el número de sílabas y la colocación de acentos imponen restricciones fuertes en el tipo de versos que se pueden generar, por cuanto que sólo se admiten como versos secuencias de palabras que cumplan las reglas.

Por otro lado las reglas de la forma estrófica a utilizar imponen restricciones a la estructura general del poema.

2.1 Colocación de acentos

Un verso formal requiere una distribución particular de las sílabas acentuadas en función del número de sílabas que tiene. Por ejemplo, un verso endecasílabo tiene once sílabas y debe tener acentos en posiciones concretas que se correspondan con alguno de los esquemas siguientes [13]: 1, 6 y 10; 2, 6 y 10; 3, 6 y 10; 4, 6 y 10; o 4, 8 y 10. En los ejemplos siguientes, las sílabas acentuadas situadas en posiciones clave aparecen subrayadas:

2, 6, 10

en verdes hojas ví que se tornaban

3, 6, 10

nunca fue corazón; si preguntado

4, 6, 10

De tan hermoso fuego consumido

4, 8, 10

soy lo demás, en lo demás soy mudo

2.2 Número de sílabas métricas

La sílaba métrica no coincide exactamente con la sílaba morfológica. Cuando una palabra termina en vocal y la siguiente palabra empieza en vocal, la última sílaba de la primera palabra y la primera sílaba de la siguiente palabra forman una única sílaba métrica. Este fenómeno se conoce como *sinalefa* e influye en el problema de establecer la cuenta de sílabas de un verso.

Por ejemplo, el verso siguiente:

bástete amor lo que ha por mí pasado
13 sílabas

se convierte en:

báste**te - a**mor lo **que - ha** por mí pasado
11 sílabas

porque tiene dos casos de sinalefa (marcados en negrita).

2.3 Formas estróficas

Un poema puede ser una secuencia de versos sin estructurar y con muy pocas restricciones, pero este artículo trata específicamente la generación de poemas que se ciñen a formas estróficas concretas.

Las restricciones impuestas por la forma estrófica escogida pueden afectar a distintos parámetros.

Pueden determinar la longitud de los versos: un romance tiene versos de ocho sílabas y un soneto tiene versos de once sílabas (endecasílabos).

Pueden determinar la rima de algunos versos sin afectar a la longitud del poema, como en el siguiente romance (se muestra la rima en negrita):

Romance
Por el Val de las Estacas
pasó el Cid a mediod**ía**.
En su caballo Babieca
muy gruesa lanza tra**ía**.
Va buscando al moro Abdala
que enojado le ten**ía**.
...

Pueden afectar a la rima de todos los versos e imponer una cierta restricción a la longitud del poema: un poema en tercetos encadenados tiene estructuras fijas de rima y el número de versos ha de ser múltiplo de tres.

Tercetos encadenados
Alma a quien todo un dios prisión ha s**ido**,
venas que humor a tanto fuego han d*ado*,
medulas que han gloriosamente ard**ido**
su cuerpo dejará, no su cuid*ado*;
serán ceniza, mas tendrán sent**ido**;
polvo serán, mas polvo enamor*ado*.

Pueden afectar a la rima de todos los versos y fijar la longitud del poema: un cuarteto tiene estructura fija de rimas y exclusivamente cuatro versos.

Cuarteto
Muérome por llamar Juanilla a Ju**ana**,
que son de tierno amor afectos v*ivos*,
y la cruel, con ojos fugit*ivos*,
hace papel de yegua galici**ana**.

El poeta que genera un poema no necesariamente tiene presente estas restricciones en forma de reglas, sino que ha desarrollado una sensibilidad (de cuya formalización concreta en la mayoría de los casos el poeta mismo no es consciente) que le permite distinguir cuando un borrador o una idea son poéticamente aceptables. Se puede definir una aproximación formal de esta sensibilidad en términos de: el número y naturaleza de las palabras que constituyen un verso, y las palabras concretas a utilizar. Este es el camino seguido en el presente artículo.

3 Consideraciones generales sobre la generación de poemas

La generación de textos poéticos formales se diferencia de la generación elemental en una serie de aspectos importantes desde el punto de vista computacional. Suponemos que el poeta tiene elegido un tipo de estrofa concreto al que espera ceñirse[1], y una noción no

[1]Dejamos abierta para futuros trabajos la posibilidad de tener en cuenta la elección de estrofa como un grado más de libertad a disposición del poeta para adecuar el texto generado al mensaje deseado.

excesivamente precisa del mensaje que espera transmitir con su poema.

1. El poema normalmente tiene una longitud fija. El mensaje tiene que acortarse o alargarse para ajustarse a la longitud del poema.
2. La unidad elemental de generación no es la oración sino el verso. Aparte de la planificación que lleva a repartir el mensaje en una serie de oraciones, hará falta una planificación adicional que reparta esas oraciones en un número fijo de versos. Estos dos procesos de planificación tienen lugar normalmente en paralelo, y, en muchos casos (estructuras estróficas particularmente restrictivas, finales de verso restringidos por la rima, versos finales de un poema) llega a darse prioridad a la planificación en términos de versos.
3. Para estrofas más complejas, compuestas de distintas formas estróficas combinadas, (como por ejemplo un soneto compuesto de dos cuartetos y dos tercetos) puede ser necesario un proceso adicional de planificación de la distribución del mensaje sobre las distintas subestrofas.
4. Dada una planificación concreta del mensaje a lo largo de los versos del poema, aquellas palabras que ocupen posiciones al final de los versos pueden tener restricciones adicionales fruto de la estructura de rimas del poema. Si estas restricciones no se tienen en cuenta al generar los primeros versos (proceso que fija las rimas de todo el poema en algunas estrofas) puede resultar muy difícil encontrar opciones válidas para los últimos versos.
5. Tanto a la hora de planificar la distribución del mensaje a lo largo de los versos como a la hora de elegir vocabulario se han de tener en cuenta las restricciones impuestas por la rima (ha de haber suficientes palabras que rimen entre sí, que sean aplicables al mensaje en alguna formulación posible, y que en esa formulación las palabras que riman aparezcan al final de los versos en las posiciones necesarias del poema).
6. En gran parte de los casos, a la hora de ajustar un poema para que su forma cuadre perfectamente con la forma estrófica elegida, es necesario sacrificar el mensaje en aras de la estética. Esto puede suponer eliminar partes del mensaje, añadir fragmentos al mensaje que no figuraban en la intención inicial, o directamente modificar radicalmente el mensaje.

De lo expuesto se seleccionan las siguientes ideas como puntos de partida para explorar el proceso de generación:

- la representación del mensaje elegido debe dejar muchísima libertad al proceso de búsqueda de la forma adecuada
- la planificación más importante a la hora de dar con una forma estéticamente correcta es la distribución en versos, prestando especial atención a la posibilidad de usar palabras que rimen al final del verso
- a la hora de elegir el vocabulario hay que tener en cuenta la disponibilidad de rimas

4 Trabajos previos

El sistema WASP [4] es un sistema basado en reglas con razonamiento hacia delante. Toma como entrada un conjunto de palabras y un conjunto de patrones de versos y devuelve un conjunto de versos.

El sistema trabajaba con un algoritmo *generate and test* donde el proceso de generación tiene dos grados de libertad: selección de patrón a utilizar para el verso siguiente, y selección de palabra a utilizar para sustituir la siguiente categoría del patrón de trabajo. Estos dos procesos de selección se llevan a cabo de manera aleatoria.

Dos elementos clave que afectan al proceso de generación (la selección de patrones y palabras disponibles) se dejan en manos del usuario.

Los primeros experimentos realizados con el sistema WASP se centran en determinar la validez de distintas estrategias de selección de palabra a la hora de conseguir versos que cumplan las normas métricas. Se toma como caso de estudio el verso endecasílabo por ser el que mayores restricciones presenta. Un segundo conjunto de experimentos estudia el efecto de la selección de patrones y de vocabulario proporcionados al sistema en la calidad estética de los resultados finales, tomando como medida la valoración subjetiva de un grupo de evaluadores. Los resultados de estos dos conjuntos de experimentos se discuten en [4].

La mayor aportación obtenida del análisis de los experimentos de WASP concierne a las limitaciones encontradas en el modelo computacional utilizado:

- el único mecanismo que WASP permite para especificar mínimamente el mensaje (la selección de vocabulario inicial) es insuficiente
- el mecanismo que WASP aporta para asegurar que cada verso es una posible continuación del verso previo funciona para enlazar versos en que queda incompleta una oración con versos adecuados para continuarla, pero fracasa cuando el final del verso coincide con el final de una oración
- la selección aleatoria de palabras no permite aplicar ninguna heurística, por simple que sea, para gobernar el mensaje transmitido por el resultado final
- se hace necesario alguna manera de restringir la selección de palabras a utilizar incluso en los casos en que el mensaje se sacrifica para cuadrar la forma
- en la mayoría de los casos, el sistema es incapaz de generar un poema por falta de datos adecuados (palabras que rimen, patrones que permitan continuar el borrador alcanzado hasta ese punto, patrones que permitan utilizar al final la rima requerida,...)

5 Modelo computacional ampliado: ASPID

ASPID (Automatic Spanish Poet Initial Development) es una arquitectura compleja para enfrentarse a la labor de la generación automática de poesía en castellano. La tarea se ha dividido en tres bloques:

- recogida de datos: han de obtenerse una estructura elegida para el poema, un esquema del mensaje, un vocabulario total, y un corpus de versos etiquetados de referencia
- planificación: dividida en tres etapas (a) distribución del esquema del mensaje entre los versos del poema, (b) validación del conjunto de datos de entrada, (c) selección de vocabulario y versos etiquetados de referencia para cada verso del poema
- realización: se asignan palabras concretas a las categorías de los patrones seleccionados

5.1 Recogida de datos

La lista de *estructuras para el poema* ha de restringirse a aquellas para las que el sistema disponga de información (reglas que definen la forma estrófica deseada).

El *esquema del mensaje* del que se parte es simplemente una lista de palabras que pueden o no constituir oraciones concretas. El sistema de generación se ocupa de darles forma lingüística específica. La única información que utiliza el sistema a parte de esa primera selección de palabras prioritarias es el orden en el que aparecen, que determinará su posición relativa en el poema. Las palabras que se utilicen en el esquema del mensaje han de estar presentes en el vocabulario total. Un ejemplo sería:

```
(esquema tan hermoso viento nunca mudo)
```

El *corpus de versos etiquetados de referencia* cumple en ASPID funciones equivalentes a las que cumplía en WASP el conjunto de patrones disponibles. Cada verso etiquetado (VER) consta de un verso específico y del patrón correspondiente a ese verso. El sistema de generación utiliza la información adicional proporcionada por el verso adjunto al patrón para guiar la selección de palabras.

```
((sample (n 72)
         (patt  NCMS PDEL NCMS
                NCMS DET PP03FS NCFS)
         (words desde_n del cielo
                error de la ventura)
         (beg 0) (end 1) (level nil))
```

Contiene además información acerca de si el verso es válido para empezar (`beg`) o terminar un poema (`term`).

El *vocabulario total* es un conjunto de palabras. Para cada palabra se ha de disponer de la información necesaria para aplicar las reglas métricas (longitud en sílabas, posición del acento, y si empieza o termina en vocal) y su categoría sintáctica.

```
(word (cual desde_n)
      (numsil 2)
      (acento 2)
```

```
(emp 0)
(term 0)
(cat NCMS)
(rima en)
(level nil))
```

El campo `level` se utiliza para anotar las prioridades de utilización en versos concretos que se establezcan para cada palabra.

El corpus de versos etiquetados de referencia y el vocabulario total se generaron automáticamente utilizando conjuntamente el analizador en Prolog presentado en [5] y el POS tagger [15].

Los resultados presentados en este artículo se obtuvieron con versos etiquetados y vocabulario extraido de una selección de tercetos de poetas del Siglo de Oro español.

5.2 Planificación

La *distribución del esquema del mensaje* entre los versos del poema se lleva a cabo con criterios puramente numéricos: se reparte el total de palabras del esquema entre los versos del poema, manteniendo únicamente un paralelismo entre el orden de aparición en el esquema y el número de verso al que son asignadas.

Para el ejemplo mencionado, suponiendo que se esté intentando elaborar un terceto, se obtendría:

```
(parteesquema 1 tan hermoso)
(parteesquema 2 viento nunca)
(parteesquema 3 mudo)
```

Cada uno de estos fragmentos corresponderá a la parte de esquema que gobierna cada uno de los versos que se han de componer.

Un módulo del sistema se ocupa de calcular las similitudes entre los fragmentos que se han obtenido del esquema y los patrones disponibles en los VER. Se establece la similitud en función de las categorías sintácticas de las palabras del fragmento de esquema y las que permite el patrón del VER.

Otro módulo de preprocesamiento de los datos se ocupa de contar el número de palabras disponibles para cada rima en el vocabulario. Esta información se declara en memoria para poder ser utilizada por el sistema en momentos sucesivos.

El paso principal de planificación consiste en seleccionar patrones y rimas concretos para cada verso. En el sistema ASPID esto tiene lugar de dos maneras distintas. Al principio del proceso se lleva a cabo una planificación inicial teniendo en cuenta toda la información disponible, escogiendose unos patrones y una rima concretos. En caso de que esa planificación no conduzca a una solución, el sistema admite que la elección de patrón sea revisada más adelante.

En primera instancia se trata de comprobar que:

- existen suficientes palabras que rimen para realizar la estructura estrófica elegida para el poema
- existen suficientes VERs cuyos patrones permitan versos con las rimas encontradas
- existe al menos una secuencia de VERs tales que sus patrones puedan hilvanarse en un todo, observando las condiciones de frontera necesarias (el primer verso empieza una oración; el último verso termina una oración; si un verso termina oración, el siguiente debe empezarla) y de modo que las palabras del esquema del mensaje asignadas a cada verso del poema puedan asignarse en el orden adecuado a los patrones que acompañan a los VERs correspondientes

Si no se cumplen estos criterios el sistema devuelve un mensaje de error informando al usuario del problema.

A partir de ese momento el sistema trabaja secuencialmente sobre cada verso, empezando por el primero. Antes de pasar a la etapa de realización, del siguiente verso, un módulo especial establece prioridades entre todas las palabras disponibles, de modo que sólo se utilicen para ese verso aquellas que resulten adecuadas en función del esquema presentado.

Esta primera selección ha de regirse por los criterios generales siguientes. Para seleccionar palabras para el vocabulario se han de escoger las palabras en el siguiente orden de prioridad:

- las palabras que aparecían en el fragmento de esquema correspondiente al verso
- las palabras que riman

Este proceso de selección ha de llevarse a cabo para cada uno de los versos del poema antes de pasar a la etapa de generalización. Una vez dejada atrás la etapa inicial de planificación previa, el sistema trabaja de hecho oscilando entre la planificación de cada verso consecutivo , su realización tentativa, y el paso al verso siguiente.

5.3 Realización

El proceso de asignación de palabras concretas a las categorías de los patrones elegidos para la secuencia inicial concluye el proceso de generación del poema final. Esta asignación ha de hacerse imponiendo a cada paso las restricciones descritas (número de sílabas, posiciones de acentos, rima).

Las unidades elementales que guían la generación son el *patrón de verso* (`posfrase`) y el *borrador de verso* (`patt`):

```
(posfrase ?n ?a ?otramas $?b)
(posgen ?n $?y)
```

donde `?n` identifica el verso sobre el que se trabaja, y las demás variables permiten ir eligiendo para cada categoría del patrón (que se van eliminado) un palabra que añadir al borrador. Al final del proceso, el patrón está vacío y el borrador lleno.

La generación se lleva a cabo secuencialmente desde el principio del patrón, eliminando progresivamente la categoría sintáctica encontrada en primera posición y añadiendo al borrador alguna palabra del vocabulario que pertenezca a esa categoría. A la hora de elegir palabra, el sistema impone condiciones que aseguren la validez formal del borrador de verso resultante según las reglas de la métrica. Estas reglas están implementadas en forma de funciones imperativas que calculan valores booleanos. En caso de que no se encuentre ninguna extensión válida (y de que el verso no esté completo) el sistema admite un grado parcial de vuelta atrás, deshaciendo los cambios recientes a la búsqueda de una solución mejor. Esto requiere un proceso adicional de anotación para evitar que se repitan las opciones ya probadas.

Si una vez comprobadas todas las opciones no se ha encontrado un verso válido, se elige un patrón alternativo para el que el parecido con el fragmento de esquema y las condiciones de frontera sean equivalentes.

6 *Resultados y conclusiones*

El sistema ASPID se ha probado en una primera versión para generar tercetos simples (con el primer verso rimando con el tercero). Para ello se le proporcionaron como entrada datos que representaban una colección de 24 tercetos extraidos de sonetos clásicos de la literatura, con la única restricción de que cada terceto constituyera una unidad de significado por sí sola. Esto proporcionaba 72 patrones posibles y un vocabulario básico relacionado con los patrones elegidos de 228 palabras. Este vocabulario se amplió con un conjunto de palabras alternativas que ampliasen las posibilidades de generación.

El sistema ejecutó tres tandas de 33 intentos, variando en cada tanda el esquema proporcionado como base, tanto en contenido como en longitud.

Se observa una tendencia generalizada a repetir esquemas de oración que recuerdan a los poetas clásicos. Dado que el sistema no utiliza de momento información semántica ni gramática, es comprensible que los resultados no ofrezcan significados claros.

A continuación se presenta un ejemplo de poema generado por el sistema:

```
marchitara_ la nieve el fin pesado
por tal caso con una lengua sola
duro rato de rastro ensangrentado
```

Se puede observar que el sistema no utiliza símbolos de puntuación de ningún tipo, no respeta las mayusculas a principio de linea, y representa las tildes mediante guiones bajos. Todas estas características están abiertas a refinamientos posteriores, una vez que el algoritmo de generación consiga resultados satisfactorios.

Como se ha comentado, el sentido del terceto presentado deja mucho que desear. Sin embargo, las restricciones formales en cuanto a métrica y cadencia prosódica son correctas. Una vez satisfechos de que el método seguido impone de manera adecuada las restricciones de forma necesarias, queda abierto el paso siguiente de añadir conocimientos al sistema para que el significado de los poemas generados mejore.

Una primera aproximación que mejoraría la apariencia de los resultados sin necesidad de recurrir a conocimiento lingüistico complejo sería establecer restricciones más fuerte en cuanto a las fronteras entre verso y verso, de modo que se establezca una continuidad sintáctica más clara. Este paso es fácilmente alcanzable desde el punto de vista de la implementación de las restricciones. Sin embargo al endurecer las restricciones sobre los patrones concretos a utilizar sería necesario ampliar el número de patrones disponibles. De no ser así el sistema respondería con mensajes de error en un número alto de las ejecuciones.

El modelo computacional da respuesta a los problemas que presentaba el sistema WASP. La solución propuesta requiera una cantidad importante de conocimiento (vocabulario, corpus, definición de formas estróficas). Sólo parte de la tarea de generación de este conocimiento en los formatos adecuados ha podido automatizarse mediante la utilización de herramientas informáticas adecuadas. Hasta el momento, la cantidad de conocimiento disponible para probar el sistema es pequeña, lo cual hace dificil precisar hasta qué punto el método es capaz de presentar resultados aceptables. Es de esperar que estos problemas se resuelvan pronto.

Otra fuente de incertidumbre a la hora de evaluar el funcionamiento del método reside en la dificultad de a la hora de definir un método riguroso y repetible de evaluación. Para el sistema WASP se presentó una selección de los poemas generados (aquellos mínimamente largos y mínimamente interpretables) a un conjunto de evaluadores. La valoración mediante métodos de este tipo es siempre altamente subjetiva por cuanto que varía mucho de un individuo a otro. Por otro lado, las valoraciones de un mismo individuo pueden presentar variaciones entre el primer poema leido (el único para el que se recibe una evaluación limpia) y los siguientes, que siempre se van comparando inconscientemente con los ya vistos. En futuros trabajos es necesario estudiar cuidadosamente este tema y encontrar soluciones más adecuadas.

Referencias

[1] Bailey, P., 'A reader-based model of story generation', *Symposium on Artificial Intelligence and Creative Language: Stories and Humour* (7th-8th April, 1999),*AISB'99 Convention*, 6th-9th April 1999, Edinburgh College of Art & Division of Informatics, University of Edinburgh.

[2] Balkan, L., Arnold, D., y Fouvry, F., 'Test suites for evaluation in natural language engineering' en: *Proceedings of the Second Language Engineering Convention*, pp 203-210, Londres, octubre 1995.

[3] Chevreau, K., Coch, J., García Moya, J.A., y Alonso, M., 'Generación multilíngüe de de boletines meteorológicos', *Procesamiento del Lenguaje Natural*, SEPLN, N. 25, septiembre 1999, pp. 51-58.

[4] Gervás, P., 'WASP: Evaluation of Different Strategies for the Automatic Generation of Spanish Verse', AISB-00 Symposium on Creative & Cultural Aspects and Applications AI & Cognitive Science, University of Birmingham, England, Monday 17th April 2000 & Tuesday 18th April 2000.

[5] Gervás, P., 'A Logic Programming Application for the Analysis of Spanish Verse', First International Conference on Computational Logic, Logic Programming Implementations and Applications stream, Imperial College, London, UK, 24th to 28th July, 2000.

[6] Horacek, Helmut y Busemann, Stephan, 'Towards a Methodology for Developing Application-Oriented Report Generation' en: Günter, Andreas and Herzog, Otthein (eds.), , *22nd German Conference on Artificial Intelligence (KI-98). Proceedings*,Bremen, Germany, 1998.

[7] Hisar Maruli Manurung, Graeme Ritchie y Henry Thompson 'Towards a computational model of poetry generation', presentado en: AISB-00 Symposium on Creative & Cultural Aspects and Applications AI & Cognitive Science, University of Birmingham, England, Monday 17th April 2000 & Tuesday 18th April 2000.

[8] Mellish, C.S. 'Some chart-based techniques for parsing ill-formed input' en: *27th Annual Meeting of the Association for Computational Linguistic, Proceedings of the Conference*, pp 102-109, Vancouver, British Columbia, Canada, junio 1989.

[9] M.-J. Nederhof, 'Efficient generation of random sentences', *Encyclopedia of Computer Science and Technology*, Vol. 41, Marcel Dekker, 1999, pp. 45–65.

[10] Personal web page of Jose Luis Pérez de la Cruz Molina, http://www.lcc.uma.es/personal/cruz/sonetos/pral.html

[11] Perez y Perez, R. y Sharples, M., 'MEXICA: A computational model of the process of creative writing', *Symposium on Artificial Intelligence and Creative Language: Stories and Humour* (7th-8th

April, 1999), *AISB'99 Convention*, 6th-9th April 1999, Edinburgh College of Art & Division of Informatics, University of Edinburgh.

[12] Purdom, P., 'A sentence generator for testing partners', *BIT*, 12: 366-375, 1972.

[13] Quilis, A., 'Métrica española', Ariel, Barcelona, 1985

[14] Real Academia Española (Comisión de Gramática), 'Esbozo de una nueva gramática de la lengua española', Espasa-Calpe, Madrid 1986.

[15] Sánchez león, F., Corpus Resources And Terminology ExtRaction project (MLAP-93/20), 'Spanish tagset for the CRATER project', ftp://ftp.lllf.uam.es/pub/CRATER/esT-tagger-1-0.tar.gz

[16] van Noord, G. y Neumann, G. 'Syntactic generation', chapter 4, en: Ronald A. Cole, Joseph Mariani, Hans Uszkoreit, Annie Zaenen and Victor Zu (eds.), *Survey of the State of the Art in Human Language Technology*, 1998.

[17] Wiliamsen, V.G. and Abraham, J.T., Association for Hispanic Classical Theater web page, ftp://listserv.ccit.arizona.edu/pub/listserv/comedia/poetic1.html