ATOS: UN SISTEMA DE CONTROL AUTOMÁTICO DEL TELÉFONO MEDIANTE COMPUTADOR

Daniel Tapias Merino
Jorge Álvarez
Ismael Cortázar

En este artículo presentamos la primera versión del sistema conversacional ATOS (Automatic Telephone Operator Service) desarrollado en la división de tecnología del habla de Telefónica I+D [5]. Este prototipo integra reconocimiento de voz, procesado del lenguaje natural y conversión texto voz y funciona en una estación de trabajo comercial.

ATOS es un sistema interactivo dirigido por medio de la voz que permite realizar tareas como configurar terminales telefónicos, crear y modificar una agenda personal, consultar un directorio de números de teléfono o realizar llamadas de diversos tipos. Con este sistema se evita que el usuario tenga que memorizar no solo números de teléfono, sino también las instrucciones precisas para configurar el terminal telefónico o para acceder a todas las opciones que permiten las PABX actuales. Adicionalmente, facilita la marcación, que puede realizarse diciendo el número o el nombre de la persona o empresa. El tiempo medio de respuesta de la versión actual es de tres segundos.

En los siguientes apartados se describen las funcionalidades del sistema ATOS y de sus componentes: el reconocedor de voz, el módulo de procesado del lenguaje natural -compuesto a su vez por el analizador semántico y el gestor de diálogo- y el conversor texto voz.

1. FUNCIONALIDADES Y DESCRIPCIÓN DEL SISTEMA

En la actualidad, las PABX ofrecen una gran variedad de servicios que pueden ser accedidos desde los terminales telefónicos conectados a ellas. La cantidad de servicios a los que se tiene acceso desde un teléfono, depende por un lado del propio teléfono y por otro de la categoría asignada al mismo en la PABX. Así, por ejemplo, la capacidad de realizar llamadas internacionales, locales, etc... depende de la categoría del terminal mientras que la posibilidad de visualizar el número de teléfono de la llamada entrante depende fundamentalmente de si el terminal tiene display o no. En cualquier caso, aunque el número de funciones disponible es variable, hay un gran número de utilidades que pueden ser usadas desde los teléfonos comunes.

El inconveniente de estos servicios es que su utilización implica memorizar códigos o consultar el manual de usuario para recordarlos, lo que desanima a muchos usuarios que finalmente no emplean estas facilidades o solo usan un pequeño conjunto de las mismas.

El prototipo del sistema ATOS pretende solventar este problema, permitiendo al usuario explicar con su propia voz el tipo de servicio que quiere emplear. Actualmente, este prototipo dialoga con el usuario hasta que averigua que tipo de funcionalidad desea utiliza y accede a las bases de datos que posee para proporcionar la información solicitada o incluir nueva información. En su segunda versión, que está en fase de desarrollo, realizará las acciones necesarias para ofrecer todos los servicios solicitados.

Las funcionalidades que ofrece este sistema se pueden dividir en tres grupos que se describen a continuación:

- Servicios de PABX: El prototipo es capaz de entender las ordenes que manejan las funcionalidades ofrecidas por una PABX como rellamada, transferencia de llamada multiconferencia, contestador automático, etc...

- Servicio de directorio telefónico: Permite realizar llamadas de teléfono diciendo el nombre y primer apellido de la persona o pronunciando directamente el número de teléfono.

- Servicio de agenda personal: Cada usuario puede crear su propia agenda para completar el directorio de uso general. Tanto el acceso a esta agenda como su creación se realiza por medio de la voz, sin necesidad de emplear un teclado.

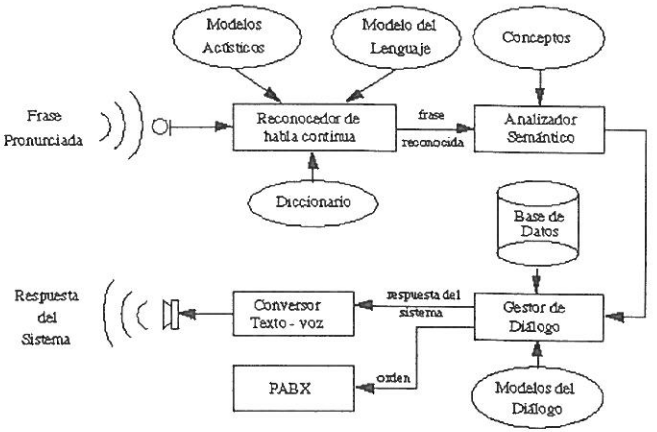

Figura 1: Esquema general del sistema conversacional ATOS

El funcionamiento del sistema se ilustra en la figura 1; Una vez que el usuario ha pronunciado una frase, el reconocedor de habla continua, que es independiente del locutor,

entrega a su salida la frase reconocida. Posteriormente, el analizador semántico se encarga de buscar palabras o secuencias de palabras que tengan información relevante para la aplicación. Esta información se almacena en estructuras semánticas, que son las entidades mínimas de significado empleadas por el sistema. Si el gestor de diálogo tiene completas todas las estructuras necesarias para realizar una acción, la orden se ejecuta, en caso contrario, genera una frase que es pronunciada por el conversor texto voz solicitando al usuario la información que necesita.

El sistema realiza tres tipos de acciones:

- envio de órdenes a la PABX,
- consulta a la base de datos y lectura de los datos obtenidos y
- modificación de la base de datos.

En la figura 2 se muestra el ejemplo de un diálogo en el que el usuario desea añadir una nueva entrada en su agenda y realizar una llamada telefónica.

Usuario: Quería añadir a Carlos en mi agenda personal.

ATOS: ¿ Con qué número de teléfono ?

Usuario: Su número de teléfono es el 337-42-30 de 8 de la mañana a 6 de la tarde y el 454-31-95 el resto del tiempo y los fines de semana

ATOS: Se ha incorporado a Carlos en la agenda personal con el número 337-42-30 de 8 de la mañana a 6 de la tarde y el 454-31-95 de 6 la tarde a 8 de la mañana y los fines de semana.

Usuario: Llama a Jorge

ATOS: Se está llamando a Jorge al teléfono de su casa

Usuario: No, llámale al trabajo, que hoy trabajará hasta tarde.

Figura 2: Ejemplo de diálogo

2. RECONOCEDOR DE HABLA CONTINUA

El módulo de reconocimiento maneja un vocabulario de 2000 palabras en la versión actual, aunque tiene capacidad para reconocer vocabularios de hasta 50000 palabras, y emplea un modelo estadístico del lenguaje, por lo que se permite cualquier combinación de las palabras que constituyen el vocabulario. Esta característica del sistema es muy útil en aplicaciones en las que los usuarios se expresan empleando habla natural, ya que el habla natural, a menudo, no es gramatical.

Adicionalmente, el sistema es capaz de reconocer frases pronunciadas por cualquier persona independientemente de las características de su voz, lo que permite que el sistema ATOS pueda dar servicio a cualquier usuario que hable castellano. La tasa media de error de palabra medida en condiciones de laboratorio es del 95%. Los errores más frecuentes del

sistema son los debidos a falta de concordancia en género, número y persona así como inserciones y omisiones de artículos y preposiciones. En general estos errores no modifican el significado de la frase, lo que permite que el analizador semántico sea capaz, en la mayoría de los casos, de extraer las células de información de la frase reconocida.

El módulo de reconocimiento emplea unidades dependientes del contexto, que se modelan por medio de modelos ocultos de Markov semicontínuos. Las unidades empleadas, llamadas trifonemas, están constituidas por el sonido central y sus contextos derecho e izquierdo. Así, por ejemplo, el sonido "m" de la palabra "puma" estaría representado por el trifonema "m(u,a)", que sería distinto del trifonema que representa a sonido "m" en la palabra "amperio" representado por "m(a,p)".

El reconocedor emplea tres etapas en el proceso de decodificación de la frase reconocida. En la primera etapa calcula la probabilidad de que los modelos acústicos y el modelo del lenguaje hayan generado la frase pronunciada. Durante este proceso se crea una red de posibles palabras reconocidas en cada instante de tiempo. En las dos etapas posteriores se recalculan las probabilidades en dicha red, buscando la frase cuya probabilidad global es máxima.

Para la tarea ATOS se ha configurado este reconocedor de voz para que utilize 24 unidades independientes del contexto (una para cada fonema) y 5700 unidades dependientes del contexto. El léxico utilizado consta de 2000 palabras básicas a las que se les han añadido 800 palabras más para poder reconocer nombres y apellidos.

3. MÓDULO DE PROCESADO DEL LENGUAJE NATURAL

En esta sección se presentan los dos bloques más importantes de procesador de lenguaje natural, como son el analizador semántico y el gestor de diálogo.

El analizador semántico es un módulo que va buscando palabras o secuencias de palabras que tengan información relevante desde el punto de vista de la aplicación. A cada trozo de información relevante se le denomina concepto; los conceptos son las entidades de significado mínimas empleadas por el sistema. El analizador semántico tiene almacenados patrones asociados con los distintos conceptos que maneja, de forma que cuando una parte de la frase se identifica con un patrón se extrae la información y se almacena en los conceptos mencionados.

El gestor de diálogo (GD) es el módulo que controla la aplicación. Recibe los conceptos detectados por el analizador semántico y los agrupa según los marcos semánticos definidos en la tarea. Si se completa algún marco semántico necesario para ejecutar una orden, el gestor de diálogo se encarga de su ejecución. En caso contrario, solicita al usuario la información que falte. Así pues este módulo realiza básicamente cinco funciones: control de la aplicación, agrupación de conceptos, acceso a bases de datos, ejecución de acciones y generación de lenguaje natural.

Tanto el analizador semántico como el gestor de diálogo están diseñados de forma que pueden adaptarse fácilmente a distintas tareas. Para ello no hay más que cambiar unos ficheros de configuración en los que se definen en primer lugar los conceptos útiles junto con las expresiones que lo validan, y por otro lado los marcos semánticos de la aplicación

junto con las acciones y las distintas situaciones a las que puede llevar el diálogo. En los siguientes apartados se explica más en detalle el funcionamiento de ambos bloques.

3.1. *Analizador semántico*

El módulo de análisis semántico tiene como objetivo detectar dentro de una frase todos los conceptos que pueden ser útiles dentro de una tarea. Para ello es necesario en una primera etapa especificar cuáles son los conceptos útiles y cómo se pueden definir mediante reglas para construir una red por cada concepto. En una segunda etapa se encontrará la mejor manera de enfrentar la frase de entrada con todas las redes que se han definido. A continuación se expondrán las principales características de los módulos que realizan estas dos etapas.

3.1.1. *Construcción de redes para el analizador semántico*

Se trata de generar una red de estados finitos por cada concepto que se defina. Un concepto se define mediante una serie de palabras clave que se combinan siguiendo reglas. El constructor de redes está encargado de interpretar estas reglas, validarlas y de comprobar los posibles errores que se hayan cometido en el proceso de definición de la red. Con las palabras y las reglas construye para cada concepto una red en formato numérico que ya es directamente legible por el analizador semántico. A continuación se describen las reglas y elementos utilizados en la definición de un concepto.

Los elementos utilizados para definir un concepto son:

- **[concepto].** Cualquier palabra entre paréntesis cuadrados expresa un concepto.
- **NO_TERMINAL.** Cualquier palabra en mayúsculas se refiere a un elemento NO_TERMINAL. Un NO_TERMINAL es un elemento bajo cuyo nombre se agrupan más reglas.
- **terminal.** Cualquier palabra en minúsculas se refiere a un terminal.

Las reglas utilizadas en la definición están compuestas de la "parte izquierda de la regla" y de la "parte derecha de la regla".

- **La parte izquierda de la regla** siempre es un [concepto] o un NO_TERMINAL, lo que indica que a continuación se va a definir la combinación de palabras que validan dicho [concepto] o NO_TERMINAL.
- **La parte derecha de la regla** viene siempre dada entre paréntesis y consiste en la combinación de elementos que van a definir bien un [concepto] o bien un NO_TERMINAL. Los elementos que se combinan pueden ser [conceptos], NO_TERMINALES ó terminales, y se combinan bien secuencialmente, o mediante dos operadores: "*" indica que el elemento es opcional (aparece cero ó más veces) y "+" indica que el elemento puede aparecer una ó más veces.

A modo de ejemplo, se define a continuación el concepto [hacer_llamada] basado en los elementos y reglas anteriormente citados, donde TELEFONO es un NO_TERMINAL, y [nombre] un concepto insertado en otro concepto más general. Ambos elementos son opcionales.

EJEMPLO:

[hacer_llamada]

(hacer llamada *TELEFONO *[nombre])

(llamar *[nombre])

TELEFONO

(telefónica)

(de teléfono)

Basándose en esta definición del concepto [hacer_llamada] el constructor de rede genera su correspondiente red de estados que se ilustra en la figura 3.

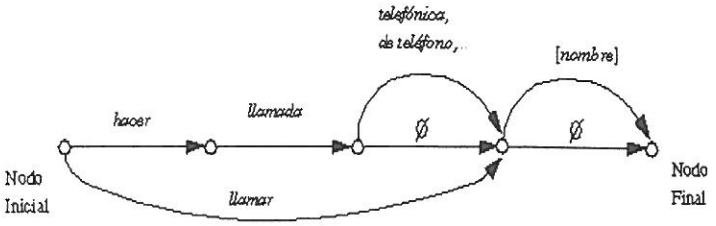

Figura 3: Red construida para el concepto [hacer_llamada]

3.1.2 *Enfrentamiento de la frase de entrada con las redes*

Es el principal módulo del analizador semántico. Su misión es utilizar las palabras de l frase de entrada para intentar recorrer el máximo número de redes posibles. E funcionamiento básico para una *red*, comenzado desde una *palabra*, es el siguiente:

1. Dada una palabra de entrada, se seleccionan una red.
2. Se comprueba si la palabra es útil para transitar desde el nodo activo de la red hast el siguiente nodo.
3. Si es útil, se transita y por tanto se cambia el nodo activo de la red. A continuació se coge la siguiente palabra de entrada y se intenta utilizar para transitar otra ve desde el nuevo nodo activo, por lo que se vuelve al punto 2. Esta tarea se realiz llamando de nuevo a la función enfrentar(siguiente palabra, misma red), de dond se puede deducir la naturaleza recursiva de la función enfrentar.
4. Si no es útil, con la misma palabra se intenta una nueva red. Esta tarea se realiz con la función enfrentar(misma palabra, siguiente red), de donde se deduce tambié su naturaleza recursiva.

El mismo procedimiento ha de seguirse con todas las palabras de la frase para comprobar si se ajustan a todas las redes.

Si se analiza en profundidad este algoritmo, puede verse cómo después de utilizar una palabra para transitar en una red se intenta utilizar *la siguiente palabra* dentro de la misma red; de esta forma es posible llegar rápidamente al nodo final de una red sin tener que comprobar antes otras redes. Solamente cuando esta red se haya completado se probarán las siguientes. Este comportamiento se ajusta a un algoritmo de búsqueda deep-first, que enfoca su búsqueda directamente hacia los caminos que se vayan a finalizar.

Después de haber detectado en la frase todas las redes posibles se hace un alineamiento de las redes completadas, y se selecciona aquel conjunto de redes que sin solaparse, obtienen la mejor puntuación [1]. A la mejor combinación de redes se le denomina interpretación y es la información que se le pasará al gestor de diálogo.

3.2 *Gestor de diálogo*

El *gestor del diálogo* [4] es el motor del sistema; controla el comportamiento del sistema global y es el órgano que realiza la comprensión propiamente dicha. El gestor de diálogo recibe todos los conceptos generados por el analizador semántico -que hasta el momento son simples piezas de conocimiento- y realiza una *unificación de conceptos* utilizando los marcos semánticos que se han definido en la tarea y que tienen significado global [2]. Estos marcos definen la forma en que unos conceptos se relacionan con otros para conseguir comunicar un objetivo. Un ejemplo del proceso de comprensión que realiza el gestor de diálogo puede verse con la frase reconocida "Por favor, quiero apuntar el teléfono de Juan en la agenda". Con esta frase el analizador semántico ha detectado los conceptos [anotar_en_agenda] y [persona] por separado. El gestor de diálogo realiza el proceso de comprensión al unificar los conceptos en la estructura *anotar_en_agenda* que se muestra en la figura 4. Una vez que los conceptos han rellenado parcialmente la estructura con sus valores, nos encontramos con una estructura incompleta *anotar_en_agenda(Juan, número_de_teléfono)* que habrá de completarse con el concepto *[nombre_de_persona]*. La ausencia de este concepto dará las directrices para generar la siguiente pregunta del diálogo.

Para un correcto desarrollo del diálogo el gestor de diálogo utiliza dos tipos de estrategias: una global, encargada de unificar los conceptos y detectar los que faltan, y otra local encargada de encontrar el mejor modo de preguntar ó informar al usuario en cada momento.

Para la estrategia global del diálogo se utiliza un *árbol jerárquico* semejante al de la figura 5 que organiza la información de cada tarea . El nodo raíz representa la tarea global (ATOS), y los nodos del primer nivel representan todas las funciones y órdenes que el sistema puede realizar y comprender (activar el modo no molesten, hacer una llamada,...). Cada nodo posee a su vez nodos hijos que representan los parámetros/valores que el nodo necesita para ser completado. Por ejemplo para anotar un número en la agenda se necesita conocer el nombre de la persona a anotar (lo cual se puede saber con el concepto "nombre de persona") y su número de teléfono (que se conoce bien con el concepto "número de teléfono" ó con el concepto "extensión"). Recorriendo este árbol el gestor de diálogo puede rellenar ordenadamente las estructuras semánticas y puede identificar en cada momento los

párametros que faltan por rellenar y que serán pedidos al usuario. La mayor ventaja de es estrategia global del diálogo organizada en árbol radica en su modularidad y *fác adaptación a nuevos diálogos*, lo que permite diseñar nuevas tareas inmediatamente, s más que adaptar el fichero de con**figuración del árbol jerárquico.**

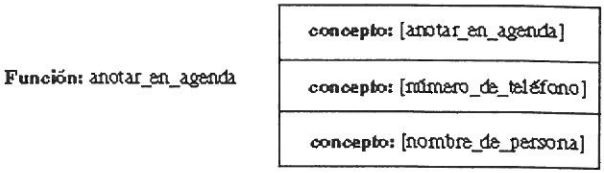

Figura 4: Marco semántico para la función "anotar en agenda" y sus conceptos asociados.

La estrategia local del diálogo es necesaria para que la conversación no sea idéntica e cualquier punto del diálogo. Por ejemplo cuando un dato sea crítico para la buer comprensión de una función será necesario pedir al usuario que confirme si el dato se h reconocido correctamente; sin embargo para datos menos críticos no es necesario ped ninguna confirmación. Otro ejemplo similar aparece cuando para completar una funció sólo es necesario rellenar uno de sus posibles parámetros (para llamar a alguien e necesario conocer bien su número de teléfono, bien su nombre); por este motivo se ha definido sobre la jerarquía del diálogo una serie de tipos de nodos que hacen comportarse gestor de diálogo de manera diferente en cada caso, de forma que dependiendo d comportamiento deseado en cada nodo del diálogo se pueda definir ese nodo como del tip más adecuado. Uno de los tipos de nodo más simple que se ha definido puede verse en figura 6. Consiste en un nodo que necesita pedir confirmación de su valor. En este cas cuando el gestor de diálogo se sitúa en este nodo por primera vez éste se encuentra en estado vacío (sin rellenar) entonces saca un mensaje de petición de su valor; cuando s recibe su valor, el gestor de diálogo transita hacia el estado lleno y saca un mensaje d confirmación del valor recibido. Si la confirmación es negativa, el nodo vuelve al estad vacío; si la confirmación es positiva entonces pasa al estado "confirmado" y se fija así s valor.

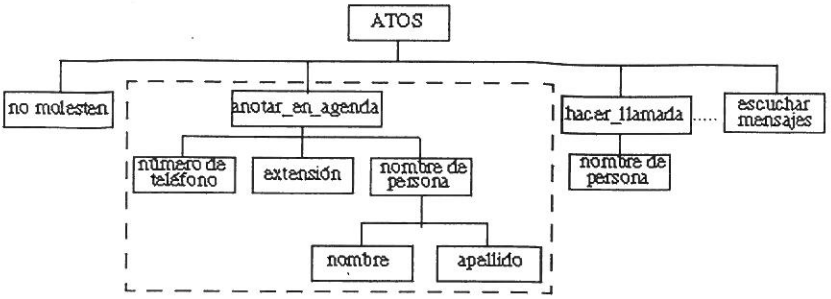

Figura 5: Estructura jerárquica del diálogo. Marco semántico "hacer_llamada"

4. CONVERSOR TEXTO-VOZ

Este módulo es el encargado de leer los mensajes generados en el gestor de diálogo como consecuencia de la interacción entre el usuario y el sistema ATOS.

El conversor texto voz [3] está basado en concatenación de unidades y emplea un sintetizador LPC multipulso. Las principales características de este sistema se detallan a continuación:

- Generación automática de la entonación de la frase.
- Disponibilidad de varias voces.
- Tablas de excepciones configurables por el usuario
- Capaz de leer temperaturas, horas, fechas, acrónimos, etc...

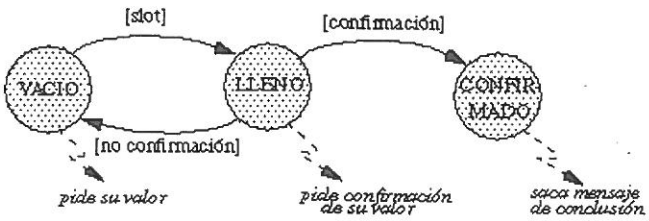

Figura 6: Tipo de nodo que requiere confirmación de su valor.

5. RESULTADOS EXPERIMENTALES

El servicio telefónico ATOS ha sido probado en laboratorio por 30 personas que n estaban involucradas en su desarrollo. A cada usuario se le ha pedido que ordene mediant voz todas y cada una de las 30 funciones que ofrece el servicio ATOS, y que dialogue co el sistema hasta que consiga completar la función con éxito. Para configurar las 3 funciones del sistema de diálogo ha sido necesario definir 41 conceptos diferentes que s describen usando 2000 palabras. A continuación se presentarán los resultados parciale obtenidos hasta el momento. Debido a la existencia de múltiples módulos en el sistema d diálogo, en lugar de obtener una tasa de error global se ha optado por evalua independientemente cada una de sus partes, lo que ha dado lugar a cuatro medida diferentes:

En primer lugar el reconocedor de voz tiene una tasa de error de palabra del 25%; s bien esta tasa de error es grande hay que hacer hincapié en la dificultad de la tarea ya que s trata de habla espontánea, y en que casi el 8% de las palabras que fueron pronunciadas n habían sido previstas en el vocabulario (palabras fuera del vocabulario).

El procesador de lenguaje natural (analizador semántico y gestor de diálogo) proces correctamente el 93.5% de las frases que recibe cuando nos ceñimos a los casos en los qu las frases no son ambiguas dado que provienen de texto escrito.

Cuando las frases provienen del reconocedor de voz, es muy probable que en ella aparezcan errores de reconocimiento, como pueden ser palabras insertadas, palabra omitidas, ó palabras confundidas por otras. En el caso de que estos errores no sean mu graves, el procesador del lenguaje natural tiene todavía capacidad de extraer los principale conceptos de la frase e interpretarlos adecuadamente. En la evaluación realizada, cuando la frases provienen del reconocedor (recuérdese que tenía un 25% de error) y además la fras reconocida sigue teniendo sentido para una persona, entonces el procesador de lenguaj natural es capaz de interpretar correctamente el 92.1% de las frases.

Se ha obtenido también una medida global del funcionamiento del conjunto d reconocedor y procesador de lenguaje natural frase por frase. Esta medida consiste e pronunciar una frase y observar si el sistema realiza una pregunta coherente, que es válid para completar la información de la función pedida. En este caso la tasa de error por fras se ha elevado al 22.1% de las frases pronunciadas.

6. CONCLUSIONES

En el presente artículo se ha presentado un prototipo de sistema conversacional formad por un módulo reconocedor de voz y un módulo de procesamiento del lenguaje natural. L idea que inspira este sistema proviene del análisis de los errores de reconocimiento qu muestran que en la mayoría de los casos es posible extraer las ideas principales del text reconocido aunque éste posea errores.

Creemos que una de las principales características que deben tener estos sistemas es s fácil adaptación a diversas tareas. Si pensamos en aplicaciones de las tecnologías del habla nos daremos cuenta que actualmente sólo tienen sentido en tareas de ámbito reducid (control de máquinas, acceso a bases de datos, etcétera) donde las prestaciones de

reconocedor pueden ser suficientemente buenas. Sin embargo se hace imprescindible que el mismo programa se pueda utilizar en muchas tareas y que su adaptación a una nueva tarea se haga principalmente mediante ficheros de configuración, característica que cumplen todas las partes de nuestro sistema de diálogo.

Las prestaciones obtenidas por el prototipo superan a las del reconocedor por separado, lo cual indica que el procesador de lenguaje natural es un método que aumenta la robustez del sistema. Las tasas de error empiezan a ser aceptables, con lo cual creemos que este tipo de sistemas conversacionales podrán ofrecer servicio real en un corto plazo,

7. REFERENCIAS

[1] Issar, S. and Ward, W. "CMU's Robust Spoken Understanding System". Proc. Eurospeech'93. Berlin.

[2] Young, S.J. and Proctor C.E. "The Design and Implementation of Dialogue Control in Voice Operated Database Inquiry Systems". Computer Speech and Language (1989) 3, Pag. 329-353.

[3] Rodríguez, M.A., Escalada, J.G., Macarrón, A. and Monzón, L. "AMIGO: Un Conversor Texto-Voz para el Español". SEPLN VIII Congress. Barcelona. Feb 93.

[4] Caminero-Gil, J., Alvarez-Cercadillo, J., Crespo-Casas, C., and Tapias-Merino, D. "Data-Driven Discourse Modelling for Semantic Interpretation". ICASSP-96. Atlanta. USA.

[5] Alvarez-Cercadillo, J., Caminero-Gil, C., Crespo-Casas, C., and Tapias-Merino, D. "The Natural Language Processing Module for a Voice Asisted Operator at Telefónica I+D". ICSLP-96. Philadelphia. USA.