

# Descripción de Entidades y Generación de Expresiones de Referencia en la Generación Automática de Discurso\*

## *Entity Description and Referring Expression Generation in Automatic Generation of Discourse*

Raquel Hervás      Pablo Gervás  
Universidad Complutense de Madrid  
Instituto de Tecnologías del Conocimiento  
28040 Madrid, Spain  
raquelhb@fdi.ucm.es, pgervas@sip.ucm.es

**Resumen:** La descripción de entidades es una tarea indispensable en la generación automática de texto para cualquier tipo de discurso. Todo texto necesita referirse a elementos del mundo con distintos fines, y la base de estas referencias serán las descripciones que se han hecho de dichos elementos anteriormente. En este proceso se hayan involucradas dos de las tareas básicas de la Generación de Lenguaje Natural (GLN): la Determinación de Contenido y la Generación de Expresiones de Referencia. En este trabajo se estudia la relación entre ambas al describir entidades y referirse a ellas en cualquier tipo de discurso generado automáticamente. Para ello será necesario revisar la arquitectura clásica de pipeline, estudiando las interacciones que pueden ser necesarias entre los distintos módulos involucrados.

**Palabras clave:** Descripción de entidades, Generación de lenguaje natural, Generación de expresiones de referencia, Arquitecturas GLN

**Abstract:** Entity description is an essential task in automatic generation of text for any kind of discourse. Every text needs to refer to domain elements with different goals, and the baseline of these references would be the descriptions of these elements presented before. Two of the main tasks of Natural Language Generation (NLG) are involved in this process: Content Determination and Referring Expression Generation. In this work the relation between them is studied from the point of view of entity descriptions and references in any type of automatically generated discourse. It would be necessary to revise the classic pipeline architecture, exploring the interactions between the different modules involved.

**Keywords:** Entity description, Natural language generation, Referring Expression Generation, NLG architectures

### 1. *Introducción*

La descripción de entidades es una tarea indispensable en la generación automática de texto para cualquier tipo de discurso. Desde los sistemas de diálogo hasta los textos narrativos todo texto generado necesita referirse a elementos del mundo con distintos fines, y la base de estas referencias serán las descripciones que se hayan hecho de dichos elementos

anteriormente. En ocasiones las entidades referidas necesitarán ser distinguidas de otras del mismo tipo o con características comunes, y para ello sus características deberán ser expuestas de manera que puedan ser útiles para tal fin. Otras veces la información con que son descritas ciertas entidades será importante para una comprensión completa del discurso.

La importancia de estas descripciones es mayor cuando el tipo de discurso hace que la única información disponible para el lector u oyente sea la dada en el propio discurso. En el contexto de un diálogo entre dos personas,

\* Esta investigación está financiada por el Ministerio de Educación y Ciencia (TIN2006-14433-C02-01) y la Universidad Complutense de Madrid y la Dirección General de Universidades e Investigación de la Comunidad Autónoma de Madrid (CCG07-UCM/TIC-2803).

el oyente dispone tanto de la información del entorno que les rodea como de las ideas intercambiadas entre ambos. Por ello, una referencia como “la mesa” podría ser entendida sin necesidad de mayores explicaciones. Sin embargo, en un discurso donde la comunicación es exclusivamente lingüística y no hay ningún contexto visual, la única información de la que dispondrá el lector u oyente es la que se ha mostrado en el texto, y por ello las descripciones de las entidades del discurso serán cruciales para que el texto transmita la información deseada.

En la generación y posterior uso de estas descripciones de entidades se hayan involucradas dos de las tareas básicas de la Generación de Lenguaje Natural (GLN): la Determinación de Contenido (DC) y la Generación de Expresiones de Referencia (GER). En la mayoría de los sistemas GLN la relación entre estas dos tareas es unidireccional.

La arquitectura más extendida para sistemas de GLN es la arquitectura secuencial o de pipeline, donde la elección del contenido del discurso y su organización en mensajes es realizada al principio del proceso. Es esta información seleccionada la que utilizan los subsiguientes pasos de la generación, entre ellos la GER, sin posibilidad de que sea revisada por etapas anteriores del pipeline. Desde el punto de vista de la relación de la DC y la GER en la descripción de entidades, el flujo de información proporcionado por esta arquitectura puede no resultar adecuado en ciertas ocasiones. Imaginemos que en la fase de GER el sistema se da cuenta de que no dispone de suficiente información sobre una entidad para distinguirla del resto que están en el mismo contexto. Dada una arquitectura secuencial el problema no tendría remedio, y la expresión de referencia generada sería ambigua.

Un problema como éste puede solucionarse de varias maneras. Una de ellas sería que la DC se encargara, además de decidir qué información es importante para el discurso como un todo, de comprobar si la información seleccionada es suficiente para que no se produzcan ambigüedades a la hora de generar las expresiones de referencia. Otra solución sería que la DC seleccionara poca o nada de información para las descripciones, y que fuera la GER la que se encargara de solicitar la inclusión de nueva información cuando resultara necesaria para generar las referencias.

En este trabajo se estudia la relación entre la Determinación de Contenido y la Generación de Expresiones de Referencia al describir entidades y referirse a ellas en discursos generados automáticamente donde la única información disponible para el lector u oyente es la contenida en el propio texto. Para ello será necesario revisar la arquitectura clásica de pipeline, estudiando las interacciones que pueden ser necesarias entre los distintos módulos involucrados.

Un dominio que resulta muy adecuado para el estudio de esta relación es el de la generación automática de texto para cuentos. Los cuentos contienen por definición muchos elementos a los que habrá que referirse, desde personajes principales o secundarios, hasta lugares u objetos implicados en la acción. Y en ellos tienen una función muy importante tanto la descripción de entidades, que se encargará de presentar todos elementos al lector, como la Determinación de Contenido en general, ya que son ricos en información relativa a los distintos elementos y esta información deberá ser filtrada adecuadamente a la hora de plasmar la historia en texto. Por ejemplo, en todo cuento nos encontraremos con mucha información relativa a los personajes (edad, color de pelo, hermosura, bondad,...) y a sus relaciones con los demás elementos de la historia (quién es su padre o madre, dónde viven, a quién aman u odian,...). Es importante que toda esta información sea organizada y filtrada adecuadamente para que el texto final resulte legible.

## 2. *Revisión del Trabajo Previo*

La generación de lenguaje natural (GLN) se subdivide en varias tareas concretas (Reiter y Dale, 2000), y cada una de ellas opera a un nivel distinto de representación lingüística (discurso, semántica, léxico, sintaxis...). La GLN se puede aplicar en dominios donde los objetivos de comunicación y las características de los textos a generar son muy distintos, desde la transcripción a lenguaje natural de contenidos numéricos (Goldberg, Driedger, y Kittredge, 1994), a la generación de textos literarios (Callaway y Lester, 2001).

### 2.1. **Arquitecturas de Sistemas GLN**

Hay muchas formas de organizar un sistema de Generación de Lenguaje Natural (GLN), y las ventajas y desventajas de ca-

da una de ellas son todavía campo de discusión (DeSmedt, Horacek, y Zock, 1995; Reiter, 1994). Teniendo en cuenta la división en módulos, podemos encontrar desde una arquitectura integrada (Kantrowitz y Bates, 1992), donde el sistema está formado por un único módulo, hasta una arquitectura con módulos separados para cada una de las tareas de la GLN (Cahill et al., 2001). Considerando el flujo de control, por un lado podemos encontrar una arquitectura de pipeline (Reiter, 1994), donde los módulos son completamente independientes, y por otro una arquitectura de pizarra (Calder et al., 1999), donde los módulos colocan información en un espacio de almacenamiento común sin preocuparse de qué otros módulos van a utilizarla.

La arquitectura más generalizada es el pipeline de Reiter y Dale (2000), que ha servido de inspiración para muchos otros (Figura 1). La arquitectura secuencial o de pipeline tiene la ventaja de la simplicidad. Cada módulo está encapsulado al máximo porque recibe la entrada de un origen y envía la salida a un destino. Se supone que la representación intermedia en cada etapa es una representación completa de lo que se sabe en ese punto sobre la declaración que se va a generar. Sin embargo, su principal desventaja es que las decisiones tomadas en una fase deben mantenerse a lo largo de toda la cadena, sin posibilidad de revisión o mejora. Por tanto, se trata de un modelo muy poco flexible.

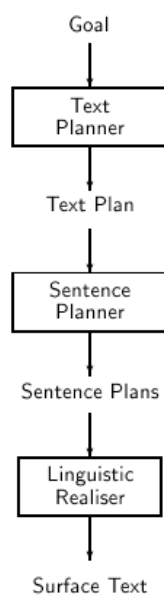


Figura 1: Esquema básico de una arquitectura secuencial

Sistemas como HYLITE+ (Bontcheva y Wilks, 2001) han explorado alternativas a la rigidez de comunicación entre módulos proporcionada por el pipeline desde el punto de vista de la Determinación de Contenido y la Realización Superficial. HYLITE+ es un sistema de generación de hipertexto dinámico que produce explicaciones de tipo enciclopédico en un determinado dominio. En este sistema el planificador de contenido produce inicialmente un plan de texto para un hipertexto conciso que sólo contiene hechos sobre el concepto a explicar. Durante la realización superficial, una vez que el formato final ha sido decidido, se escoge la alternativa adaptable más apropiada. En ocasiones esto lleva a la inclusión de un nuevo objetivo comunicativo que resulta en expandir el texto básico inicial utilizando información extra.

STREAK (Robin y McKeown, 1996) es un sistema para la generación automática de resúmenes de partidos de baloncesto utilizando una arquitectura de revisión. En una primera fase se construye un borrador que contiene sólo la información esencial para el resumen, y en una segunda pasada se revisa incrementalmente este borrador para añadir tanta información de carácter secundario como sea necesaria para mejorar la legibilidad y aprovechar al máximo el espacio dado. Este modelo requiere un nuevo tipo de conocimiento lingüístico que son las reglas de revisión, que especifican las distintas formas en las que un borrador puede ser modificado para añadir nueva información de forma concisa.

## 2.2. Generación de Expresiones de Referencia

El uso adecuado de expresiones de referencia para competir con los textos generados por autores humanos entraña cierta dificultad. Según Reiter and Dale (Reiter y Dale, 2000), una expresión de referencia debe comunicar suficiente información para identificar unívocamente un referente dentro del contexto del discurso, pero siempre evitando modificadores innecesarios o redundantes.

Sin embargo, hay ocasiones en las que la información usada en la referencia puede ser desconocida para el oyente o innecesaria para distinguir a la entidad. El uso de este tipo de referencias podría ser considerado como una violación de la Máxima Conversacional de Cantidad de Grice *“Do not make your contribution more informative than is requi-*

*red*” (Grice, 1975). Esto sería cierto si el objetivo fuera exclusivamente proporcionar suficiente información al oyente para identificar la entidad a la que su interlocutor se está refiriendo. No obstante, si estamos considerando como objetivo no sólo la identificación del referente, sino también alertar al oyente sobre algunas de sus propiedades, entonces la Máxima de Cantidad se cumple.

### 2.3. Descripción de Entidades

Según Milosavljevic (2003), una descripción de una entidad es la realización lingüística de un conjunto de una o más proposiciones que tienen como propósito hacer que el oyente se construya un modelo mental de la entidad descrita. En su trabajo de tesis (Milosavljevic, 1999) exploró el uso de comparaciones en la descripción de entidades, postulando que en el proceso de producir una descripción resulta útil el uso de comparaciones con entidades similares y familiares. Otros trabajos han estudiado la generación de descripciones de entidades de diversos tipos.

En (Lavoie, Rambow, y Reiter, 1997) se presenta el sistema MODEx, que genera descripciones en lenguaje natural de modelos software orientados a objetos. MODEx permite al usuario personalizar los planes de texto en ejecución, de manera que cada texto refleja las preferencias individuales del usuario en cuanto al contenido y/o la presentación de la salida generada.

En (Ardissono y Goy, 2000) se presenta SETA, un sistema de generación automática de catálogos web personalizados. Para ello emplea técnicas de GLN basada en plantillas, que permite la generación dinámica de categorías de productos y sus elementos. En estas descripciones el sistema mezcla diferentes tipos de información sobre las características y propiedades de los productos presentados.

El sistema de generación de texto Methodius (Isard, 2007) crea descripciones personalizadas de objetos de museo que pueden ser presentadas de diversas maneras, desde texto o voz en un aparato en mano hasta diálogo con un guía de museo robótico.

### 3. Generación de Descripciones y su Relación con el Resto del Proceso GLN

Dos tareas dentro de la Generación de Lenguaje Natural están involucradas en el proceso de generación de descripciones para

entidades en un discurso y su posterior relación con el resto del texto: la Determinación de Contenido y la Generación de Expresiones de Referencia. La relación entre ambas a la hora de referirse a las entidades en un discurso es importante para transmitir la información deseada y evitar ambigüedades.

La Determinación de Contenido (DC) se encarga de seleccionar la información que se pretende comunicar en la descripción de la entidad. Dependiendo del lugar que ocupa dicha descripción dentro del discurso, y de la importancia de la entidad en el mismo, la información a expresar en la descripción podrá ser más o menos extensa. Según el tipo de discurso habrá entidades que sobresalen sobre el resto o son más importantes que las demás. En un cuento, por ejemplo, los personajes protagonistas o el lugar donde se desarrolla la acción serán presentados con descripciones mucho más detalladas que los personajes secundarios o los objetos menos importantes. La DC deberá determinar qué elementos del discurso sobresalen sobre los demás y deberá seleccionar información acorde a este hecho a la hora de describirlos.

La Generación de Expresiones de Referencia (GER) se encarga de generar referencias a entidades que permitan distinguirlas del resto de elementos del discurso, al mismo tiempo que proporciona cualquier información extra que el lector pueda encontrar útil más adelante. Para ello depende no sólo de la información proporcionada por la DC, sino también del contexto del texto en el momento de realizar la referencia. Este contexto está determinado, entre otras cosas, por cómo ha sido descrita previamente cada entidad, es decir, por la información que se puede considerar conocida por el lector.

La mayoría de las referencias que aparecen en cualquier tipo de discurso se usan básicamente para referirse a entidades del contexto que deben ser distinguidas de otras que tienen alrededor. Este tipo de referencias se pueden considerar *referencias distintivas*. Un ejemplo de este tipo de referencias sería “Put the box on *the table next to the door*”, donde el hablante señala que la caja debe ponerse en la mesa que está al lado de la puerta y no en ninguna otra mesa. En estos casos las propiedades mencionadas son conocidas por el oyente y necesarias para distinguir una entidad determinada del resto del contexto.

En la línea de las implicaciones conversa-

cionales expuestas por Grice, si la referencia contiene información que no es necesaria para la identificación del referente, entonces el oyente podría asumir que esa información es importante para algo más, y dependiendo del contexto hará diferentes inferencias a partir de esta información. Este tipo de referencias se consideran *referencias informativas*.

En el caso de las *referencias distintivas*, la GER debe tener en cuenta a la hora de generar la referencia no la información completa de que dispone, sino la información que ha sido transmitida previamente en el texto. En el caso de las *referencias informativas*, es posible que la información extra que se pretende proporcionar ya haya sido expuesta anteriormente. Por ejemplo, en el cuento de La Cenicienta, casi todas las menciones a los zapatos de Cenicienta incluyen la información extra de que éstos son de cristal. En estos casos la reiteración de dicha información pretende destacar este hecho al lector. Por otro lado, si la información a presentar no ha sido dada previamente, es tarea de la GER encargarse de que sea entendida correctamente y no produzca ningún tipo de confusión. Esto podría ocurrir, por ejemplo, si no se relaciona correctamente el concepto referido anteriormente con la nueva característica. Si por ejemplo tenemos una historia en la que ha aparecido una princesa de la que sólo se ha dicho que es guapa, y más tarde se usa una referencia como “la princesa morena”, el lector podría pensar que la princesa morena es una nueva princesa y no tiene nada que ver con la anterior.

### 3.1. Generación de Descripciones Guiada por la Determinación de Contenido

La Determinación de Contenido podría ser la encargada de decidir qué información comunicar en la descripción de entidades teniendo en cuenta todos los elementos que aparecerán en el texto y sus características. En el momento de decidir qué información incluir en la descripción de una entidad la DC debe ser consciente de todos elementos que aparecerán en el discurso e incluir la información necesaria para que todos ellos sean distinguibles.

Una consecuencia de esta solución es que la DC tendrá que prever parte de las decisiones que tomará más adelante la GER. Para ello deberá asegurarse de añadir al discurs-

so suficiente información para que algoritmos como la selección de atributos, que se encarga de elegir qué atributos de un elemento deben ser mencionados para distinguirlo del resto de elementos del mismo tipo, no se encuentren con menos información de la que necesitan para hacer las referencias.

### 3.2. Generación de Descripciones Guiada por la Generación de Expresiones de Referencia

Otra posible solución para el problema planteado sería proporcionar una comunicación bidireccional entre los módulos de Determinación de Contenido y Generación de Expresiones de Referencia. Ante la falta de información para crear una referencia adecuada, como en el ejemplo de las princesas mencionado anteriormente, el módulo de GER haría una solicitud al módulo de DC para obtener más información sobre las dos princesas involucradas. Supongamos que se sabe que la segunda princesa es morena, aunque inicialmente este hecho no parecía relevante para el discurso y no se había incluido en su descripción. Con esta nueva información sí será posible generar referencias que distingan a las dos princesas.

Por supuesto, podría darse la situación en que el módulo de DC no pudiese proporcionar información adicional para la descripción de una entidad. En estos casos, la GER deberá recurrir a otro tipo de técnicas, como por ejemplo el uso de demostrativos (*this princess* o *that princess*) o de otros identificadores (*the first princess*). Pero estos aspectos de la GER quedan fuera del estudio realizado en este artículo.

## 4. Propuesta de Implementación en una Arquitectura GLN

En esta sección se estudian las implicaciones de las dos propuestas expuestas anteriormente desde el punto de vista de una posible implementación de las mismas dentro de un sistema de generación previamente existente. TAP (Text Arranging Pipeline) (Gervás, 2007) es una arquitectura software para la generación automática de texto. TAP está formado por una serie de interfaces que definen una funcionalidad genérica para un pipeline de tareas orientadas a la generación de lenguaje natural, desde una entrada conceptual inicial hasta la realización superficial final en forma de texto, con etapas intermedias de

planificación de contenido y planificación de oraciones. A partir de la entrada conceptual que debe procesarse, los diversos módulos de TAP trabajan sobre las representaciones intermedias utilizadas para almacenar resultados parciales de manera que progresivamente se filtra, agrupa y enriquece la entrada hasta conseguir estructuras cada vez más cercanas al lenguaje natural tanto en estructura como en contenido.

La filosofía seguida en el diseño de TAP es identificar la naturaleza estructural genérica de la información que requiere de procesamiento en las diferentes fases de la generación de lenguaje natural, y proporcionar interfaces para acceder a esta estructura. La arquitectura de TAP ha sido específicamente diseñada para permitir el desarrollo de un conjunto reutilizable de componentes software capaces de resolver tareas básicas de GLN, trabajando con información codificada en formatos estándar, y proporcionando textos adecuados a diferentes tareas en diferentes dominios. Con respecto a las tareas que se discuten aquí, son relevantes el módulo de determinación de contenido y el módulo de generación de expresiones de referencia.

#### 4.1. Determinación de Contenido

La implementación actual del módulo de determinación de contenido de TAP recibe como entrada un conjunto de referentes, un conjunto de propiedades, un conjunto de relaciones y un conjunto de eventos. Las propiedades pueden aplicarse tanto a referentes (en cuyo caso suelen acabar representadas como adjetivos) como a eventos (en cuyo caso acaban representadas como adverbios). Las relaciones pueden establecerse indistintamente entre cualesquiera de los cuatro elementos posibles (referentes, propiedades, relaciones o eventos), de manera que permiten tanto la representación de modificadores de frases nominales, complementos de oraciones, o de relaciones de causalidad.

Dentro del módulo de determinación de contenido, la tarea básica que se lleva a cabo es agrupar los eventos en secuencias de eventos contiguos en el tiempo y que ocurren en el mismo lugar. Estos bloques se consideran como una primera aproximación al concepto de escena, y se utilizan para establecer una planificación inicial del discurso. Las escenas se ordenan relativamente según el orden cronológico de los primeros eventos de

cada una. Esta primera aproximación del discurso a generar debe enriquecerse con información adicional relativa a las descripciones de los referentes que participan en cada evento. La información que se introduzca en este punto aparecerá en el texto final mencionada explícitamente como parte del discurso en oraciones independientes (“**La princesa es rubia**”).

#### 4.2. Generación de Expresiones de Referencia

La versión actual (Gervás, Hervás, y León, 2008) del módulo de generación de expresiones de referencia aplica una implementación sencilla del algoritmo de Reiter y Dale (Reiter y Dale, 1992), que permite seleccionar el conjunto de atributos (propiedades y relaciones) para un referente que lo identifican con respecto a posibles menciones previas del mismo en el discurso precedente. Esta tarea se apoya en el conjunto de propiedades y relaciones que se hayan establecido como válidas para ese referente durante la determinación de contenido. La información que se seleccione en este punto aparecerá en el texto final mencionada como expresiones de referencia que participen en oraciones que describen eventos (“**La princesa rubia** salió del castillo”).

El algoritmo de Reiter y Dale opera a base de generar un *conjunto de contraste* que contiene todos los referentes que podrían confundirse con el referente que se pretende mencionar, e ir progresivamente eliminando elementos de ese conjunto cada vez que se añade a la expresión a generar una propiedad del referente a mencionar que los distingue. La generación de la expresión tiene éxito si el conjunto de contraste queda vacío al final del proceso. En caso de que el conjunto de contraste no quede vacío, se ha encontrado un caso en el que la información proporcionada durante la generación de contenido no es suficiente para distinguir unos referentes de otros.

#### 4.3. Adaptaciones Necesarias

De cara a implementar las soluciones descritas en abstracto más arriba, resulta importante dotar al módulo de generación de expresiones de referencia de la capacidad de identificar este tipo de situaciones anómalas, y, en cada caso, almacenar los datos que permitan identificar los referentes entre los que surge el conflicto. Esto es tan sencillo como,

en cada caso en que el conjunto de contraste no esté vacío al final del proceso, almacenar ese remanente del conjunto de contraste junto con el referente original: estos referentes son los que quedan indistinguibles con la información proporcionada por el módulo de determinación de contenido.

Con respecto al módulo de determinación de contenido, sería necesario añadir una funcionalidad que permitiera solicitar una revisión de una asignación de contenido previa, de modo que esta solicitud pueda venir acompañada de un conjunto de conjuntos de referentes indistinguibles según la asignación previa.

#### 4.4. Opciones Básicas de Combinación

Todo lo expuesto anteriormente daría lugar a tres opciones de utilización.

En la primera, la determinación de contenidos no añadiría ningún tipo de información sobre los referentes en la primera pasada. Durante la generación de expresiones de referencia se detectan los conjuntos de referentes indistinguibles, generando una lista de los mismos para los que no ha podido generar una referencia correcta por falta de información. Se solicitaría entonces una segunda pasada de determinación de contenido para añadir información que los distinga. Este bucle podría continuar hasta que el módulo de generación de expresiones de referencia no encuentre problemas a la hora de generar las referencias, en cuyo caso el flujo del pipeline continuaría hasta la generación del texto final.

En la segunda opción, la determinación de contenido puede añadir información descriptiva en la primera pasada, pero no necesariamente tiene que ocuparse de que esa información sea distintiva, sino que serán más bien descripciones iniciales apropiadas para cada entidad según su papel en el discurso completo. Por ejemplo, los personajes y lugares principales siempre se describen más exhaustivamente que los secundarios. La generación de expresiones de referencia se lleva a cabo como en la primera opción y se solicita al módulo de determinación de contenido una segunda pasada para completar las descripciones con la información necesaria para distinguir. En esta segunda opción, pueden combinarse las funcionalidades de las descripciones para facilitar la identificación y para añadir infor-

mación que no distingue de otros referentes pero se considera importante.

La tercera opción incluiría en el módulo de determinación de contenido llamadas a los fragmentos de la generación de expresiones de referencia que son independientes del discurso. Para cada referente se haría una exploración genérica del conjunto de contraste máximo posible aplicando los criterios mencionados anteriormente, y se utilizaría la realimentación obtenida para añadir la información que pudiera ser necesaria ya en la primera pasada de la determinación del contenido.

#### 5. Conclusiones y trabajo futuro

La generación de descripciones guiada por la Determinación de Contenido puede resultar un proceso muy costoso dependiendo de la forma en que sea abordado. A la hora de seleccionar la información correspondiente para la descripción de una entidad, el resto de entidades con que se compara pueden haber sido ya “descritas”, es decir, la información que se usará para su descripción puede haber sido seleccionada anteriormente. Habrá por tanto que decidir si la DC considera esta información seleccionada a la hora de realizar la descripción de las otras entidades o no. No es lo mismo tener en cuenta toda la información disponible sobre una entidad a la hora de distinguirla de otra, que tener sólo en cuenta la información que va a ser mencionada sobre esa entidad en el discurso. En el segundo caso la descripción estará teniendo en cuenta el contexto del discurso y no sólo la información disponible inicialmente.

La generación de descripciones guiada por la Generación de Expresiones de Referencia provocaría que en las descripciones del texto sólo se incluyera la información necesaria para la distinción de los elementos del discurso. Esto podría provocar textos demasiado escuetos, con pocas descripciones, si el número de entidades que pueden ser confundidas con otras es bajo. Además, esta aproximación no se encarga del uso de información adicional no necesaria para distinguir pero que se considera interesante para la comprensión completa del texto. Una posible solución sería que en la DC se decidiera qué información no necesaria se quiere expresar en el texto, y que la fase posterior de REG se encargue de solicitar información adicional a ésta cuando lo necesite.

**Bibliografía**

- Ardissono, L. y A. Goy. 2000. Dynamic generation of adaptive web catalogs. En *Proc. Conference on Adaptive Hypermedia and Adaptive Web-based Systems*, Italy.
- Bontcheva, K. y Y. Wilks. 2001. Dealing with dependencies between content planning and surface realisation in a pipeline generation architecture. En *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI'01)*.
- Cahill, L., R. Evans, C. Mellish, D. Paiva, M. Reape, y D. Scott. 2001. The RAGS reference manual. Informe Técnico ITRI-01-07, Information Technology Research Institute, University of Brighton.
- Calder, J., R. Evans, C. Mellish, y M. Reape. 1999. Free choice and templates: how to get both at the same time. En *May I speak freely? Between templates and free choice in natural language generation*, páginas 19–24, Saarbrücken.
- Callaway, C. y J. Lester. 2001. Narrative prose generation. En *Proceedings of the 17th IJCAI*, páginas 1241–1248, Seattle.
- DeSmedt, K., H. Horacek, y M. Zock. 1995. Architectures for natural language generation: Problems and perspectives. En G. Ardoni y M. Zock, editores, *Trends in natural language generation: an artificial intelligence perspective*, LNAI 1036. Springer Verlag, páginas 17–46.
- Gervás, P., R. Hervás, y C. León. 2008. Nil-ucm: Most-frequent-value-first attribute selection and best-scoring-choice realization. En *Referring Expression Generation Challenge 2008, Proc. of the 5th International Natural Language Generation Conference (INLG'08)*.
- Gervás, P. 2007. TAP: a text arranging pipeline. Informe técnico, Natural Interaction based on Language Group, Universidad Complutense de Madrid, Spain.
- Goldberg, E., N. Driedger, y R.I. Kittredge. 1994. Using natural-language processing to produce weather forecasts. *IEEE Expert: Intelligent Systems and Their Applications*, 9(2):45–53.
- Grice, H.P. 1975. Logic and conversation. *Syntax and Semantics*, 3:43–58.
- Isard, A. 2007. Choosing the best comparison under the circumstances. En *Proceedings of the International Workshop on Personalization Enhanced Access to Cultural Heritage (PATCH'07)*, Corfu, Greece.
- Kantrowitz, M. y J. Bates. 1992. Integrated natural language generation systems. En R. Dale E. Hovy D. Rösner, y O. Stock, editores, *Aspects of Automated Natural Language Generation*. Springer Verlag, Berlin, páginas 13–28.
- Lavoie, B., O. Rambow, y E. Reiter. 1997. Customizable descriptions of object-oriented models. En *Proceedings of ANLP'97*.
- Milosavljevic, M. 1999. *The Automatic Generation of Comparisons in Descriptions of Entities*. Ph.D. tesis, Department of Computing, Macquarie University, Sydney, Australia.
- Milosavljevic, M. 2003. Defining comparison. En P. Slezak, editor, *Proceedings of the Joint International Conference on Cognitive Science with the Australasian Society for Cognitive Science*, University of New South Wales.
- Reiter, E. 1994. Has a consensus NL generation architecture appeared, and is it psychologically plausible? páginas 163–170.
- Reiter, E. y R. Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press.
- Reiter, Ehud y Robert Dale. 1992. A fast algorithm for the generation of referring expressions. En *Proceedings of the 14th conference on Computational linguistics*, páginas 232–238, Morristown, NJ, USA. Association for Computational Linguistics.
- Robin, J. y K. McKeown. 1996. Empirically designing and evaluating a new revision-based model for summary generation. *Artif. Intell.*, 85(1-2):135–179.