Capítulo 2. *El CU Communicator y el sistema Phoenix*

2.1 Introducción

En abril de 1999, el grupo de procesamiento de voz del **CSLR** (***C****enter for* ***S****poken* ***L****anguage* ***R****eseach*) de la Universidad de Colorado EUA, comenzó el desarrollo del sistema CU Communicator; la implementación de este sistema cumple con los estándares establecidos por la DARPA en cuanto a sistemas de lenguaje hablado [Pellom et al 2000]. El sistema combina reconocimiento de habla continua, entendimiento de Lenguaje Natural y un control flexible del diálogo, para lograr una interacción conversacional más natural vía telefónica, permitiendo a los usuarios obtener información sobre líneas aéreas, hoteles y la renta de autos, en distintas ciudades de EE.UU. y con todo esto, poder elaborar un plan de viaje.

Formando parte del mismo CU Communicator se encuentra el sistema Phoenix [Ward 1991], el cual se encarga de la parte del Entendimiento de Lenguaje Natural, este sistema fue desarrollado originalmente en la Universidad de Carnegie Mellon, y esta diseñado para responder al habla espontánea en un dominio específico. Tomando en cuenta el ruido en la comunicación y las oraciones mal formadas, características propias del habla espontánea, su robustez radica en obtener información relevante, aunque el reconocimiento de una entrada de voz no sea completo, dando como resultado un proceso de parseo más flexible y una mayor capacidad de respuesta del sistema en general.

2.2 Apreciación general del sistema CU Communicator

El sistema CU Communicator esta compuesto por varios servidores y un concentrador o módulo central como se muestra en la figura 2.1, La interacción entre los servidores no puede ser directa, por lo que el módulo central actúa como un ruteador de mensajes, y es por medio de él, que los servidores pueden comunicarse unos con otros. Los servidores se comunican a través de mensajes que tienen un formato definido y el módulo central direcciona estos mensajes de acuerdo a un script basado en reglas.

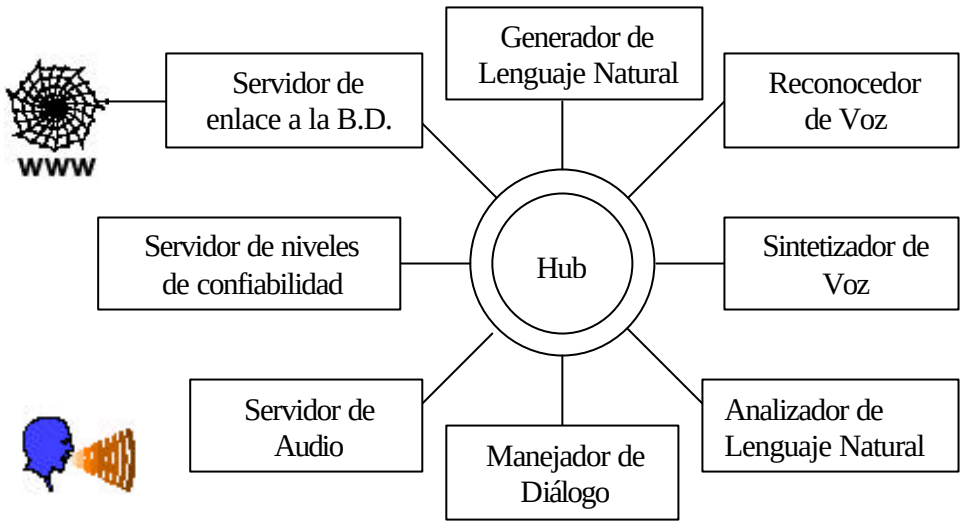

Figura 2.1. Diagrama a bloques de los componentes funcionales que abarca el sistema CU Communicator. Tomado de [*Pellom et al 2001*].

Los servidores que componen el sistema son: (Vease figura 2.1)

1. Audio.- Recibe señales desde el teléfono y las envía al reconocedor, envía también la síntesis de voz al teléfono.

2. Reconocedor automático de voz.- Toma las señales del servidor de audio y produce un arreglo de palabras.

3. Analizador de Lenguaje Natural.- (Parser) Retoma el arreglo de palabras obtenido por el reconocedor y produce para éste la mejor interpretación.

4. Manejador de Diálogo.- Resuelve ambigüedades en la interpretación, estima grados de confiabilidad, hace aclaraciones con el usuario si es necesario, las integra al contexto del diálogo presente, forma las consultas para la base de datos (SQL), envía datos al Generador de Lenguaje Natural para que sean presentados al usuario, solicita información necesaria al usuario.

5. Enlace con la Base de Datos.- Recibe las consultas en SQL del Manejador de Diálogo, las envía a la Base de Datos y obtiene información. La información es tomada del Web.

6. Generador de Lenguaje Natural.- Genera cadenas de texto para ser enviadas al sintetizador.

7. Sintetizador de voz.- Recibe cadenas de palabras del Generador de Lenguaje Natural, las sintetiza y las envía al servidor de audio.

8. Servidor de niveles de confiabilidad.- Este servidor evalúa el grado de confiabilidad de los resultados del proceso de reconocimiento.

9. Perfiles de Usuario.- Este servidor permite a los usuarios introducir las preferencias de su perfil de uso, y este es ocupado por el manejador de diálogo para suplir valores por default sobre las líneas aéreas, hoteles, autos, etc.

2.3 Descripción de los componentes del sistema CU Communicator

2.3.1 Servidor de Audio

El servidor de audio es el responsable de contestar las llamadas telefónicas entrantes al sistema, reproduce mensajes y graba las frases dichas por el usuario. Actualmente el sistema usa el servidor de audio MIT/MITRE que fue provisto por la DARPA, a los participantes en el proyecto de los sistemas de lenguaje hablado. El hardware para el uso de la telefonía consiste de un modem externo de conexión serial, que contiene la entrada para el micrófono y las salidas para los altavoces de la computadora servidor. El proceso de grabación es paralelo al proceso de reconocimiento de voz, y el proceso de reproducción de mensajes es también paralelo al proceso de conversión de texto a voz. Este servidor de audio no soporta interrupciones (barge-in).

Recientemente se esta desarrollando un nuevo servidor de audio que soporte interrupciones, el cual utiliza una plataforma basada en una tarjeta Dialogic, este nuevo servidor implementa un algoritmo del tipo Fast Normalized Least-Mean-Square (LMS) para la cancelación de eco basada en software [Zhang et al. 2001].

Durante la operación el eco es producido por el sistema, este es cancelado dinámicamente, desde la misma señal de audio grabada, para permitir que el usuario pueda interrumpir al sistema mientras esta hablando. El nuevo servidor opera en un ambiente operativo Linux, y esta siendo probado en el mismo laboratorio del CSRL. Debido a que este servidor desarrolla cancelación del eco basada en software, este puede trabajar a un costo relativamente bajo en la plataforma Dialogic. Este servidor podrá estar disponible como recurso a la comunidad investigadora dentro de muy poco.

2.3.2 Reconocedor automático de voz

Actualmente el CU Communicator usa el sistema Sphinx II desarrollado por la Universidad de Carnegie Mellon, para el proceso de reconocimiento. Este es un reconocedor semicontinuo basado en HMM y con un Modelo de Lenguaje basado en Trigramas [Lee 1989]. El servidor de reconocimiento recibe del servidor de audio los vectores de entrada y produce un conjunto de palabras desde las cuales la de mejor hipótesis es escogida para ser enviada al Hub, y luego esta sea procesada por el manejador de diálogo. Próximamente se cambiará un poco la arquitectura de este servidor para permitir que el Parser acceda directamente al conjunto de palabras hechas por este mismo servidor.

2.3.3 Analizador de Lenguaje Natural (Parser)

El CU Communicator utiliza una versión modificada del parser Phoenix, para mapear la salida del reconocedor de voz en una secuencia de frames semánticos. Un frame construido por el sistema Phoenix es un conjunto de slots con nombre, donde estos slots representan partes relacionadas de información. Cada slot tiene asociado una Gramática

Libre de Contexto (CFG), que especifica el patrón de orden de las palabras que pueden contener los slots. Las gramáticas son compiladas en Redes de Transición Recursiva, las cuales son nuevamente cotejadas con el resultado obtenido por el reconocedor para llenar los slots. Cada slot llenado contiene un árbol semántico con el nombre del slot como raíz [Ward 1991].

Phoenix ha sido modificado para producir también una representación extraída del análisis, que es mapeada directamente a frames conceptuales de la tarea presente. Por ejemplo:

"Deseo ir de México a Monterrey el jueves en la mañana"

Lo que produciría el proceso de parsing seria:

Restricción_vuelo: Lugar_salida.Ciudad.México
Restricción_vuelo:Lugar_llegada.Ciudad.Monterrey
Restricción_vuelo:[Tiempo_Fecha].[Fecha].[Dia].jueves
[Tiempo_Intervalo].[Horario].mañana

Más adelante, en la sección 2.4 se mostrará con un mayor detalle la arquitectura y funcionamiento de este sistema.

2.3.4 Manejador de Diálogo (DM, Dialog Management)

El manejador de diálogo controla la interacción entre el usuario y la aplicación servidor, Este módulo es el responsable de decidir que acción siguiente será la que el sistema desarrolle, en cada paso de la interacción. El CSRL de la Universidad de Colorado ha desarrollado un manejador flexible de eventos de diálogo, el cual en el contexto presente del sistema decide que hacer después. Este sistema no usa una red de diálogo o un script de diálogo, mas general a estos es un mecanismo que opera entre las representaciones semánticas y el contexto presente para controlar el flujo de la interacción, propiamente como un "Manejador de eventos".

El manejador de diálogo recibe del analizador de Lenguaje Natural la representación significativa de lo reconocido, esto es integrado al contexto presente. El contexto consiste en un conjunto de frames[1] y un conjunto de variables globales. Cuándo una nueva extracción de información es obtenida, esta es puesta en frames de contexto y algunas veces es usada para actualizar a las variables globales. El sistema provee un conjunto de librerías con rutinas de propósito general para manipular los frames.

La arquitectura del "Manejador de eventos" funciona similarmente a un sistema de produccion. Un análisis entrante a este manejador produce un conjunto de acciones que se activan cuando se modifica el contexto presente. Después de que el análisis ha sido integrado al contexto presente, el Manejador de diálogo examina el contexto para decidir cual es la acción siguiente. El Manejador de diálogo intenta seguir estas acciones en este orden:

- Clarificar si es necesario
- Terminar si todo esta hecho
- Recuperar información y presentarla al usuario
- Dar aviso al usuario sobre información faltante

Las reglas para decidir que hacer son muy concretas. El frame actual es evaluado para determinar si es el que sirve para presentar información al usuario en el momento, o es el último en prioridad para ser tomado en cuenta por el sistema. Las reglas son:

- Si existen slots[2] sin llenar en el frame actual, entonces se busca un valor para el slot con mayor prioridad en el frame.
- Si no hay ningún slot vacío en el frame actual, entonces se busca las piezas de información faltantes en el contexto, de mayor prioridad.

[1] Los frames del Manejador de diálogo son estructuras jerárquicas que representan información de la interacción usuario-sistema.

El mecanismo para manejar la interacción del usuario y sistema no sugiere formas separadas para cada uno de ellos, mas bien si el sistema requiere mayor información entonces procede a obtenerla. El sistema no requiere que el usuario siempre responda. Además el usuario puede responder con cualquier cosa y el sistema intentará analizarlo y relacionarlo con el frame correspondiente, esto permite al usuario dirigir la interacción aunque no es definitivo, los mensajes del sistema para el usuario, están organizados al nivel de frame. El usuario o el sistema pueden enfocar el frame y el sistema intentara llenarlo. Este tipo de representación es fácil de personalizar, ya que no existe una especificación por separado del control de diálogo que sea requerida. Este mecanismo además es robusto por que posee un control simple que no pierde la pista del estado actual en que se encuentra.

2.3.5 Servidor de Enlace con la Base de Datos

Este servidor consiste de una base de datos en SQL y un conjunto de scripts en Perl de dominio específico, para acceder a información desde paginas en Internet. Durante esta operación las consultas son enviadas desde el Manejador de diálogo a este servidor con un formato específico. La información procesada por este servidor consiste de una parte dinámica y una estática, la parte estática consiste de tablas que contienen datos que no son actualizadas constantemente, por ejemplo nombres de ciudades, estados, y localización de aeropuertos, la dinámica consiste en tablas de base de datos que contienen información sobre autos, hoteles y líneas aéreas, que requieren ser constantemente actualizadas para que la aplicación provea datos vigentes.

Cuando una consulta es recibida del Manejador de diálogo, un comando de SQL es ocupado para seleccionar todos aquellos registros que cumplen con el criterio de búsqueda, en la memoria local, si no existe ningún registro, el servidor envía una consulta basada en http para el Internet, los registros obtenidos son insertados como filas nuevas en las Bases de datos locales y el proceso de búsqueda original, vuelve a efectuarse.

---

[2] Los slots de un frame son conjuntos de palabras que tienen un nombre para identificarlos, y las palabras pueden tener diferentes patrones de orden, sin embargo se refieren al mismo concepto.

2.3.6 Generador de Lenguaje Natural

El módulo generador de lenguaje natural usa templates[3] para construir texto, a partir de eventos presentes en un diálogo hablado. Ejemplos de estos eventos son: "Mensaje" para solicitar información al usuario, "Resumir" para concentrar el informe de los vuelos, habitaciones, modelo de autos, y "Clarificar" para resolver la ambigüedad en palabras que comparten el mismo nombre.

2.3.7 Sintetizador de voz

Para producir la salida de audio, se utiliza un sintetizador de voz concatenativo[4], dependiente de dominio. Este sintetizador une desde fonemas, palabras, hasta oraciones completas para que oraciones que contengan información puedan ser reproducidas por el servidor de audio, un algoritmo. Para el modelado del domino se ocupa un locutor con buena dicción que grabó frases hechas dependiendo de la aplicación del sistema. Cada frase es ortográficamente transcrita y fonéticamente alineada usando un reconocedor basado en HMM.

Para la selección de unidades a concatenar se utiliza un algoritmo de búsqueda híbrido, que opera al nivel de palabra o fonema. Durante la síntesis de voz secciones de texto al nivel de palabra que ya han sido grabadas anteriormente, son automáticamente concatenadas. Las palabras o secuencias de palabras que no han sido concatenadas son sintetizadas usando una búsqueda Viterbi beam, a través de la cual se buscan todas las posibles unidades fonéticas disponibles. Durante la síntesis de voz el texto es automáticamente dividido en oraciones individuales y luego entonces sintetizadas y paralelamente reproducidas en el servidor de audio.

---

[3] *Templates*.- Formatos preestablecidos para la generación de oraciones de acuerdo a ciertos parámetros proporcionados por el usuario.

[4] Un sintetizador concatenativo utiliza una colección de intervalos de habla grabados previamente, para unirlos y generar a partir de un texto voz, están subdivididas de un modo tal que permitan generar lo más cercano posible a un sonido natural. Estas subdivisiones abarcan desde fonemas elementales, hasta frases completas de un lenguaje, y son estas ultimas las que en primer orden de prioridad se buscan para dar un mejor énfasis en la generación de un lenguaje fluido.

2.3.8 Servidor de niveles de confiabilidad

Anteriormente el trabajo hecho en la estimación del nivel de confiabilidad, se basaba en la detección y el rechazo de los errores de reconocimiento a nivel palabra y en las frases fuera de dominio a través del uso de características de modelo de lenguaje [Pellom et al 2000]. Recientemente se considera la detección y rechazo de errores de reconocimiento a nivel concepto. Esto es por que los conceptos son usados para actualizar el estado del diálogo en el sistema, los desarrolladores de este sistema piensan que una confiabilidad al nivel de concepto es de vital importancia, para asegurar una interacción humano-computadora elegante. El trabajo actual de investigación en la detección de errores considera características de lenguaje y características acústicas. Las características de confiabilidad son combinadas para calcular los grados de confiabilidad a nivel palabra, oración y concepto.

2.3.9 Servidor del perfil de usuario

Una página de Internet es utilizada como enlace para establecer el perfil de usuario, para que sea empleado por el manejador de diálogo. Esta página permite a los usuarios personalizar la funcionalidad de la interactividad del diálogo, tanto como sea provisto de recursos para ordenar las opciones de líneas aéreas, hoteles y la renta de autos, recuperadas desde el servidor de enlace con la Base de Datos.

Específicamente, el CU Communicator permite a los usuarios introducir información general sobre su nombre, dirección de correo electrónico, y numero telefónico. Para las líneas aéreas los usuarios pueden escoger el asiento, tipo de comida, y la compañía aérea de su preferencia. Los usuarios también pueden configurar el manejador de diálogo para recobrar información de los vuelos más económicos, las líneas con mejor puntualidad. Estas características son similares para el establecimiento de las preferencias de la renta de habitaciones de hotel y autos.

2.4 El sistema Phoenix

El entendimiento del habla espontánea presenta algunos problemas que no encontramos en la pronunciación de un texto escrito previamente. Este contiene tartamudeos, pausas, reinicios, repeticiones, interjecciones, etc. Usuarios fortuitos no conocen el vocabulario y la gramática empleados por el sistema. Por esto, es muy difícil entender el habla espontánea. Wayne Ward, investigador del CSRL, argumenta: "No hablamos como escribimos", y en los sistemas conversacionales es aún más difícil alcanzar una buena cobertura del vocabulario y la gramática que los usuarios pueden emplear [Ward 2001].

Como ya hemos mencionado anteriormente, en el uso de estos sistemas el objetivo no solo es obtener un buen entendimiento de ambas partes, sino que el sistema deberá desempeñar una tarea o actividad, basándose en lo dicho por el usuario. En este momento es necesario agregar que un error de entendimiento al nivel de palabra no es tan trascendente como un error de entendimiento al nivel de oración. En la Universidad de Carnegie Mellon se desarrolló el sistema ***Phoenix***, para el entendimiento del habla espontánea el cual fue inicialmente implementado para una aplicación de servicio de información de transporte aéreo (***ATIS***, ***A****ir* ***T****ravel* ***I****nformation* ***S****ystem*), aquí presentaremos su diseño y la forma de operación.

2.4.1 Estructura del sistema Phoenix

El sistema Phoenix es más que sólo una máquina de estado finito, esta descrito como un chart parser[5] restringido debido a que su definición de lenguaje permite gramáticas regulares [Ward 1991].

[5] Un chart parser recibe ese nombre debido a que durante el proceso de anális , llena un arreglo llamado chart, que tiene N entradas. Para cada posición de las palabras asigna un numero y en una oración el chart contiene una lista de estados que representan los arboles de análisis semanticos parciales generados hasta cierto momento. Cuando se ha llegado al final del análisis de una oración se pueden tener varios arboles que no se repiten y pueden usarse para generar una representación total.

Phoenix es equivalente al parser GRL de Tomita [Ward 1991]. Al igual que el parser GRL, en Phoenix la ciclicidad, la densidad o la ambigüedad infinita no son posibles [Rosas et al 1999]. Este parser acepta transcripciones de un reconocedor de voz y realiza el proceso de parsing de estas entradas. En la figura 2.2 se muestra la estructura del sistema completo.

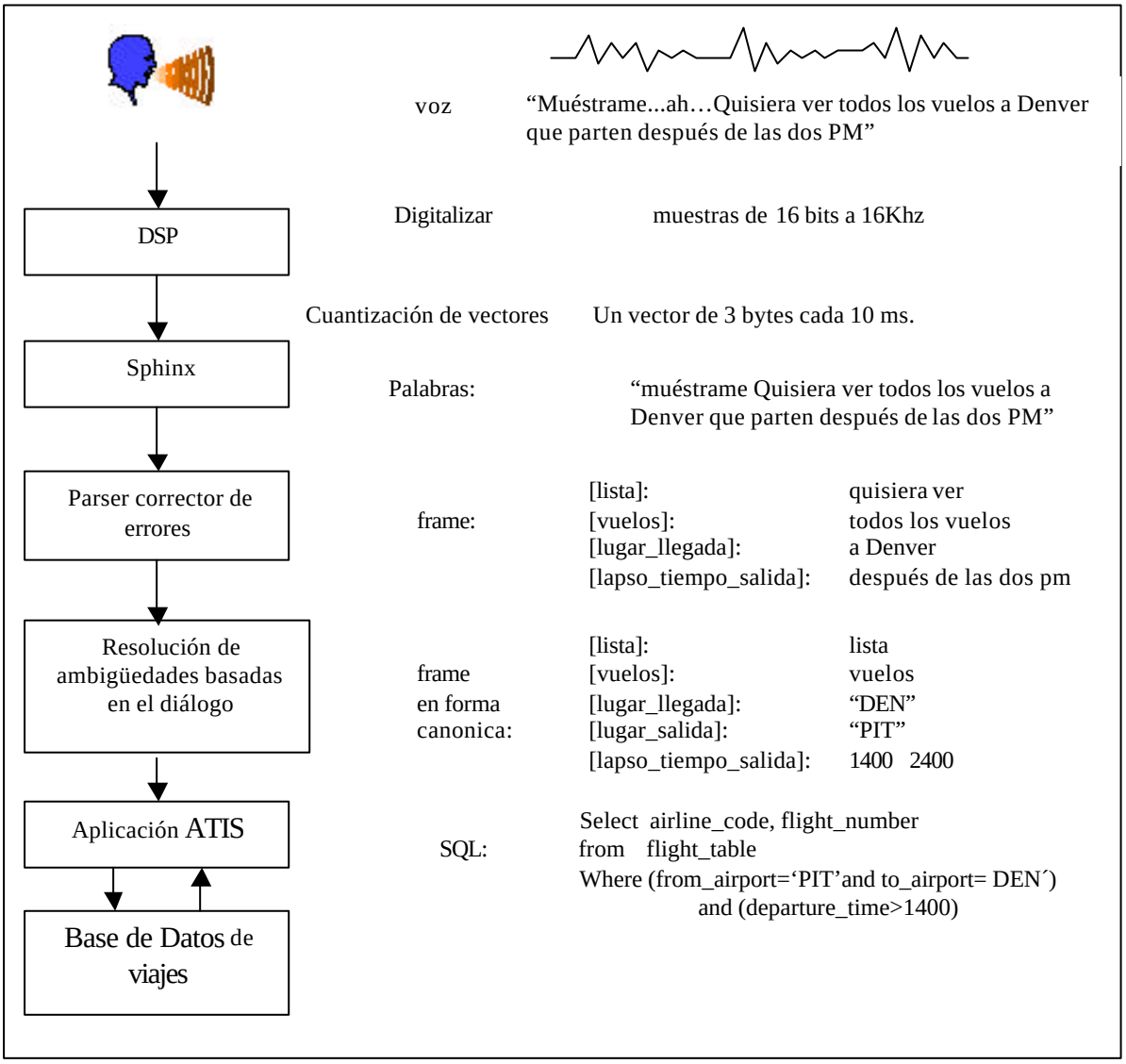

Figura 2.2: Estructura del sistema Phoenix. Tomado de [Ward 1991]

2.4.2 Modo de operación del sistema Phoenix

Algunos problemas se presentan en el habla espontánea, por ejemplo:

- *Ruido del usuario*.- respiración, pausas y otros ruidos generados por el usuario
- *Ruido del ambiente*.- Portazos, timbre del teléfono, etc.
- *Cobertura gramatical*.- Los usuarios a menudo usan oraciones mal formadas, reinician y repiten frases

Phoenix intenta solucionar estos problemas a través del uso de modelos de sonido no verbales, modelos de palabras fuera de vocabulario y un parseo flexible. Estos modelos se explicaran brevemente a continuación:

2.4.3 Modelos de sonidos no verbales

En esta técnica se agregan modelos al sistema que representan sonidos no verbales, como si fueran modelos de palabras que representan sonidos verbales. Estos modelos son entrenados como si fueran modelos de palabras, pero usando la entrada del ruido. De esta manera, los sonidos que no son palabras son mapeados en tokens que no representan palabras.

2.4.4 Modelos de palabras fuera del vocabulario

Para poder manejar a las palabras fuera del vocabulario se crea un modelo explícito para este tipo de palabras. Este modelo se basa en trifonemas[6] (fonema dependiente del contexto), donde un trifonema sigue a otro trifonema (y sobrentendiendo que el contexto sea el mismo) se aplica entonces un modelo de probabilidad de bigramas(Ver sección 3.2.1.2). Los bigramas son entrenados a partir de un gran diccionario de pronunciaciones en Inglés.

[6] Trifonema en el modelado fonético este modelo permite determinar la identidad del fonema dependiendo de los fonemas que aparecen en ambos lados de donde se encuentra.

2.4.5 Parseo flexible

En el concepto de parseo flexible combina una semántica basada en frames con una gramática semántica de oraciones, para permitir al usuario generar entradas al sistema mas complejas, manejando oraciones que puedan o no contenerse en la gramática preestalecida, sin que estas sean rechazadas totalmente. La información semántica es representada en un conjunto de frames. Un frame es una estructura con nombre que contiene un conjunto de slots y estos a su vez son conjuntos de palabras que representan partes de información. En el orden en que se llenan estos slots, se usa una gramática semántica de frases particionada. Cada slot a su vez esta representado por una red de estados finitos la cual especifica todos los modos que en que pueden ordenarse las palabras y que representen algo con sentido.

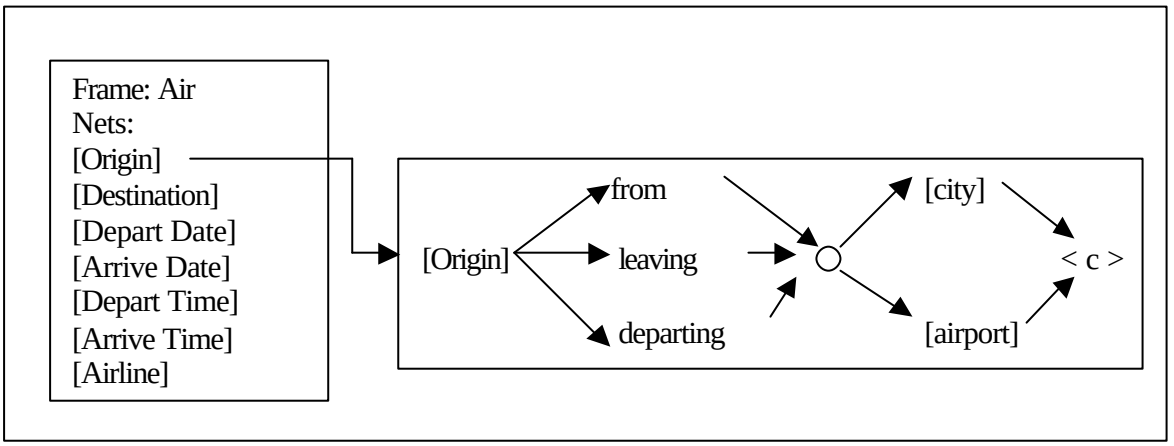

Figura 2.3: Ejemplo de un frame construido por el sistema Phoenix. Tomado del CAT Parser Manual (Ver Apéndice D).

La gramática empleada es una gramática semántica, los nodos no terminales son conceptos semánticos en lugar de partes de voz. Las oraciones que no forman una buena representación gramatical en Inglés permanecen para ser analizadas por el sistema. La gramática es compilada también en redes de transición de estados finitos. Es particionada en el sentido que en una gran red, se contienen varias redes pequeñas. Las redes pueden llamar a otras redes, y por lo tanto se reduce significativamente el tamaño total del sistema. El proceso de parsing puede verse como una "detección de frases". Varias interpretaciones son perseguidas simultáneamente. Una interpretación es un frame con algunos de sus slots llenos.

Cuando una oración es reconocida, se intenta extender a todas las posibles interpretaciones, esto quiere decir que se intenta asignar a slots en los que pueda caer, los slots con la misma interpretación la de menor puntuación es descartada. La puntuación para una interpretación esta dada por el numero de palabras de entrada que toma en cuenta para dar un sentido correcto.

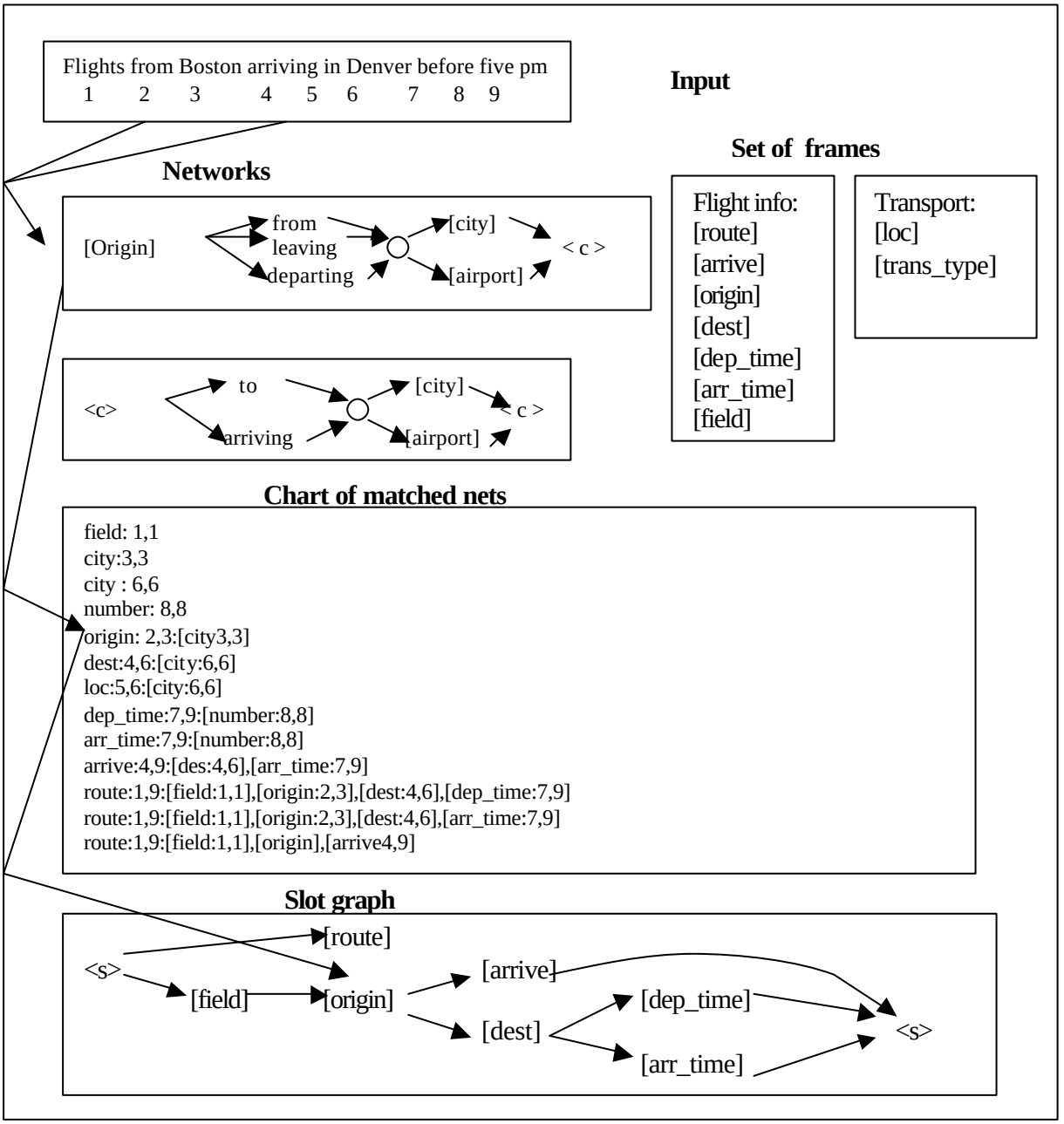

Figura 2.4: El proceso de parsing. Tomado del CAT Parser Manual (Ver Apéndice B).

Aplicar restricciones al nivel de frase es más flexible que reconocer oraciones como un conjunto donde se dan mas restricciones que en la "detección de palabras clave". Los reinicios y repeticiones a menudo se encuentran entre oraciones, pero las oraciones tomadas individualmente son reconocidas correctamente. Gramáticas construidas pobremente a menudo consisten de frases bien formadas, y también a menudo son semánticamente correctas, lo incorrecto únicamente es su sintaxis [Ward 1991].

2.4.6 Resultados de un proceso de parsing con Phoenix

Los slots con mejor puntuación son usados para construir objetos. Aquí toda la información es estructurada de tal forma que sirva para construir consultas a la B.D. Los objetos representan la información extraída de lo que dijo el usuario. Hasta el momento se conforma un conjunto de objetos que representan las condiciones de intervenciones hechas anteriormente. Los nuevos objetos son agregados a este conjunto, en este paso la elipsis[7] y la anáfora[3] están resueltos. La resolución de los fenómenos de elipsis y anáfora es fácil relativamente en este sistema. Los slots en los frames son semánticos por lo cual se sabe el tipo de objeto necesario para la resolución de los fenómenos. Para la elipsis se agregan nuevos objetos, para la anáfora simplemente se tiene que verificar si un objeto de determinado tipo existe. Cada frame tiene asociada una función. Después de que se extrajo la información de lo dicho por el usuario y que se han construido los objetos correspondientes, se ejecuta la función del frame, a menudo es para recuperar información de la Base de Datos a través de una consulta en determinado lenguaje, usado por el manejador de la Base de Datos.

[7] La elipsis es un fenómeno del lenguaje hablado que se caracteriza por la deliberada omisión de una o varias palabras que sin embargo se sobreentienden de acuerdo al contexto gramatical.

[8] La anáfora se refiere a la repetición de palabras al inicio de oraciones o versos

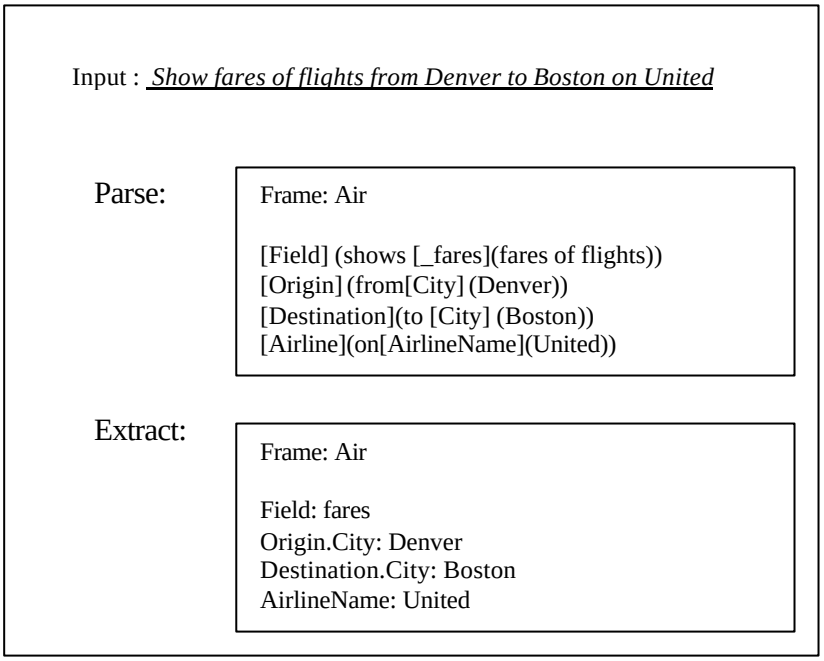

Figura 2.5: Resultado de un proceso de parsing en Phoenix. Tomado del Manual del CAT Parser (Ver Apéndice B).

El trabajo realizado por el CSLR, esta enfocado a construir sistemas productivos y altamente eficaces. Se están construyendo sistemas de aplicación específica, que integran nuevos algoritmos de aplicación altamente funcional. El resultado de toda esta actividad investigadora es la de proveer plataformas de desarrollo (toolkits), que posean gran portabilidad, y que sirvan para desarrollar una nueva generación de interfaces que soporten diversos medios de acceso. En el caso particular de este trabajo de tesis es comprender la forma en que opera la herramienta desarrollada por este grupo de trabajo el CU Communicator y de su posible aplicación al idioma Español de México en un dominio de aplicación definido (altas académicas). En el siguiente capítulo se mostrará un enfoque para lograr una comprensión del lenguaje basado en métodos estadísticos, este último enfoque ha dado lugar a toda una serie de esfuerzos, para lograr un mayor acercamiento a la comprensión y modelado, de cómo los humanos entendemos el lenguaje hablado.