

APLICACIÓN DE LEKTA AL ENTORNO ATOS

María Teresa López Soto
Depto. Lengua Inglesa. Universidad de Sevilla
Palos de la Frontera, s/n. 41004 Sevilla
Tel: +34-5-455.1588. Fax: +34-5-455.1516
email: teresa@fing.us.es

José Francisco Quesada Moreno
CICA (Centro Informático Científico de Andalucía)
Avda. Reina Mercedes, s/n Sevilla
email: josefran@cica.es

Jorge Álvarez Cercadillo
División Tecnología del Habla. Telefónica I+D
Emilio Vargas, 6 Madrid
email: jorge@craso.tid.es

En este artículo analizamos los resultados obtenidos a partir del análisis del corpus ATOS con LEKTA. Nos centraremos en la descripción del analizador semántico que, en la actualidad, está implementado en ATOS y lo compararemos con los resultados obtenidos del análisis hecho con LEKTA. El objetivo de este artículo es demostrar la eficiencia de LEKTA como módulo de procesamiento del lenguaje natural (PLN) en ATOS. LEKTA analiza correctamente las oraciones ofrecidas por un reconocedor de voz y asigna la función telefónica deseada.

1. Motivación.

En un artículo publicado en SEPLN en 1996 [1], presentamos por primera vez ante la comunidad científica este proyecto que se enmarca en un acuerdo de colaboración entre Telefónica I+D y la Universidad de Sevilla. En dicho artículo se describía el sistema ATOS y se planteaba la posibilidad de incluir a LEKTA como alternativa al analizador semántico ya existente. La motivación teórica en la implementación de LEKTA como módulo de PLN en ATOS no es otra que la utilización de técnicas de PLN para la detección y corrección de los errores que genera el reconocedor de voz. Asimismo, la motivación práctica deriva del nivel de eficacia demostrado por las técnicas de PLN, que utilizan múltiples fuentes de conocimiento: léxico, sintáctico y funcional.

2. Descripción General del Sistema ATOS.

ATOS (Automatic Telephone Operator Service) es un sistema conversacional desarrollado en la División de Tecnología del Habla de Telefónica Investigación y Desarrollo [2]. Este prototipo integra reconocimiento de voz, PLN y conversión texto-voz.

ATOS es un sistema interactivo dirigido por medio de la voz que permite realizar tareas como configurar terminales telefónicos, crear y modificar la agenda personal, consultar el directorio de números de teléfonos o realizar llamadas de diversos tipos. Con este sistema se evita que el usuario tenga que memorizar, no sólo números de teléfono, sino también los códigos necesarios para configurar el terminal telefónico o para acceder a todas las opciones que permiten las PABX actuales. Además, facilita la tarea de marcar, que puede realizarse diciendo el número o el nombre de la persona o empresa.

El inconveniente actual de estos servicios es que su utilización implica memorizar códigos o consultar el manual de usuario cada vez que se desea activar una función. Esto puede llegar a desanimar a muchos usuarios, que, finalmente, no emplean estas facilidades, o sólo se limitan a usar algunas, normalmente las más simples de activar.

El prototipo ATOS ofrece al usuario la facilidad de poder activar todas las funciones telefónicas empleando el lenguaje natural y sin necesidad de consultar un manual.

El funcionamiento del sistema se ilustra en la figura 1.

Figura 1

Una vez que el usuario ha pronunciado una frase, el reconocedor de habla continua, que es independiente del locutor, devuelve como 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 (GD) tiene completas todas las estructuras necesarias para realizar una acción, la orden se ejecuta. En caso contrario, inicia un diálogo pronunciado por el conversor texto-voz solicitando al usuario la información que necesite.

El sistema realiza tres tipos de acciones:

- envío 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.

Figura 2

El módulo de reconocimiento maneja un vocabulario de 2.000 palabras, aunque tiene capacidad para reconocer vocabularios de hasta 50.000 palabras, y emplea un modelo estadístico del lenguaje, por lo que se permite cualquier combinación de las palabras que constituyen el vocabulario. 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, medida en condiciones de laboratorio, es del 95% de acierto. Los errores más frecuentes del sistema son los debidos a la falta de concordancia en género, número y persona, así como inserciones y omisiones de palabras. En contadas ocasiones, el sistema reconoce una palabra muy distinta de la que realmente fue dicha.

3. Descripción General del Sistema LEKTA.

LEKTA [3, 4, 6] fue diseñado como sistema de TA basado en LFG [5] y en un enfoque de transferencia. En la figura 3 se muestra cómo se integran la teoría LFG y el enfoque de transferencia en el sistema.

Figura 3

En el sistema se pueden observar, por tanto, tres módulos bien definidos. En primer lugar, el módulo de análisis devuelve una doble representación sintáctica (estructura de constituyentes y estructura funcional) para cada oración gramatical según las reglas del sistema. A continuación mostramos un ejemplo parcial de gramática en LEKTA:

```
Sag
  (RG: S)
  (GF: subj obj)

  (1: VP -> NP VP)
  {
    UP = SELF-2;
    IF (MEMBER(subj, SELF-2.ggf))
      THEN {UP.subj = SELF-1};
    IF (MEMBER(obj, SELF-2.ggf))
      THEN {UP.obj = SELF-1};
    COMPLETENESS(GF)
  }
```

El siguiente módulo, el de transferencia, toma como entrada la estructura funcional obtenida a partir del análisis sintáctico. Un ejemplo de estructura funcional en forma de matriz puede verse en la figura 4.

Figura 4

A partir de esta información, y siguiendo las reglas definidas en el módulo de transferencia, LEKTA devuelve la estructura funcional en la lengua destino, tal y como puede observarse en la figura 5.

Figura 5

A continuación se puede observar un ejemplo de reglas de transferencia:

```
(almacenar => AlmacenarUltimoEscuchado WHEN (ggf:[obj])
  DO (obj:NOTTRANSFER()))
```

Finalmente, el módulo de generación produce la estructura de constituyentes en la lengua de destino.

4. LEKTA como Analizador Semántico en ATOS.

La tarea ATOS consiste, en esencia, en trasladar la información semántica que se ha extraído de la oración enunciada hasta un sublenguaje de funciones, generando la función telefónica pertinente. Podemos considerar que el corpus de oraciones producidas en lenguaje natural es la lengua de origen, y el sublenguaje de funciones telefónicas es la lengua de destino. Siendo esto posible, LEKTA trabaja en esta tarea como lo haría en un entorno de TA: trasladar la información semántica desde una lengua origen hasta su traducción en la lengua de destino.

Como veremos, LEKTA ofrece ciertas ventajas como módulo de análisis semántico si lo comparamos con el ahora existente en ATOS. Esta conclusión se deriva de los resultados obtenidos en un primer análisis de las oraciones del corpus ATOS con LEKTA, y de otros motivos que se exponen posteriormente. En primer lugar se define el funcionamiento del analizador semántico que existe actualmente en ATOS, en segundo lugar, enumeramos las ventajas generales que supone la inclusión de LEKTA en este sistema telefónico.

4.1. Descripción del Analizador Semántico de ATOS.

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 de patrones para cada concepto. Es decir, el analizador semántico es un módulo que trata de identificar palabras o secuencias de palabras que tengan información relevante desde el punto de vista de la aplicación. A cada fragmento 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 los conceptos que se analizan para cada frase pueden extraerse para su identificación con el patrón asociado. Se trata, en definitiva, de crear una especie de red semántica referente a la tarea pertinente.

la información funcional que será utilizada más tarde. Veamos el ejemplo de la regla que predice las construcciones transitivas, donde se especifica que el NP ha de unificar como objeto de la oración.

```
(X: VP -> VP NP PP)
{
  UP = SELF-1;
  IF ((MEMBER(obj, SELF-1.ggf))
      &&
      (MEMBER(pobj, SELF-1.ggf)))
  THEN {UP.obj = SELF-2;
        UP.pobj = SELF-3}
  ELSE {IF (MEMBER(obj, SELF-1.ggf))
        THEN {UP.obj = SELF-2};
        IF (MEMBER(pobj, SELF-2.ggf))
        THEN {UP.obj, pobj = SELF-3}
        ELSE {UP.obj.postmod = SELF-3}};
        IF (MEMBER(subj, SELF-1.ggf))
        THEN {UP.subj = SELF-2;
              UP.subj.postmod = SELF-3};
  COMPLETENESS (GF)
}
```

Las entradas léxicas incluyen la información funcional.

```
(CAT:v, LU:activar,
  pred:activar,
  ggf:lobj],
)
```

4.2.2. Módulo de Transferencia.

Sólo los elementos que aportan información relevante unifican en el módulo de análisis. En nuestro ejemplo, estos elementos son el verbo principal ("activar") y el núcleo del NP ("modo_no_molesten"). Siguiendo las reglas que se han definido en este módulo:

```
(activar => On)
(modo_no_molesten => FuncionNoMolesten)
```

podemos definir la función telefónica correspondiente que pasa al módulo de generación.

El módulo de transferencia de LEKTA es el lugar donde se resuelve el problema de las selecciones de restricción. Así, el verbo "avisar" es analizado de manera muy diferente según el contexto en que se encuentre: si va acompañado de objeto o de objeto

preposicional o, incluso, si el nombre que funciona como objeto es un término léxico u otro, como se puede ver a continuación:

```
(avisar=> Off WHEN (polar:neg, ggf:[pobj], pobj:(pcase:de), pobj:
(pred:llamada_en_espera))
          DO (polar:NOTTRANSFER())
Off WHEN(polar:neg, ggf:[pobj], pobj:(pcase:de),
pobj:(pred:llegada))
          DO (polar:NOTTRANSFER())
Off WHEN(polar:neg, ggf:[pobj], pobj:(pcase:de),
pobj:(pred:conversacion))
          DO (polar:NOTTRANSFER())
Off WHEN(polar:neg, ggf:[pobj], pobj:(pcase:de),
pobj:(pred:recepcion))
          DO (polar:NOTTRANSFER())
Off WHEN(polar:neg, ggf:[pobj], pobj:(pcase:de),
pobj:(pred:llamada_entrante))
          DO (polar:NOTTRANSFER())
Off WHEN(polar:neg, ggf:[pobj], pobj:(pcase:de),
pobj:(pred:esperando_llamada))
          DO (polar:NOTTRANSFER())
Off_EsperandoLlamada WHEN(polar:neg, ggf:[pobj],
pobj:(pred:llamada,pcase:de))
          DO (pobj:NOTTRANSFER(), polar:NOTTRANSFER())
EsperandoLlamada WHEN (ggf:[pobj], pobj:(pred:llamada))
          DO (pobj:NOTTRANSFER())
NULL WHEN (ggf:[pobj], pobj:(pred:llamada_en_espera))
Reintento WHEN (ggf:[pobj], pcase:de)
Off_Reintento WHEN (ggf:[0], polar:neg)
Reintento WHEN (ggf:[0]))
```

4.3. Ventajas que Ofrece LEKTA sobre el Analizador Semántico de ATOS.

LEKTA ofrece la ventaja de que fue diseñado como herramienta para el desarrollo de sistemas de TA. Esto es, permite al usuario crear sus propias gramáticas, léxicos, módulos de transferencia, etc., según los requisitos de la aplicación que pretenda desarrollar. Para comprobar que el sistema era eficiente en el contexto de ATOS, se pasaron 1.500 oraciones diferentes referidas a funciones ofrecidas por el terminal. Este corpus recogía un total de 24 funciones telefónicas. El número total de oraciones correctamente analizadas fue de un 84,25%. Es decir, sólo en un 15,75% de los casos no se generaba la función telefónica deseada. El sistema funciona de la siguiente manera (a cada entrada corresponde una función telefónica):

> quisiera anular el modo esperando llamada

@LktrE > (Transl)

(quisiera anular el modo esperando llamada)

====>

(Off EsperandoLlamada)

Algunas oraciones del corpus se referían a entradas que habían sido hechas dentro de un diálogo con la máquina. LEKTA no incorpora información de este tipo, por lo que, si prescindimos de este tipo de oraciones, el grado de acierto en el análisis de oraciones referidas a peticiones telefónicas sube hasta un 93%. Estos resultados prueban la eficiencia del sistema para la tarea de ATOS, más en concreto, como sistema de PLN capaz de analizar oraciones extraídas del habla espontánea.

No obstante, el sistema presenta actualmente ciertas deficiencias derivadas, principalmente, de las características del lenguaje hablado y espontáneo. Por otra parte, se hace necesario incluir en LEKTA mecanismos que puedan tratar oraciones que no han sido correctamente reconocidas por el reconocedor de voz. A continuación se detallan aquellos aspectos que identifican al lenguaje espontáneo y se explica cuál ha sido su tratamiento en el sistema LEKTA.

4.3.1. Características del Lenguaje Espontáneo y su Tratamiento en LEKTA.

El aspecto más conflictivo cuando abordamos la tarea de emplear LEKTA como módulo de PLN en ATOS es que LEKTA tenía que analizar oraciones representativas del habla espontánea. El lenguaje espontáneo presenta unas características muy diferentes al lenguaje "más gramatical" que aparece en textos escritos.

a) En primer lugar, en el lenguaje hablado abundan, por ejemplo, las expresiones coloquiales o idiomáticas. La petición,

> so palma deja de desviar las llamadas

realmente enunciada por un hablante, no representa problemas de análisis en LEKTA. En el caso de las expresiones idiomáticas, éstas no unifican en el módulo de análisis, pues se consideran como no relevantes para el contenido semántico. (Hay que recordar aquí que LEKTA no considera la información pragmática). Así, la regla que define el tratamiento de las expresiones idiomáticas en LEKTA:

```
(X: S -> IDIOM VP)
  {
    UP = SELF-2
  }
```

establece que sólo el segundo constituyente (VP) unifica, por lo que el primero no va más allá en el proceso de análisis.

b) Otro fenómeno en el habla espontánea son las repeticiones debidas a la indecisión, énfasis, etc. En la mayoría de los casos dentro del corpus de ATOS, estas repeticiones se refieren a expresiones coloquiales o idiomáticas:

> bueno bueno ya esta dejalo como estaba

Podemos considerar que estas repeticiones son secuencias idiomáticas recursivas. LEKTA permite la recursividad tanto por la derecha como por la izquierda.

```
(X: IDIOM -> idiom IDIOM)
```

En esta regla encontramos definida la recursividad por la derecha. Pero también se permite recursividad por la izquierda:

```
(X: DET -> DET det)
```

que analiza los determinantes que aparecen en el NP "*este mi teléfono*".

c) Debido a la naturaleza de la tarea, que básicamente se centra en el enunciado de peticiones a un terminal, el sistema no ha de incorporar complejos mecanismos de resolución de ambigüedad (léxica o estructural), anáfora, elipsis, etc. Aunque, como se verá más adelante, estas cuestiones, en especial la anáfora y la elipsis, son problemáticas en el análisis de oraciones dentro de un contexto de diálogo. No obstante, los mecanismos de resolución de los distintos fenómenos lingüísticos que incorpora LEKTA son suficientes para abordar el análisis del corpus de ATOS con éxito. A continuación vemos cómo se resuelve, en la fase de análisis sintáctico, el problema de la ambigüedad estructural cuando un sintagma preposicional (PP) puede modificar a un sintagma nominal (NP) o a un sintagma verbal (VP).

```

(X: VP -> VP NP PP)
{
  UP = SELF-1;
  IF ((MEMBER(obj, SELF-1.ggf))
  &&
    (MEMBER(pobj, SELF-1.ggf)))
  THEN {UP.obj = SELF-2;
        UP.pobj = SELF-3}
  ELSE {IF (MEMBER(obj, SELF-1.ggf))
        THEN {UP.obj = SELF-2};
        IF (MEMBER(pobj, SELF-2.ggf))
        THEN {UP.obj, pobj = SELF-3}
        ELSE {UP.obj.postmod = SELF-3}};
  IF (MEMBER(subj, SELF-1.ggf))
  THEN {UP.subj = SELF-2;
        UP.subj.postmod = SELF-3};
  COMPLETENESS (GF)
}

```

Esta regla especifica que el sintagma preposicional PP modifica al NP cuando el núcleo del VP no requiere un objeto preposicional (pobj).

5. Limitaciones Actuales y Futuros Desarrollos.

Los resultados obtenidos a partir de un primer análisis de las oraciones del corpus demostraban la posibilidad de usar LEKTA como sistema de PLN en ATOS. No obstante, este análisis se hizo sobre oraciones gramaticalmente correctas. Sin embargo, el sistema de PLN en ATOS actúa sobre oraciones que, en ocasiones, no son las realmente pronunciadas por el usuario, sino las reconocidas por el reconocedor de voz. Como se dijo anteriormente, se hace necesario incluir en LEKTA mecanismos que puedan tratar oraciones que no han sido reconocidas correctamente. Por otra parte, el sistema de PLN ha de actuar a la par con el GD. El GD realiza tres tipos de acciones, entre las que se encuentra la lectura de los datos obtenidos desde las órdenes hechas a la PABX. Además, en relación al GD, existen oraciones que en la actualidad no pueden ser analizadas por LEKTA pues se refieren a intercambios en el diálogo usuario-máquina. Son oraciones que han de incorporar información que se ha dicho anteriormente.

5.1. Tratamiento de las Oraciones que no han sido Reconocidas Correctamente.

En primer lugar, de un corpus representativo de oraciones no reconocidas correctamente por el reconocedor, LEKTA tiene un margen de acierto de un 59,67%. Los errores más frecuentes del reconocedor son la falta de concordancia en género, número y persona, así como inserciones y omisiones de palabras. En ocasiones, el sistema reconoce una palabra muy distinta de la que realmente se le dijo.

a) En lo que se refiere a los errores debidos a la falta de concordancia dentro del NP, éstos no impiden el análisis correcto en LEKTA. Veamos el ejemplo siguiente, en el que se genera la función telefónica deseada:

```
> que servicios tiene ACTIVADAS mi terminal
```

```
@LktRe > (Transl)
(que servicios tiene activadas mi terminal)
====>
(Servicios_Activos)
```

Para la tarea ATOS, la información de concordancia (género y número en el nombre, género, número y persona en el verbo) no es relevante, por lo que se desactiva esta función en LEKTA. Sin embargo, la información de concordancia puede especificarse en el módulo de transferencia, de manera que se haga necesaria para transferir un término en función de la concordancia que presente. Un ejemplo de cómo la concordancia determina un análisis concreto lo podemos observar en la siguiente entrada del módulo de transferencia:

```
(anular => Off WHEN (ggf:lobj, pobj])
                DO (pobj:NOTTRANSFER())
Off_Reintento WHEN (ggf:lobj], obj: (pred: llamada,
agr: (num:sing))
                DO (obj:NOTTRANSFER()))
```

para el caso "anula la llamada" que ha de generar la función telefónica Off_Reintento, frente a la petición "anula todas mis llamadas", que genera la función telefónica on FunciónNoMolesten.

b) En cuanto a los errores debidos a la falta de concordancia entre el VP y otros elementos oracionales, hay que tener en cuenta que sólo la información que incluye el "pred:" pasa al módulo de transferencia, por lo que el ejemplo siguiente tampoco representa problemas de análisis en LEKTA:

```
> quiero ACTIVA el modo esperando llamada
@LktRe > (Transl)
(quiero activa el modo esperando llamada)
====>
(On EsperandoLlamada)
```

c) Los casos más conflictivos son aquellos en los que el reconocedor elide un término que realmente fue dicho (infra-reconocimiento) o cuando incluye un término que no fue dicho por el hablante (sobre-reconocimiento). Sin embargo, LEKTA consigue resultados satisfactorios en casos en los que se elide una preposición:

```
> quiero saber cual es la extensión ** Santiago
@LktRe > (Transl)
(quiero saber cual es la extension Santiago)
====>
(ConsultaAgenda Santiago)
```

No obstante, existen otras limitaciones, en oraciones del tipo:

```
REF: hacer la llamada a tres.
HYP: hacer la llamada ***
```

donde encontramos un caso de infra-reconocimiento, o en:

```
REF: uno dos tres cuatro cinco
HYP: uno dos tres cuatro O cinco
```

caso típico de sobre-reconocimiento.

El número total de entradas, dentro del corpus de oraciones con errores de reconocimiento, que presentan los fenómenos de infra-reconocimiento y sobre-reconocimiento es del 18,95%. Para este tipo de errores se prevé tener solución con nuevos mecanismos que se están desarrollando actualmente. Si los resultados del análisis, una vez demostrada la eficiencia de estos mecanismos, son correctos, la eficacia general del sistema en el análisis de sentencias incorrectamente reconocidas subiría de 59,67 a 78,62%. Parece que el camino a seguir es la incorporación de estos

mecanismos de detección de errores que permitirían, para los casos de sobre-reconocimiento, elidir un fragmento reconocido si LEKTA puede llegar al análisis correcto prescindiendo del elemento intruso. Asimismo, para los casos de infra-reconocimiento, se está implementando en LEKTA el mecanismo de inclusión de "*categoría desconocida*", que ocuparía el lugar que habría de tener el elemento realmente pronunciado por el hablante. El objetivo general es diseñar mecanismos sintácticos, léxicos, funcionales y estadísticos para conseguir el análisis correcto de las oraciones generadas por el reconocedor del habla.

5.2. Consulta de Base de Datos.

En segundo lugar, LEKTA es capaz de encontrar la lectura de los datos obtenidos desde las órdenes hechas a la PABX. Es decir, puede analizar oraciones del tipo:

```
> incluye a Santiago en mi agenda
@LktRe > (Transl)
(incluye a Santiago en mi agenda)
====>
(IncluirAgenda Santiago)
```

aunque no se prevé que el sistema tenga que ser el encargado de gestionar y almacenar entradas en una base de datos.

5.3. Gestión de Diálogo

Por último, es preciso incorporar mecanismos en LEKTA que tengan en cuenta el contexto del diálogo, que se gestiona desde el GD. Los ejemplos del tipo presentado en la figura 2 hacen necesaria la creación de herramientas que solucionen los problemas de referencia intratextual: anáfora, elipsis, etc.

6. Conclusión.

De los resultados aquí presentados, podemos llegar a la conclusión de que LEKTA es una alternativa válida al analizador semántico que existe en la actualidad en ATOS. En primer lugar, hemos demostrado que el sistema es eficiente en el análisis de entradas representativas del habla espontánea, que son:

- ♦ abundancia de expresiones coloquiales e idiomáticas,
- ♦ repeticiones y
- ♦ ambigüedad estructural con PP.

Por otra parte, LEKTA también obtiene resultados positivos en el análisis de oraciones realmente reconocidas en los casos de:

- ♦ falta de concordancia en el NP,
- ♦ falta de concordancia del VP con otros elementos oracionales y
- ♦ elisión de la preposición.

Además, el sistema es capaz de encontrar la lectura de los datos obtenidos desde las órdenes hechas a la PABX.

Una ventaja adicional de LEKTA es que puede considerarse como módulo de PLN aplicable a cualquier otra tarea por los motivos siguientes:

a) En primer lugar, LEKTA fue diseñada como una plataforma plurilingüe. Es decir, LEKTA admite la traducción entre un número indefinido de lenguas simplemente especificando la lengua origen y destino. Esta característica es definitiva a la hora de integrar LEKTA a un sistema global de PLN con entrada por voz. El diseño general de LEKTA no es consecuencia de la tarea a la que se aplique, como ocurre en el actual analizador semántico de ATOS. Así, LEKTA podría utilizarse como módulo de PLN cuando la lengua es el castellano o cualquier otra lengua. Asimismo, el sistema puede devolver la traducción a otra lengua, o a una serie de tareas telefónicas. Es decir, se prueba que LEKTA es compatible con múltiples tareas.

b) En segundo lugar, LEKTA es un sistema basado en transferencia. Es en este módulo donde se definen los contenidos semánticos. Precisamente, al consistir en un sistema plurilingüe, sólo sería necesario cambiar el módulo de transferencia para adaptar el sistema a las distintas tareas. En ATOS, el módulo de transferencia actúa como analizador semántico: para cada entrada relevante desde el punto de vista semántico, LEKTA asigna una función telefónica. En caso de que la tarea cambiara, sólo se tendrían que crear otras gramáticas, léxicos, módulos de transferencia, etc., según los requisitos de la aplicación que se pretenda desarrollar.

7. BIBLIOGRAFÍA.

- [1] ÁLVAREZ, J., C. Crespo, G. Escalada, D. Tapias y M.T. López-Soto. 1996. "Modelado del Lenguaje y Gestor de Diálogo en un Sistema de Operador Asistido por Voz". En *Procesamiento del Lenguaje Natural*, 19:179-88.
- [2] ÁLVAREZ-CERCADILLO, J., J. Caminero-Gil, C. Crespo-Casas y D. Tapias-Merino. 1996. "The Natural Language Processing Module for a Voice Assisted Operator at Telefónica I+D." en *JCSLP*.
- [3] AMORES, G. 1996 "LEKTAII: Fase de análisis de un prototipo de traducción automática con entrada y salida por voz." En *Estudios Computacionales del Español y el Inglés*. p. 199-225.
- [4] AMORES CARREDANO de J.G., J.F. Quesada Moreno y D. Tapias Merino. 1994. "Traducción Automática basada en el formalismo LFG con entrada y salida por voz". En *Comunicaciones de Telefónica I+D*, 5, nº 2:132-47.
- [5] BRESNAN, J. The Mental Representation of Grammatical Relations. Cambridge, Mass: MIT Press.
- [6] QUESADA, J.F. 1996. "LektaII: Toward Real-Time Machine Translation. Applications in the Banking Domain". En Sierra, G. y E. Bonsón. *Intelligent Systems in Accounting and Finance*, 77-102.

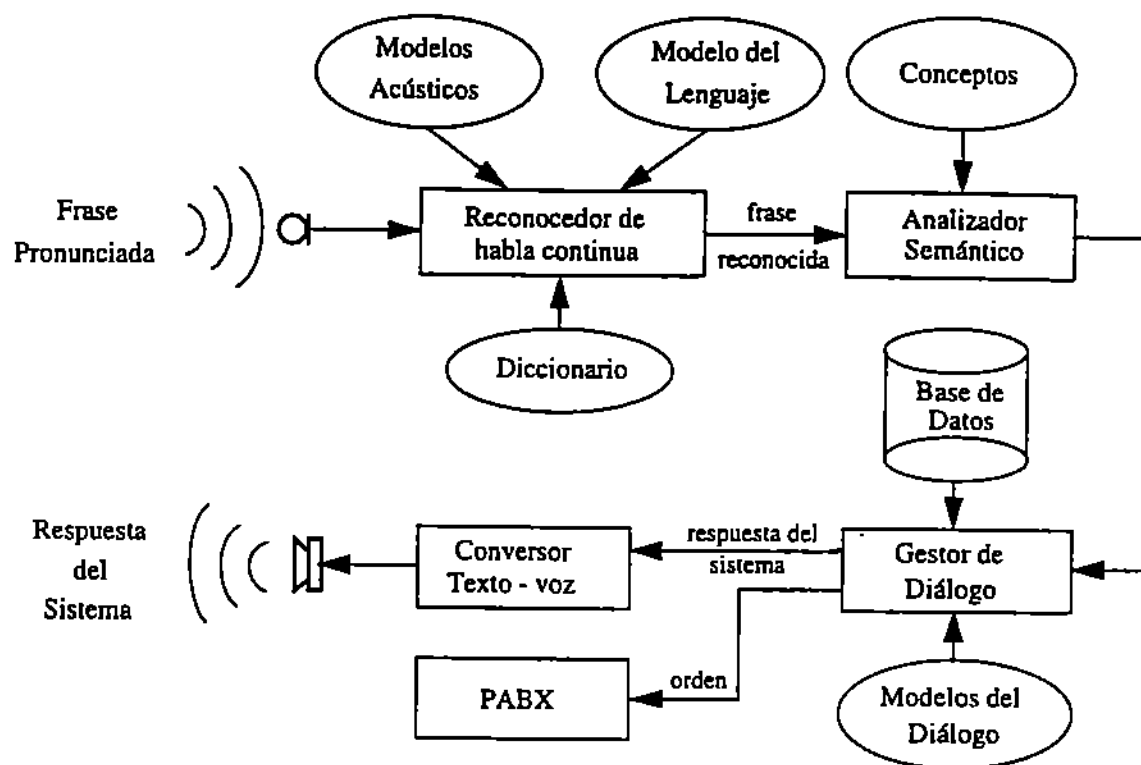


Figura 1: Esquema general del sistema conversacional ATOS

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

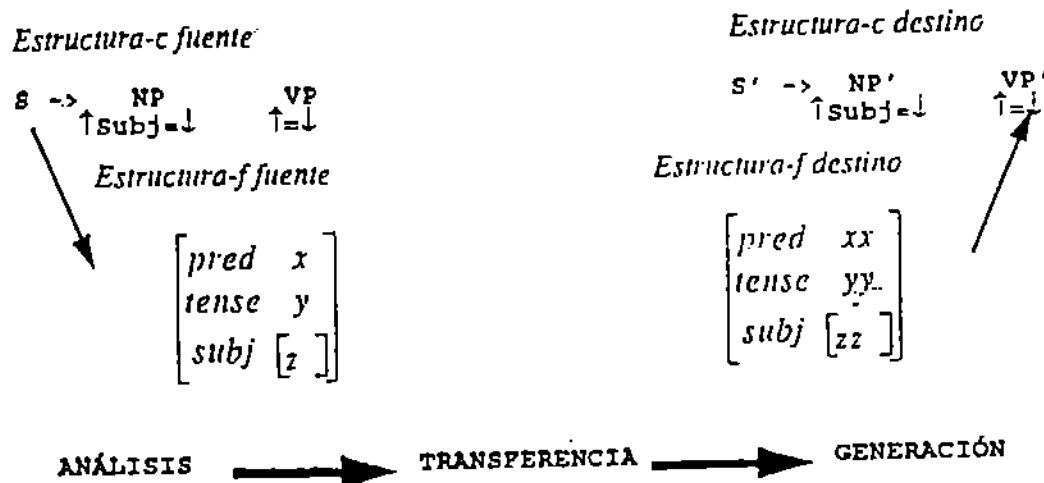


Figura 3: Traducción Automática con enfoque de Transfer basada en LFG

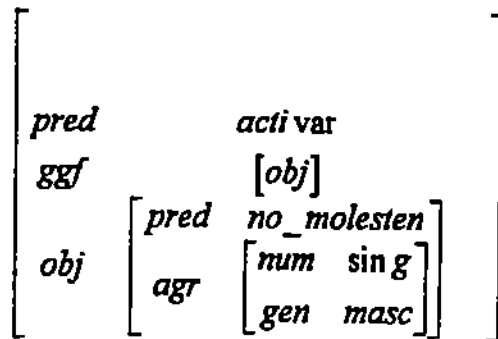


Figura 4: Ejemplo de estructura funcional en LEKTA

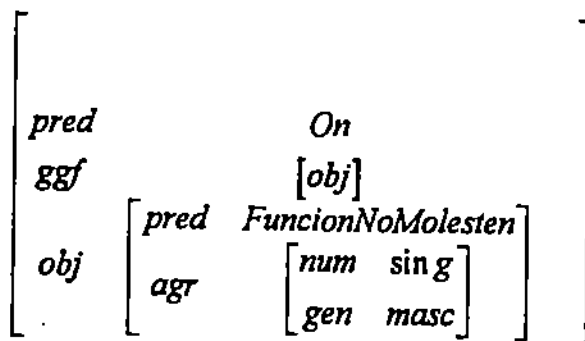


Figura 5: Ejemplo de estructura funcional en LEKTA.