Interacción con entornos inteligentes mediante diálogos basados en contexto

Germán Montoro
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Crtra. De Colmenar Km. 15
28049 Madrid - Spain
+34 91 497 22 10
German.Montoro@uam.es

Xavier Alamán
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Crtra. De Colmenar Km. 15
28049 Madrid - Spain
+34 91 497 22 50
Xavier.Alaman@uam.es

Pablo A. Haya
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Crtra. De Colmenar Km. 15
28049 Madrid - Spain
+34 91 497 22 67
Pablo.Haya@uam.es

ABSTRACT
En este artículo presentamos los mecanismos de interacción de una interfaz de diálogos hablada con un entorno inteligente. Esta interfaz se crea de forma automática y la interacción con el entorno varía dependiendo del contexto. La interfaz se basa en una estructura de diálogos en árbol. Diversos módulos procesan esta estructura y la información contextual extraída del entorno, de modo que producen los diálogos específicos para cada uno. Posteriormente, la interacción con los usuarios varía dependiendo del contexto, haciendo que los procesos de interpretación y generación se adapten a la situación actual del entorno. Esta interfaz de diálogos basada en contexto está siendo implementada y evaluada en un entorno inteligente real desarrollado en un laboratorio de la Escuela Politécnica Superior de la Universidad Autónoma de Madrid.

Palabras clave
Interfaz en lenguaje natural, entornos inteligentes, contexto, computación ubicua.

1. INTRODUCCIÓN
Los entornos inteligentes han aparecido como un nuevo campo de investigación en la comunidad científica de las interfaces de usuario. Estos entornos interactúan con los usuarios de forma natural, ayudándoles en sus tareas cotidianas. Los ordenadores y dispositivos computacionales se ocultan a los usuarios, y éstos obtienen los servicios del sistema por medio de interfaces en lenguaje natural sensibles al contexto. Esto hace posible que sea más sencillo y natural interaccionar y gestionar habitaciones, oficinas y entornos en general, que éstos puedan tomar la iniciativa y mejorar la calidad de vida de sus ocupantes.

La aparición de tales entornos hace necesario construir nuevas interfaces que permitan interactuar con los usuarios de forma natural. Las interfaces tienen que ser ahora capaces de llevar a cabo conversaciones relativas al entorno, sus elementos y los servicios que pueden ofrecer a sus usuarios. Las interfaces de diálogo habladas se tienen que adaptar a estos sistemas, de modo que puedan hacer frente a los nuevos retos que ofrecen los entornos inteligentes.

Interacción 2004, 3-7 mayo, 2004, Lleida (España).

En este artículo presentamos una interfaz de diálogos hablada para entornos inteligentes que se adapta a cada dominio del entorno. Los diálogos se crean de forma automática y permiten interaccionar con los servicios del entorno y controlar sus dispositivos mediante una interacción oral en lenguaje natural. El artículo se centra en los procesos de interpretación y generación de los diálogos y explica brevemente cómo el sistema construye la interfaz de forma automática para cualquier entorno dado, información completa sobre este asunto se puede encontrar en [10].

El artículo se organiza de la siguiente forma: la siguiente sección presenta una descripción de nuestro entorno inteligente, a continuación, describimos la ontología empleada para modelizarlo, seguidamente explicamos las áreas relacionadas con los módulos de interpretación, generación e interacción hombre-máquina, en el siguiente apartado introducimos la evaluación del sistema y, finalmente, presentamos las conclusiones y el trabajo futuro.

2. DESCRIPCCIÓN DEL ENTORNO INTELIGENTE
Los entornos inteligentes se basan en el concepto de computación ubicua, definido originalmente por [19].

Los sistemas de computación ubicua proporcionan acceso a servicios computacionales pero ocultan los dispositivos a los usuarios. Los usuarios no tienen que encontrar las interfaces, sino que el sistema tiene la responsabilidad de servirles [1].

Siguiendo con las ideas de computación ubicua, un entorno inteligente es un espacio interactivo, embebido, que trae la computación al mundo físico real. Este entorno permite a los ordenadores participar en actividades que nunca habían implicado computación y a la gente interaccionar con sistemas computacionales de la misma forma que lo harían con otras personas [4].

Dentro de un entorno inteligente se pueden encontrar tecnologías muy heterogéneas; desde dispositivos hardware como sensores, interruptores, electrodomésticos, webcams, etc. hasta aplicaciones software tales como reconocedores de voz, servidores multimedia, agentes de correo electrónico, etc. Estas entidades se tienen que integrar y controlar utilizando la misma interfaz de usuario. Por

ejemplo, un usuario tiene que poder iniciar un servicio de música en broadcast tan fácilmente como apagar las luces. Teniendo en cuenta estas condiciones hemos desarrollado un entorno inteligente real. Este incluye una ontología, que proporciona un mecanismo sencillo para representar el entorno y comunicar su estado, y una interfaz de diálogos automática, que interactúa y controlas los elementos de un entorno real. Esta interfaz se crea y gestiona de forma automática, basándose en la información extraída de la ontología. Después de su creación el sistema puede interactuar con el usuario, empleando los módulos de interpretación y generación.

Este entorno consiste en un laboratorio decorado como una sala de estar donde se han colocado diversos dispositivos. Existen dos tipos de dispositivos: de control y multimedia. Los dispositivos de control son controladores de las luces, un mecanismo de apertura de la puerta de entrada, de detector de presencia, tarjetas inteligentes, etc. Los dispositivos multimedia, como altavoces, micrófonos, un televisor y una cámara IP, son accesibles a través de un backbone IP. Los dispositivos de control están conectados a una red EIB (EIBA) y una pasarela une ambas redes. La aplicación que accede a la capa física se armoniza mediante una capa SMNP (Simple Management Network Protocol) [8].

3. ONTOLOGÍA DEL ENTORNO

La ontología del entorno se describe mediante un documento XML. Al inicio, el sistema lee este documento y construye automáticamente:

- Una pizarra [7], que funciona como una capa de interacción entre el mundo físico y la interfaz de diálogos hablada.
- Una interfaz de diálogos que, por medio de la pizarra, funciona como una capa de interacción entre los usuarios y el entorno.

Esta pizarra almacena una representación de las características del entorno. Estas incluyen su distribución (los edificios y habitaciones), las entidades activas del entorno, su situación, su estado, las posibles relaciones entre ellas y los flujos de información. La naturaleza de una entidad puede variar desde un dispositivo físico hasta un concepto abstracto, tal como el número de personas en una habitación o la lista de personas que tienen permiso para acceder a ella.

La pizarra se utiliza como un servidor de contexto. Las aplicaciones e interfaces pueden preguntar a la pizarra para obtener información sobre el estado de una entidad, o para cambiarlo. Las entidades se pueden añadir o eliminar de la pizarra en tiempo de ejecución y la información se puede utilizar de forma instantánea por el resto de aplicaciones. Las aplicaciones y las interfaces no interaccionan directamente con el mundo físico o entre ellas, sino que sólo tienen acceso a la capa de la pizarra (ver figura 1).

La capa de la pizarra aísla a las aplicaciones del mundo físico. Los detalles de las entidades del mundo físico se ocultan a los clientes [13], haciendo más fácil y estándar desarrollar módulos e interfaces conscientes del contexto.

Las entidades se asocian a un tipo de entidad. Todas las entidades del mismo tipo heredan algunas propiedades generales. Esto significa que si definimos una nueva entidad sus propiedades vendrán automáticamente asociadas a ella. Como resultado,

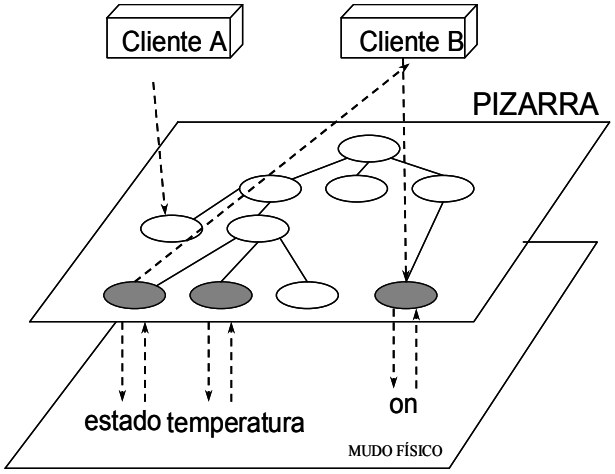

Figura 1. Interacción con la pizarra

construir la ontología sólo requiere definir qué entidades están presentes en el entorno y de qué tipo son.

Algunas de las propiedades asociadas al tipo de entidad representan información lingüística. Esta información está formada por una parte verbal (las acciones que se pueden realizar con la entidad), una parte objeto (el nombre que se le puede dar), una parte modificadora (el tipo de entidad objeto), una parte de situación (dónde se encuentra en el entorno) y una parte de objeto indirecto (a quién se dirige la acción). Un conjunto de estas partes establece una forma posible que puede utilizar un usuario para interaccionar con la entidad. Una entidad tiene asociada una colección de conjuntos de partes, que corresponden con todas las formas posibles de interactuar con la entidad. Cada una de las partes puede estar compuesta de una o más palabras, permitiendo el uso de sinónimos. Adicionalmente, las entidades heredan el nombre de la plantilla de la gramática que tienen asociadas y el método de acción al que se ha de llamar después de que la información lingüística de la entidad se ha procesado completamente. Los métodos de acción son específicos para cada tipo de entidad y ejecutan todas las acciones posibles que pueden ser requeridas por los usuarios (por ejemplo, en una entidad de tipo *luz_regulable*, encenderla, apagarla, subir o bajar la luz).

La información lingüística se transforma en gramáticas específicas y nodos de diálogo que posibilitan el proceso de interacción hablada. Los usuarios gestionan e interaccionan con el entorno mediante la interfaz de diálogos hablada y la interfaz emplea la información representada en la pizarra para proporcionar las capacidades de diálogo.

4. REPRESENTACIÓN DEL DIÁLOGO

Como se ha dicho anteriormente la interfaz de diálogos se compone de un conjunto de gramáticas y una estructura de diálogos.

Las gramáticas posibilitan el proceso de reconocimiento especificando las posibles oraciones que los usuarios pueden pronunciar, limitando de esta forma el número de entradas esperadas por el reconocedor [6]. Así, sólo se permite a los usuarios establecer diálogos relativos a la configuración actual del entorno, no considerando otras posibles interacciones. El sistema

crea una gramática para cada tipo de entidad. Las gramáticas se basan en la plantilla asociada a su tipo. En el proceso de creación de la interfaz las entidades sólo tienen que rellenar su plantilla de gramática correspondiente con su colección de conjuntos de partes lingüísticas.

La estructura de diálogos se basa en un árbol lingüístico. Antes de crear la interfaz de diálogos el árbol sólo tiene un nodo raíz vacío. Cada conjunto de partes lingüísticas se transforma en una ruta del árbol, con un nodo para cada parte. Los nodos cuelgan de nodos padre que representan partes anteriores del mismo conjunto. Los nodos almacenan la palabra correspondiente a esa parte y el nombre de la entidad a la que pertenecen. Las partes con más de una palabra (sinónimos) se transforman en nodos diferentes y las siguientes partes del mismo conjunto cuelgan de cada uno de los nodos sinónimos. Las palabras se analizan mediante un analizador morfológico [3] para obtener su número y género. Las palabras repetidas se analizan sólo la primera vez y esta información se almacena para utilizarla posteriormente en el proceso de generación de diálogos. A modo de ejemplo podemos suponer que la entidad *luz_1* tiene el siguiente conjunto de partes: {“encender dar”, “luz”, “”, “techo arriba”, “”}, donde la primera columna corresponde a la parte verbal, la segunda a la parte objeto, la tercera columna a la parte modificadora, la cuarta a la parte de situación y la última columna a la parte de objeto indirecto. “Encender” y “dar” son sinónimos, lo mismo que “techo” y “arriba”. Así, este conjunto de partes corresponde con cuatro posibles formas que se pueden emplear para interactuar con la entidad *luz_1*. Empezando de un árbol vacío, el sistema crearía el árbol lingüístico mostrado en la figura 2.

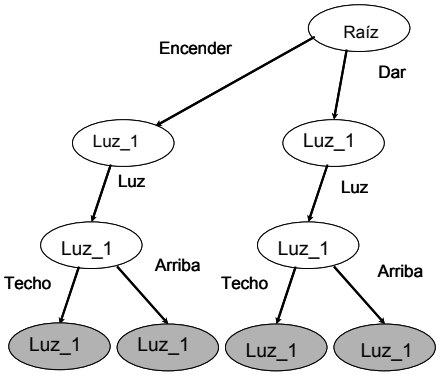

Figura 2. Árbol lingüístico parcial

Los nodos sombreados corresponden con nodos de acción. Cuando el diálogo alcanza uno de estos nodos el sistema ejecuta el método de acción asociado a la entidad a la que pertenece (en este caso encendería la *luz_1*).

Otros conjuntos de partes pueden tener una parte con una palabra al mismo nivel que un conjunto previo. En este caso el sistema no crea un nuevo nodo en el árbol para esa parte, sino que reutiliza el nodo previo añadiéndole, si es necesario, el nombre de la entidad a la que pertenece. Supongamos, por ejemplo, que la entidad *luz_1* tiene dos conjuntos de partes: {“apagar”, “luz”, “”, “techo arriba”, “”} y {“apagar”, “fluorescente”, “”, “”, “”}, que corresponden que tres formas posibles de interactuar con la entidad *luz_1*. En este caso la palabra “apagar” está en el mismo nivel en ambos conjuntos, así que sólo se crea un nodo “apagar” y de él colgarán los nodos “luz” y “fluorescente”. Si ahora tenemos una nueva entidad denominada *radio_1*, con este conjunto de partes lingüísticas {"apagar", "radio", "", "", ""}, el sistema sólo tiene que añadir el nombre de la entidad *radio_1* al nodo “apagar”. A continuación, añade un nodo “radio” como hijo de “apagar”, al mismo nivel que “luz” y “fluorescente”. Empezando con un árbol vacío, el sistema crearía el árbol lingüístico que se muestra en la figura 3.

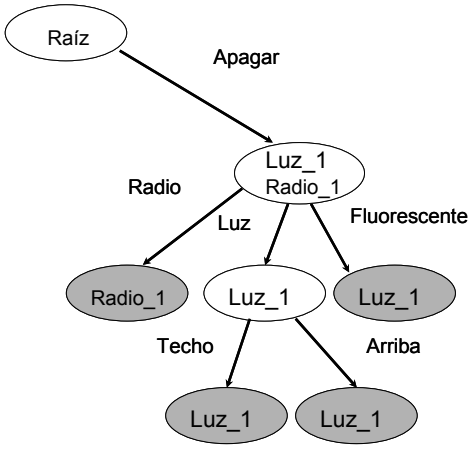

Figura 3. Árbol lingüístico para las entidades luz_1 y radio_1

Este proceso automático es seguido por todas las colecciones de conjuntos de partes de todas las entidades presentes en el entorno inteligente. Una vez completados las gramáticas y el árbol lingüístico, el sistema posee una interfaz de diálogos hablada para todas las posibles interacciones con el entorno actual.

5. INTERPRETACIÓN Y GENERACIÓN

Cuando la interfaz de diálogos se ha completado el sistema y el usuario pueden empezar a mantener conversaciones sobre el entorno. Los diálogos siguen una aproximación de iniciativa mixta [16] donde se asume que los usuarios saben qué pueden decir, dado el contexto de interacción actual. El sistema no tiene que guiar continuamente al usuario y cualquiera de los dos puede tomar la iniciativa.

La interfaz de diálogos se gestiona mediante un supervisor de diálogos, que está a cargo de recibir del reconocedor la frase pronunciada por el usuario, interpretarla y generar un resultado (una respuesta hablada o una acción).

Inicialmente, el sistema está dormido y no reconoce ninguna frase. Si un usuario quiere iniciar una conversación éste tiene que despertarlo pronunciando la palabra “odisea” (el nombre dado a la interfaz de diálogos). En ese momento, siguiendo las ideas definidas por [15], el sistema considera que el objetivo de todas las frases pronunciadas por el usuario es completar una acción. Si éste no dice nada en los siguientes siete segundos (o siete segundos después de la última interacción) el sistema retorna a su estado dormido.

5.1 Interpretación

Cuando el sistema recibe una oración del reconocedor el supervisor la envía módulo de procesamiento de la oración (UPM, por sus siglas en inglés). El UPM busca coincidencias entre la oración y los hijos de la raíz del árbol lingüístico (los nodos

correspondientes a la parte verbal). Si se produce alguna coincidencia, el UPM desciende por el árbol al nodo donde se produjo la coincidencia y busca nuevas coincidencias con sus hijos. Este proceso continúa hasta que el UPM alcanza un nodo de acción (por lo que la oración se ha interpretado completamente) o hasta que no hay más coincidencias (el sistema necesita realizar clarificación).

5.1.1 Ejecutando una acción

Si el UPM alcanza un nodo de acción envía la información del nodo al módulo de procesamiento de nodos (NPM). El NPM obtiene la palabra almacenada en el primer nodo de la ruta que se ha seguido a través del árbol (que corresponde con la parta verbal, esto es, la acción que el usuario ha solicitado que se tome). Entonces, el NPM ordena la ejecución de la parte verbal por parte del método de acción asociado al nodo de acción alcanzado. Esta ejecución normalmente implica un cambio físico en el entorno, aunque también puede producir una respuesta del sistema. Por ejemplo, basándonos en el árbol lingüístico representado en la figura 3, supongamos que el usuario pronuncia la oración "por favor, quiero que apagues la luz del techo". El UPM recibe la oración y comprueba coincidencias con el árbol lingüístico. "Apagues" coincide con el nodo "apagar", "luz" coincide con el hijo de "apagar" y "techo" coincide con el hijo de "luz". El nodo "techo" es un nodo de acción. Por lo tanto el NPM toma el control y obtiene que la acción solicitada por el usuario es "apagar". Como este nodo está asociado a la entidad *luz_1*ejecuta el método de acción de *luz_1,* considerando que el usuario ha solicitado la acción "apagar". Así, la ejecución de este método de acción puede tener dos efectos: puede apagar la *luz_1* o puede informar al usuario de que la *luz_1* ya estaba apagada. En cualquier caso, el UPM considera que la acción se ha completado y regresa a su estado inicial, esperando otras oraciones.

5.1.2 Clarificación

Si, después de que el usuario pronuncie una frase, el UPM no alcanza un nodo de acción, se necesita clarificación. En este caso el TPM vuelve a recibir la información del nodo del UPM. Este, nuevamente, obtiene la parte verbal (la acción solicitada por el usuario) pero ahora no ejecuta ninguna acción. En su lugar el UPM envía la información del nodo al módulo de procesamiento del árbol (TPM). El TPM visita todos los hijos del nodo actual, construyendo una frase de respuesta para solicitar más información al usuario. Esta oración se basa en los nodos visitados y en la información de contexto del entorno representada en la pizarra.

El TPM se basa en una búsqueda recursiva en profundidad [5]. Su procedimiento es:

- Visita el primer hijo del nodo y comprueba si la acción solicitada por el usuario es diferente al estado actual del entorno para alguna de las entidades asociadas al nodo, utilizando para ello la información de la pizarra. En ese caso, almacena el nombre de las entidades con un estado diferente y su nivel en el árbol. Además, obtiene la palabra almacenada en el nodo, para construir la oración de respuesta.
- A continuación continúa de forma recursiva con el primer hijo del nodo que acaba de procesar, siguiendo los mismos pasos.
- Este proceso recursivo con el primer hijo de un nodo se repite hasta que el TPM alcanza un nodo de acción o hasta que el nodo no tiene ninguna entidad con un estado en el entorno diferente al solicitado. Si el TPM alcanza un nodo de acción, y corresponde con una entidad que tiene un estado distinto al estado solicitado por el usuario, incrementa en uno el número de acciones que se pueden ofrecer al usuario y almacena el nombre de la entidad que produce la acción. En cualquier caso, sube un nivel al nodo padre y continúa la inspección recursiva del árbol con el siguiente hijo del nodo.
- Cuando el TPM visita un nodo que no corresponde con el primer hijo (el segundo o sucesivos), primero comprueba que las entidades que contiene no son las mismas que las de otros nodos ya procesados en el mismo nivel. Esto hace posible que los sinónimos se puedan representar en el árbol, pero sólo el primero se considere en los procesos de interpretación y generación.
- Este algoritmo se repite hasta que el TPM ha inspeccionado todos los nodos hijo de la raíz y, si es necesario, sus subsecuentes hijos.

Los hechos de considerar sólo aquellos nodos que tienen entidades con un estado diferente al solicitado por el usuario y de procesar sólo el primer sinónimo hacen que la búsqueda por el árbol sea óptima. El TPM solo tiene que seguir los caminos a los nodos de acción que pueden ser procesados para el estado actual del entorno, dejando de considerar todos los demás.

Una vez que el TPM ha visitado todos los nodos apropiados tiene una respuesta completa (este proceso se explica posteriormente en la sección sobre generación) y el número de acciones que se pueden ofrecer al usuario. El UPM recibe esta información y, dependiendo del número de acciones ofrecidas, se comporta de la siguiente forma:

- Si el número de acciones es igual a cero, el UPM responde al usuario que no hay ningún elemento en el entorno que, con su estado actual, pueda realizar la acción solicitada.
- Si el número de acciones es igual a uno, el UPM directamente ejecuta el método de acción asociado a la única entidad que puede producir una acción. Con esto, se reduce el número de turnos y apoya al reconocedor, al emplear el contexto del entorno actual como una de las posibles fuentes de información que mejoran la interpretación y la comprensión [18].
- Si el número de acciones es dos o tres, el UPM pronuncia la oración de respuesta construida durante la búsqueda del árbol. Esta oración presenta todas las posibles entidades que pueden realizar la acción solicitada, guiando así al usuario por el diálogo [20].
- Si el número de acciones es superior a tres, el UPM no ofrece todas las opciones posibles. En su lugar crea una pregunta de clarificación más general. Con esto el sistema no pronuncia una oración con demasiadas opciones, que puede ser difícil de recordar y tediosa de oír, pero todavía guía al usuario [9].

Como hemos visto la interpretación difiere dependiendo del contexto actual del entorno. La misma oración puede producir diferentes interpretaciones en contextos distintos. Dependiendo

del contexto el UPM puede ejecutar una acción o considerar que el usuario quiere interactuar con algunas entidades del entorno.

Como ejemplo, supongamos dos casos diferentes para el entorno representado en la figura 3. Ambos comparten el mismo escenario, donde la *luz_1* y la *radio_1* están encendidas:

1. En el primer caso el usuario pronuncia "apaga la luz", por lo que el UPM para en el nodo "luz". Entonces el NPM obtiene que la acción solicitada es "apagar" y el TPM empieza el proceso de clarificación. Primero, desciende al nodo "techo" y comprueba que el estado de la *luz_1* en el entorno (encendida) es diferente al estado solicitado (apagar), por lo que procesa el nodo. Entonces añade la entidad *luz_1* a la lista de entidades visitadas bajo el nodo "luz", adjunta la palabra "techo" a la oración de respuesta y, dado que es un nodo de acción, incrementa el número de acciones que se pueden ofrecer y almacena el nombre de la entidad de acción *luz_1*. Después de esto regresa al nodo luz y pasa a inspeccionar el nodo "arriba". Entonces comprueba si las entidades de este nodo están en la lista de entidades procesadas bajo el nodo "luz" (esto es, si es un sinónimo de un nodo procesado previamente). Como el nodo "arriba" pertenece a la entidad *light_1*, que está en la lista de los nodos revisados por el nodo "luz", no lo procesa. Finalmente regresa al nodo original, lo que concluye la búsqueda del árbol. El TPM retorna la oración de respuesta, el número de acciones que se ofrecen y la lista de las entidades de acción al UPM. Este comprueba que sólo existe una posible acción y la ejecuta, esto es, apaga la luz del techo.
2. En el segundo caso el usuario pronuncia "apaga" por lo que el UPM para en el nodo "apagar". Nuevamente, después de que el NPM obtiene que la acción solicitada es "apagar" el TPM empieza con el proceso de clarificación. Primero comprueba que el estado de *radio_1* es diferente al solicitado y que es un nodo de acción. El TPM añade la entidad *radio_1* a la lista de entidades procesadas bajo "apagar", incrementa el número de acciones para ofrecer, adjunta la palabra "radio" a la oración de respuesta y añade la entidad *radio_1* a la lista de entidades de acción. A continuación regresa al nodo "apagar" y después al nodo "luz". Una vez allí funciona como se ha explicado en el caso anterior. Finalmente regresa otra vez al nodo "apagar" y comprueba el nodo "fluorescente". Este nodo contiene la misma entidad que la de un nodo bajo "apagar" previamente procesado (es un sinónimo), así que no se considera. El UPM recibe la oración de respuesta, el número de acciones que se pueden ofrecer y la lista de las entidades de acción. Como el número de acciones para ofrecer es igual a dos pronuncia la oración de clarificación "¿quieres apagar la radio o la luz del techo?". Nuevamente, como ya se ha explicado anteriormente, los múltiples sinónimos se omiten en la respuesta de clarificación (por lo que esta oración sólo se refiere a la luz del techo y no al fluorescente).

Estas mismas oraciones pueden tener diferentes interpretaciones para un contexto de entorno distinto. Supongamos un escenario donde la *radio_1* está encendida y la *luz_1* está apagada. Si el usuario repite las dos oraciones anteriores:

1. Para la oración "apaga la luz", ahora el UPM informará al usuario de que todas las luces están apagadas.
2. Para la oración "apagar", ahora el UPM apagará la *radio_1*, ya que es la única entidad con un estado distinto al estado solicitado por el usuario.

Otra situación posible se produce cuando hay más de una entidad del mismo tipo en el entorno (por ejemplo, dos o más luces). En este caso para la oración "apaga la luz" puede haber tres situaciones posibles:

1. Si todas las luces están apagadas el UPM informará al usuario sobre este asunto.
2. Si sólo una luz está encendida el UPM apagará esa luz.
3. Si hay más de una luz encendida el UPM pronunciará una pregunta de clarificación. Esta oración de clarificación se referirá sólo a aquellas luces que están encendidas, omitiendo aquellas que ya están apagadas.

Como se ha ilustrado en estos ejemplos, la interpretación varía dependiendo del estado actual y de las entidades activas. Se pueden añadir o eliminar entidades, entidades activas puedan cambiar su estado o puede haber múltiples entidades del mismo tipo. La interfaz de diálogos se adapta automáticamente a estas situaciones, alterando su comportamiento.

Finalmente, después de una solicitud de clarificación por parte del sistema, el proceso de interpretación del UPM sufre una ligera modificación en las siguientes interacciones. Como habitualmente, comprueba coincidencias con los nodos del árbol lingüístico, empezando por la raíz. Sin embargo, después de una solicitud de clarificación, también comprueba coincidencias empezando desde el nodo donde paró en la interacción anterior. De ambas búsquedas del árbol, el UPM selecciona el nodo en un nivel inferior o el nodo que corresponde con un nodo de acción. Con esto, el UPM permite continuar con una interacción previa (después de una respuesta de clarificación) o iniciar un nuevo diálogo (dejando atrás el diálogo de clarificación).

5.1.3 Recuperación de errores

Los procesos de reconocimiento, interpretación y clarificación pueden conducir, en algunas ocasiones, a errores de reconocimiento o interpretación. Además, los usuarios pueden no proporcionar información suficiente para procesar totalmente una oración. Para paliar estos problemas el sistema incorpora algunas características específicas.

Como se ha visto en el apartado anterior, después de una respuesta de clarificación el UPM permite continuar con un diálogo previo o iniciar uno nuevo. Con esto el sistema intenta solucionar posibles interpretaciones equivocadas. Si la respuesta de clarificación producida por el sistema no corresponde con ningún objetivo del usuario éste puede iniciar un nuevo diálogo desde el principio, en lugar de tener que continuar con un diálogo equivocado.

Adicionalmente, el UPM no sólo empieza a realizar comprobaciones desde la raíz del árbol lingüístico y desde el nodo donde paró en la última interacción. Si para estos dos nodos no se ha producido ninguna coincidencia, también comprobará coincidencias con cada uno de sus hijos. Con esto el sistema puede subsanar fallos en el reconocimiento o interpretar correctamente oraciones donde el usuario sólo proporcionó parte de la información. Por ejemplo, si para el escenario representado en la figura 3, el usuario pronuncia la frase "… la luz del techo" el UPM no obtendrá ninguna coincidencia para el nodo raíz.

Entonces pasará a comprobar los hijos del nodo raíz, esto es, los hijos del nodo “apagar”, con lo que obtiene una coincidencia para el nodo “luz” y con su nodo hijo “techo”. Esto permite recuperarse de problemas en el reconocimiento y puede interpretar correctamente gracias al uso del contexto del entorno [12] que la oración que el usuario pronunció correctamente era “apaga la luz del techo”.

5.1.4 Resolución de la anáfora

La interfaz de diálogos también implementa un mecanismo de resolución de la anáfora pronominal, para referirse a la última entidad mencionada.

Para permitir el uso de la anáfora, al árbol se añaden nuevos nodos de resolución de la anáfora, uno por cada verbo. Estos nodos se componen por el verbo y un pronombre. Además, después de que el UPM alcance un nodo de acción y ejecute la acción correspondiente, almacena la información relativa al nodo que se considera la última acción ejecutada.

Cuando el UPM alcanza un nodo de resolución de anáfora, esto eso, el usuario ha empleado un pronombre para referirse a un objeto, éste va al nodo que atañe a ese verbo. Una vez allí, desciende el resto del camino que corresponde con la ruta seguida en la última ejecución hasta alcanzar un nodo de acción y ejecuta el método de acción correspondiente. Supongamos que el usuario pronuncia la frase “enciende la radio”. El UPM alcanza el nodo de acción radio y enciende la radio. A continuación almacena el nodo radio como el correspondiente a la última acción ejecutada. En la siguiente interacción el usuario pronuncia “bájala”. El UPM alcanza el nodo de resolución de la anáfora “bajarla” por lo que irá al nodo del árbol “bajar”. Una vez allí sigue la misma ruta que se siguió en la interacción que produjo la última ejecución, por lo que descendería al nodo “radio”. Como este ya es un nodo de acción, ejecuta su método de acción asociado y baja el volumen de la radio.

5.2 Generación

Como se ha visto anteriormente, el proceso de generación se realiza al mismo tiempo que se produce la clarificación. La respuesta se forma con las palabras de aquellos nodos que tienen alguna entidad con un estado distinto al solicitado por el usuario.

Inicialmente la oración de respuesta está vacía. Antes de comenzar con el proceso de clarificación se rellena con la palabra “¿quieres“ y con las palabras que se encuentran en los nodos que van desde la raíz hasta el nodo donde el UPM ha parado. Las palabras se añaden a la oración de respuesta mediante un módulo de adición de palabras (AWM). El AWM obtiene el número y el género de la palabra y anexa la forma correcta, precedida por el artículo adecuado. Por ejemplo, si en la figura 3 el UPM para en el nodo “apagar”, la oración de respuesta estará inicialmente formada por “¿quieres apagar”. A continuación el TPM adjunta el resto de las palabras que va obteniendo durante el proceso de clarificación. Las palabras se añaden como se ha explicado en la sección correspondiente a clarificación. Adicionalmente, si ya se han añadido palabras que estaban al mismo nivel, antes de adjuntar la nueva palabra se añade “o”, para mostrar que se trata de una alternativa. Además, antes de añadir palabras que correspondan a partes modificadoras o de situación se adjunta la palabra “de” y en el caso de objeto indirecto la palabra “a”. Siguiendo con el ejemplo anterior, el TPM utiliza el AWM para adjuntar “la radio”. A continuación, dado que el nodo “luz” está en el mismo nivel que el nodo “radio”, adjunta las palabras “o” y “la luz”. En el siguiente paso, como “techo” corresponde con una parte de situación añade la palabra “de” seguida de “el techo”. En este caso, el AWM une las palabras “de” y “el” en “del”. La oración de respuesta final es “¿quieres apagar la radio o la luz del techo?”.

Como se puede ver la generación también emplea y se adapta al contexto del entorno, obteniendo así de forma automática oraciones apropiadas para cada situación.

Adicionalmente, la interfaz no sólo genera respuestas en forma de oraciones sino que también crea señales de audio no intrusivas [11]. Estas señales de audio son sonidos de ambiente que tratan de proporcionar información sin tener que distraer a los usuarios en el caso de que no sea necesario. Actualmente, los sonidos de ambiente se utilizan en dos casos distintos:

1. Se produce una señal de audio (similar a un bostezo) cuando el sistema retorna al estado dormido. Si esto ocurre porque el usuario finalizó la interacción con el entorno no es necesario distraerle con esta información. Si es por otro motivo, el usuario seguramente sigue prestando atención a la interacción y una señal de audio es suficiente para que conozca el nuevo estado del sistema.
2. Otra señal de audio (similar a una exclamación) se produce cuando el UPM no consigue obtener ninguna coincidencia. Esta señal se repite por segunda vez consecutiva y sólo después de que el UPM no haya podido realizar ninguna interpretación por tercera vez el sistema informa al usuario mediante una oración. Esta señal se introdujo después de comprobar que con algunos usuarios el reconocedor producía, en ocasiones, errores de sustitución o inserción, esto es, proporcionaba una oración diferente a la que éste pronunció o interpretaba el ruido de ambiente como una oración [14]. En el primer caso, comprobamos que es más rápido y eficiente reproducir una señal de audio que pronunciar toda una oración. En el segundo caso un error del reconocedor no tiene por qué distraer innecesariamente al usuario.

6. EVALUACIÓN

El sistema de diálogos experimentó sus primeras pruebas de evaluación cuando se probó durante cuatro días en la IV feria Madrid por la Ciencia. Los visitantes de la feria pudieron utilizar el sistema, interactuar y cambiar el estado físico del entorno inteligente. Los resultados de sus interacciones se podían ver en tiempo real a través de una webcam conectada con el entorno. Múltiples usuarios de diferentes edades y condiciones utilizaron el sistema sin recibir instrucciones especiales sobre su funcionamiento. Como resultado obtuvimos un análisis del corpus de cómo la gente interactúa con el entorno. Esta información ha sido empleada para refinar las gramáticas y el árbol lingüístico. En este punto inspeccionamos que las gramáticas y el árbol lingüístico se creaban de forma adecuada, permitían a los usuarios interactuar con el entorno e implementaban las ideas necesarias para realizar una búsqueda del árbol para la interpretación y la generación.

La interfaz de diálogos ha sufrido algunos cambios después de aquellas pruebas de evaluación, modificando algunas partes y añadiendo nuevas funcionalidades. Por lo tanto, en estos momentos, estamos realizando nuevas pruebas de evaluación que

intentan determinar el éxito de la tarea y los costes de establecer un diálogo [17]. El entorno se ha abierto para uso público, lo que nos permite examinar los procesos de interpretación y generación. El sistema registra las oraciones recibidas del reconocedor, el número de turnos y el tiempo requerido para completar una tarea, las entidades que fueron objeto de la interacción y la respuesta proporcionada. Con esta información intentamos definir el nivel de facilidad para completar una tarea, si las respuestas del sistema resultan de ayuda y son entendidas por los usuarios, si se producen problemas de interpretación o reconocimiento, etc.

Los ejemplos utilizados en el artículo son muy simples para facilitar la tarea de comprensión del rendimiento del sistema. Sin embargo, la interfaz de diálogos que se está probando actualmente está basada en el entorno presentado en la sección dos, donde los usuarios pueden encontrarse con varias luces, un mecanismo de apertura de la puerta de entrada, una radio, etc.

Cada prueba realizada con los usuarios sigue una idea común. Están hechas en un entorno real. Las interacciones de los usuarios producen cambios reales en el entorno y las respuestas del sistema se basan en el estado físico del entorno.

7. CONCLUSIONES Y TRABAJO FUTURO

En este artículo hemos presentado una interfaz de diálogos hablada que se adapta a entornos inteligentes. La adaptación ocurre en los procesos de creación e interacción con la interfaz. En ambos casos la interfaz y su comportamiento dependen del entorno dado y de su estado actual. En el futuro todavía se han de añadir nuevas funcionalidades a las partes correspondientes a la interpretación y generación.

Los diálogos actuales permiten gestionar las entidades del entorno pero no preguntar por su estado. Se deben realizar algunas modificaciones para permitir a los usuarios realizar preguntas. Para ello se debe proveer a la interfaz con una nueva parte interrogativa y con un nuevo árbol de preguntas, muy similar al árbol lingüístico explicado. Sólo algunas características de los módulos de interpretación y generación deben sufrir cambios para permitir la creación de este nuevo árbol.

Actualmente, el sistema sólo realiza una acción por turno. Por ejemplo, si el usuario solicita dos acciones diferentes en la misma oración sólo se considera la primera. El UPM ha de sufrir algunos cambios para permitir que se realicen acciones múltiples y simultáneas.

El uso de aproximaciones multimodales puede beneficiar a la interfaz. Se va a añadir al sistema un nuevo módulo de reconocimiento de caras, que permita identificar quién está dentro del entorno. Esta información se puede utilizar por varios módulos del sistema, incluida la interfaz de diálogos, para mejorar su funcionalidad. Siguiendo con estas ideas, la sincronización del habla y los gestos de las manos puede ayudar a mejorar la interacción [2]. Para ello, se debería construir un nuevo módulo de reconocimiento de gestos. Otros modos de interacción se pueden producir mostrando la información en una pantalla, en lugar de pronunciado una oración. El usuario podría entonces responder mediante el habla o seleccionando la opción adecuada de la pantalla.

8. AGRADECIMIENTOS

Este proyecto está subvencionado por el Ministerio de Ciencia y Tecnología TIC2000-0464.

9. REFERENCIAS

[1] Abowd, G.D. “Software design issues for ubiquitous computing”. IEEE CS Annual Workshop on VLSI: System Level Design (IWV '98), Orlando, FL, April 16-17, 1998.

[2] Bourguet, M. and Ando, A. Synchronization of speech and hand gestures during multimodal human-computer interaction. In Proceedings of CHI’98 (Los Angeles, GA, April 18-23), 241-242, 1998.

[3] Carmona, J.; Atserias, J.; Cervell S.; Márquez, L.; Martí, M.A.; Padró, L.; Placer, R.; Rodríguez, H.; Taulé, M. and Turmo, J. “An Environment for Morphosyntactic Processing of Unrestricted Spanish Text”. Proceedings of 1st International Conference on Language Resources and Evaluation (LREC'98), Granada, Spain, 1998.

[4] Coen, M.H. “Design Principles for Intelligent Environments”. Proceedings of the AAAI Spring Symposium on Intelligent Environments (AAAI98). Stanford University in Palo Alto, California, 1998.

[5] Cormen, T.H., Leiserson, C. E., and Rivest, R. L.. Introduction to Algorithms. The MIT Press. Cambridge, Massachusetts. London, England. 2001.

[6] Dahlbäck, N. and Jönsson, A. “An empirically based computationally tractable dialogue model”. Proceedings of the 14th Annual Conference of the Cognitive Science Society (COGSCI’92), July 1992.

[7] Engelmore, R. and Mogan, T. Blackboard Systems. Addison-Wesley, 1988.

[8] Martínez, A.E., Cabello, R., Gómez, F. J. and Martínez, J. “INTERACT-DM. A Solution For The Integration Of Domestic Devices On Network Management Platforms”. IFIP/IEEE International Symposium on Integrated Network Management. Colorado Springs, Colorado, USA, 2003.

[9] Marx, M. and Schmandt, C. MailCall: Message presentation and navigation in a nonvisual environment. In Proceedings of CHI’96 (Vancouver, April 13-18), 1996.

[10] Montoro, G.; Alamán, X. and Haya, P. A plug and play spoken dialogue interface for smart environments. In Proceedings of Fifth International Conference on Intelligent Text Processing and Computational Linguistics (CICLing'04). Seoul, Korea. February 15-21, 2004.

[11] Mynatt, E.D.; Back, M.; Want, R. and Frederick, R. Audio Aura: Light-weight audio augmented reality. In Proceedings of ACM UIST’97 (Banff, Canada), 211-212, 1997.

[12] Nagao, K. and Rekimoto J. Ubiquitous talker: Spoken language interaction with real world objects. In Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI-95), Vol. 2, 1284-1290, 1995.

[13] Salber, D. and Abowd, G.D. The design and use of a generic context server. In Proceedings of Perceptual User Interfaces (PUI’98), 1998.

[14] Schmandt, C. and Negroponte, N. Voice communication with computers: conversational systems. Van Nostrand Reinhold, New York. 1994.

[15] Searle, J. Speech Acts. Cambridge University Press. London, 1969.

[16] Walker, M.A.; Fromer, J.; Di Fabbrizio, G.; Mestel, C. And Hindle D. What can I say?: Evaluating a spoken language interface to email. In Proceedings of CHI'98 (Los Angeles, GA, April 18-23), 582-589, 1998.

[17] Walker, M.A.; Litman, D.J.; Kamm, C.A. and Abella, A. PARADISE: A framework for evaluating spoken dialogue agents. Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics. 1997.

[18] Ward, K. and Novick, D.G. Integrating multiple cues for spoken language understanding. In Proceedings of CHI'95 (Denver, CO, May 7-11), 1995.

[19] Weiser, M. The computer of the 21st century. *Scientific American*, 265, 3, 66-75, 1991.

[20] Yankelovich, N. "How do users know what to say?" ACM Interactions, 3, 6, December, 1996.