2.1 Introducción

Los importantes avances en las tecnologías del habla están permitiendo el desarrollo de sistemas capaces de operar en condiciones reales (Zue, 1997a). En este punto, los SVIs se han revelado como la mayor fuente de aplicación de estas tecnologías. En la última década se han desarrollado gran cantidad de estos servicios con funcionalidad muy variada: sistemas de información como JÚPITER (Zue et al, 1997b; Zue et al, 2000) que proporciona información meteorológica, STACC y TADE sistemas que ofrecen las calificaciones de los alumnos y que han sido desarrollados en las Universidades de Granada y la Politécnica de Madrid respectivamente (Rubio et al, 1997 y Casas, 1997). Existe también gran número de servicios de información y reserva de billetes de viaje. En esta línea cabe comentar los servicios de información de horarios de trenes desarrollados en los proyectos europeos RailTel (Lamel et al, 1997) y su continuación ARISE (Lamel et al, 2000; Baggia et al, 2000) y el servicio de información de horarios de avión, reserva de hotel y alquiler de coches desarrollado en Estados Unidos dentro del proyecto DARPA Communicator (http://fofoca.mitre.org), (Ward y Pellom, 1999; Pellom et al, 2000; Rudnicky et al, 2000).

Los servicios de información de números de teléfono también han tenido un gran auge como el sistema desarrollado en el proyecto IDAS (San-Segundo et al, 1999; Lehtinen et al, 2000; Córdoba et al, 2001) y el sistema de la empresa Philips (Schrâmm et al, 2000). También, es posible comentar otros tipos de sistemas como un servicio de reserva de cita con el médico (Constantinides et al, 1998) o el sistema de redirección de llamadas de AT&T (How may I help you?) (Riccardi et al, 2000).

Desde 1992, el Grupo de Tecnología del Habla, ha apostado firmemente por la integración de las tecnologías del habla en los sistemas de información a través de la línea telefónica. Con esta orientación se comenzó un ambicioso proyecto cuyo objetivo fue desarrollar un entorno integrado de desarrollo de aplicaciones telefónicas TADE (Telephone Application Development Environment). Este entorno proporciona un nuevo lenguaje, con ciertas primitivas de alto nivel, para el diseño de aplicaciones telefónicas, principalmente SVIs, incluyendo utilidades para cubrir todo el ciclo de vida de una aplicación: diseño, compilación y ejecución (para ver más detalles consultar apéndice A, apartado A.2).

En los siguientes apartados haremos un encuadre científico-tecnológico concretando para cada uno de los temas en los que se ha profundizado en el transcurso de la presente tesis: reconocimiento de nombres deletreados, reconocimiento de habla continua para dominios restringidos (ambos por teléfono), análisis de medidas de confianza y diseño de gestores de diálogo en SVIs.

2.2 Reconocimiento de nombres deletreados

La tarea del deletreo es un problema de reconocimiento de un vocabulario reducido (29 letras) pero con gran confusión entre las palabras que lo componen. En inglés los mayores problemas de discriminación residen en el conjunto E-

set={B,C,D,E,G,P,T,V,Z}. En castellano, además de este conjunto, aparecen otros conjuntos de letras, que al igual que el anterior, dificultan enormemente la tarea de reconocimiento (San-Segundo et al, 2000b). Cuando se trabaja con habla continua una fuente de errores de reconocimiento importante es la coarticulación entre palabras. Este efecto es más peligroso cuando las palabras del vocabulario son pequeñas y con gran parecido entre ellas como es nuestro caso. Otro aspecto a tener en cuenta es la utilización de pronunciaciones alternativas de las letras como por ejemplo la letra LL: elle o doble ele, I: i o i latina (San-Segundo et al, 2000b; San-Segundo et al, 2002). Estas segundas pronunciaciones, aunque en algunos casos sean incorrectas, deben ser modeladas para incrementar la robustez del sistema.

Los primeros sistemas, desarrollados para Inglés, pretendían el reconocimiento de letras de forma aislada con técnicas de comparación de patrones mediante alineamientos dinámicos en varios parámetros de análisis (Cole et al, 1986; Junqua, 1991). En la última década, tanto los modelos ocultos de Markov (HMM) (Brown, 1987; Euler et al, 1990; Junqua, 1997) como las redes neuronales (Cole et al, 1991b; Rogniski, 1991; Hild et al, 1993), adaptadas oportunamente a la tarea, han demostrado funcionar bastante bien. Mientras los sistemas basados en HMMs generalmente se centran en la compartición de estados para mejorar la discriminación entre letras, las redes neuronales se centran en la discriminación de tramas de voz anteriormente segmentadas utilizando técnicas de alineamiento temporal (DTW o HMMs).

En los años 1990s, el problema del reconocimiento de habla continua para nombres deletreados despertó un gran interés (Jouvet et al, 1993a; Kaspar et al, 1995; Loizou et al, 1995). Este hecho coincidió con la disponibilidad de varias bases de datos grabadas y distribuidas por The Center for Speech and Language Understanding (CSLU) en OGI (Oregon Graduate Institute). Este mismo centro hizo uso de este corpus para desarrollar el sistema EAR (Fanty et al, 1990). La investigación fue más adelante extendida al entorno telefónico en el que cabe destacar los trabajos de Cole y Junqua (Cole et al, 1991b; Junqua et al, 1997). Muchas de las soluciones o sistemas implementados hacen uso de estrategias en las que se trabaja con varias secuencias de letras candidatas para posteriormente encontrar el nombre que mejor se adapta a estas secuencias (Jouvet et al, 1993b).

Cuando la cadena de letras deletreada pertenece a un conjunto de cadenas posibles, como por ejemplo un nombre perteneciente a un directorio de nombres propios, las restricciones que este directorio impone, pueden ser utilizadas por el sistema de reconocimiento para mejorar su tasa de acierto. Betz y Hild (Betz y Hild, 1995) expusieron detalladamente las diferentes formas de incorporar estas restricciones en el proceso de decodificación. A continuación las presentamos brevemente:

- Definición de gramáticas N-gram de letras. Con los nombres del directorio se pueden deducir estadísticas de las secuencias de letras más probables y aplicar estas probabilidades en el proceso de decodificación. Algunos ejemplos son los trabajos de Junqua y San-Segundo (Junqua, 1997; San-Segundo et al, 2002).

- Definición de funciones para el cálculo de la distancia entre la secuencia de letras y los nombres del directorio, de forma que nos permita elegir el nombre que mejor se ajuste a una secuencia de letras determinada. Estas distancias deben tener implícito un modelo de error del reconocedor de forma que se consideren diferentes penalizaciones para los casos de inserción, borrado, o substitución de letras. En los primeros trabajos realizados en France Telecom (Jouvet et al, 1993b) podemos ver un ejemplo de la utilización de esta técnica. En este caso, los autores utilizan N secuencias de letras (una óptima y N-1 subóptimas) para conseguir una mayor robustez en la definición de la distancia.

- Definición de un espacio de búsqueda en el que únicamente se permitan las secuencias de letras que corresponden a los nombres del directorio. Es la más restrictiva de las soluciones ofreciendo una mayor tasa a cambio de un mayor coste de procesado (San-Segundo et al, 2000b).

El hecho de disponer de estadísticas sobre los nombres del directorio más frecuentemente deletreados en el servicio, puede hacer que la incorporación de este conocimiento en el proceso de decodificación sea aún más útil para mejorar la tasa de acierto (Pacheco, 1999).

La incorporación de sistemas de reconocimiento de nombres deletreados en SVIs no es una tarea fácil y depende fuertemente de la costumbre de los usuarios a deletrear. En el caso de que los usuarios no estén acostumbrados (como nos ocurre a los castellano-hablantes), una petición de deletreo por parte del sistema, puede ser visto por el usuario como una interacción desagradable que haga aumentar su rechazo hacia el servicio automático. En esta situación, los sistemas de deletreo se deben utilizar como estrategia de reconocimiento en condiciones adversas; ante fallos en reconocimientos anteriores o como último paso antes de redirigir la llamada a un operador humano (Bauer et al, 1999; Córdoba et al, 2001). Otra posible aplicación es la detección de nombres o expresiones fuera del vocabulario (Jouvet et al, 1999; San-Segundo et al, 2001b). En el caso de que el deletreo sea una interacción frecuente, como puede ser en inglés o alemán, se puede utilizar el sistema de deletreo como una interacción previa a la petición de un dato cuyo vocabulario sea muy elevado, por ejemplo el apellido de una persona (> 50.000 palabras). En este caso, la salida del reconocedor de nombres deletreados se utiliza para realizar una selección de palabras sobre el vocabulario, que facilite la tarea de adquisición del dato en una interacción posterior (Seide y Kellner, 1997; Souvignier et al, 2000).

Como se puede deducir de las aplicaciones comentadas en el apartado anterior, la fiabilidad del sistema debe ser muy elevada. Por esta razón se está extendiendo la utilización de arquitecturas de reconocimiento con varias etapas (Mitchell et al, 1999; Junqua, 1997; San-Segundo et al, 2001b; Jouvet y Droguet, 2001), y la utilización de modelos de lenguaje más potentes (Thiele et al, 2000).

Un problema importante de los sistemas de reconocimiento de nombres deletreados es la detección del final de la secuencia de letras. Generalmente las pausas entre letras son bastante grandes lo que repercute en que se detecten finales de voz en la mitad de la

secuencia, produciendo cortes en la cadena de letras. En el trabajo de Hanel y Jouvet (Hanel y Jouvet, 2000) se presenta una solución que consiste en detectar prefijos de los nombres que forman el diccionario, activando la espera del resto del nombre deletreado en esos casos.

En esta tesis se pretende realizar un análisis de la tarea de deletreo para el caso del castellano y a partir de él, desarrollar un sistema de reconocimiento completo. En primer lugar probaremos las técnicas de reconocimiento que mayores éxitos ofrecieron en el caso del inglés (San-Segundo et al, 2000b), para después proponer nuevas técnicas, como es el modelado de silencios con pausas contextuales (San-Segundo et al, 2001b), que permitan ajustar el sistema a las características del castellano. Además, en el sistema desarrollado se propondrá e incorporará una simplificación del algoritmo de Ney (Ney et al, 1994; Ney y Ortmanns, 1999) para la obtención de un grafo de letras a la salida del reconocedor. Este grafo permitirá, en un postprocesado posterior, calcular las N mejores secuencias de letras e incorporar modelos de lenguaje potentes con bajo coste computacional.

2.3 Reconocimiento de habla continua

Muchos de los sistemas de reconocimiento de habla continua basados en HMMs (Rabiner, 1986), hoy en día, comparten una arquitectura similar formada por dos etapas (Ravishankar, 1996; Ney y Ortmanns, 1999; Deshmukh et al, 1999). En la primera etapa se realiza una búsqueda completa sobre los modelos acústicos de todas las palabras del diccionario mediante un algoritmo de programación dinámica (Picone, 1990; Silverman y Morgan, 1990). En esta etapa, el modelado acústico tiene una mayor relevancia que el gramatical, es decir, se aplican modelos de lenguaje de poco alcance para evitar tener que duplicar los modelos acústicos de las palabras y que el coste de procesado se incremente de forma importante. Por otro lado, con el fin de evitar la exploración completa del espacio de búsqueda se suele hacer uso de técnicas de Beam Search (Ney et al, 1992a; Ney, 1992b; Colás, 1999) que permitan recortar el tiempo de exploración sin que este hecho produzca una degradación significativa de la tasa de reconocimiento.

En la segunda etapa de reconocimiento se construye un grafo o lattice de palabras con aquellas secuencias que más parecido acústico presentan con las tramas de voz. Sobre este grafo, mucho más reducido que el espacio total de palabras, se vuelve a realizar otra fase de reconocimiento. Es en esta etapa en la que se aplican modelos gramaticales más potentes, que exigen la duplicación de nodos (3-gram) para definir nodos sin ambigüedad gramatical (Colás, 1999). Debido a que el espacio de búsqueda se ha reducido considerablemente con la generación del grafo, el incremento de tiempo de proceso producido por la necesidad de duplicar nodos es considerablemente menor. El postprocesado sobre el grafo de palabras puede ser aprovechado también, para el cálculo de las N mejores secuencias de palabras con bajo coste en tiempo. Estas secuencias pueden ser utilizadas para hacer más robusta la labor de comprensión en una fase posterior a la de reconocimiento. Estos sistemas descritos han demostrado sus grandes prestaciones con habla limpia, leída o dictada, llegando a trabajar con varias decenas de miles de palabras y tasas bastante elevadas.

En la literatura encontramos principalmente dos opciones para la generación de un grafo o lattice de palabras como resultado de la primera etapa de reconocimiento:

- La primera de ellas es la utilizada en el sistema SPHINX desarrollado por la Universidad Carnegie Mellon (Ravishankar, 1996). Este sistema dispone de la técnica de Beam Search para la reducción del espacio de búsqueda. Pues bien, gracias a esta técnica, es posible determinar a lo largo de la búsqueda, las tramas en las que ha estado activa cada una de las palabras. Estas tramas de voz corresponden con aquellas en las que el último estado del modelo acústico de la palabra ha sobrevivido a la poda que introduce el Beam Search. Este margen de tramas, que son los posibles finales de la palabra, llevan asociados sus respectivas tramas de inicio (tramas en las que se transitó a la palabra considerada), lo que nos da pié a definir otro margen de posibles tramas iniciales. Una vez obtenida esta información para cada una de las palabras que han sobrevivido al proceso de poda, se puede construir fácilmente un grafo de palabras, sin más que definir enlaces entre palabras cuyos márgenes de tramas finales se solapen con los márgenes de tramas iniciales de las palabras siguientes.

- La segunda es la utilizada por Ney y Ortmanns (Ney et al, 1994; Ney y Ortmanns, 1999). En este caso se plantea una modificación del algoritmo de One-pass del que podemos ver una primera formulación en (Ney, 1984). Esta modificación consiste en almacenar durante el proceso de alineamiento dinámico (para cada estado del modelo acústico de cada palabra), diferentes historias dependiendo de la palabra predecesora. De esta forma, a la hora de realizar el backtracking por las historias almacenadas, se puede obtener un árbol inverso que incluye varias hipótesis de reconocimiento en lugar de una única hipótesis. Esta opción, para la generación de un grafo de palabras, incrementa considerablemente el tiempo de proceso, lo que obliga a introducir técnicas de poda o reducción del espacio de búsqueda como el Beam Search. En esta tesis, presentaremos una simplificación de este algoritmo en la que por un lado el número de palabras predecesoras está limitado a un valor fijo, y por otro lado, la variedad de historias se considera únicamente en las transiciones entre palabras, propagándose a lo largo de los estados internos de la palabra.

La utilización de estos reconocedores en Servicios Vocales Interactivos no sólo ha requerido adaptar los modelos acústicos a las nuevas características del canal (red telefónica fija o móvil), sino que ha sido necesario hacer frente a nuevos problemas. En primer lugar, aparece una mayor cantidad y variedad de ruidos que deben ser caracterizados, y por otro lado, el habla adquiere grandes tintes de espontaneidad dado que se produce como respuesta a preguntas que va formulando el sistema. Esta espontaneidad en el habla, da lugar a muchos errores e imprecisiones por parte del hablante y a construcciones gramaticales más relajadas con fuerte coarticulación entre las palabras (Soclof, 1990). Estas nuevas características han obligado a modelar y caracterizar estos fenómenos así como a reducir considerablemente el tamaño del diccionario de reconocimiento para obtener tasas aceptables (Wessel et al, 1999; Pellom et al, 2000; Lamel et al, 2000; San-Segundo, D., 2001). También, ha sido necesario analizar las variantes gramaticales en las construcciones y generar nuevos modelos de lenguaje que aprendan estas variaciones.

Podemos decir por tanto, que para obtener buenas tasas de reconocimiento en Servidores Vocales Interactivos son necesarios reconocedores de habla continua para dominios restringidos. Restringidos en cuanto al vocabulario de reconocimiento y sobretodo a las construcciones gramaticales permitidas. En esta tesis se realizará el desarrollo de un sistema de reconocimiento de habla continua para frases que expresan fechas y horas. Se analizará la generación de grafos o lattices de palabras y el modelado tanto de ruidos como de las variantes gramaticales ocasionadas en el habla espontánea.

2.4 Medidas de confianza

Debido a que el reconocimiento automático del habla dista mucho de ser perfecto, se debe analizar la calidad de lo reconocido/comprendido por el sistema con el fin de detectar posibles errores o zonas de gran ambigüedad. Esta necesidad es aún más importante en Servidores Vocales Interactivos donde una mala interpretación de la frase pronunciada puede llevar al sistema a realizar un comportamiento erróneo. Típicamente en los SVIs existen dos módulos anteriores al módulo de gestión de diálogo: reconocimiento y comprensión. Dichos módulos se encargan de extraer la información semántica de la frase pronunciada por el usuario. Esta información es utilizada por el gestor de diálogo para avanzar en su interacción con él. Las medidas de confianza obtenidas en estos dos módulos tienen como objetivo evaluar su comportamiento de forma que el gestor de diálogo pueda medir la calidad de la información recibida y en consecuencia, elegir la acción concreta a realizar: rechazar la frase, preguntar otra vez, o pedir confirmación de alguno de los datos obtenidos. Por otro lado, la evolución del propio diálogo también puede darnos información que ayude a mejorar su gestión. Los parámetros considerados para obtener estas medidas de confianza se pueden clasificar según su origen en:

- *Parámetros del Decodificador*: son medidas obtenidas de la evolución del reconocedor a lo largo de la etapa de descodificación de la señal de habla. Estas medidas tratan de detectar zonas de voz en las que existe un desacople importante entre los modelos acústicos y la voz pronunciada, o zonas en las que aparecen varias alternativas con gran confusión acústica entre ellas.

- *Parámetros exclusivos del Modelo de Lenguaje:* estos parámetros tratan de validar que la secuencia de palabras obtenida, corresponde con patrones gramaticales característicos, observados en las frases de los usuarios a lo largo de sus interacciones con el sistema.

- *Parámetros de Comprensión:* son medidas obtenidas del analizador semántico y tratan de reflejar la fiabilidad con la que han sido obtenidos los conceptos a partir de la secuencia de palabras reconocidas.

- *Parámetros del Gestor de diálogo:* el punto del diálogo en el que estemos, también puede ayudarnos a evaluar la calidad de los datos obtenidos: por ejemplo si el sistema está solicitando del usuario el nombre de la ciudad origen de un viaje, el hecho de obtener un nombre de un hotel nos debe alertar sobre un cambio de intención por parte del usuario o de la existencia de problemas en el reconocimiento.

Por otro lado la información obtenida durante la propia evolución del diálogo, como el número de interacciones necesarias para conseguir un determinado objetivo o el número de confirmaciones negativas y/o positivas durante la consulta, nos puede dar una idea de cómo se está desarrollando dicha interacción. En el caso de que existan problemas, estos pueden ser debidos a dificultades del usuario para interaccionar, lo que estaría demandando una adaptación mayor del sistema a la destreza del usuario, o pueden deberse a problemas en el reconocimiento por tratarse de un entorno ruidoso. En este caso, el gestor podría seleccionar reconocedores más restringidos pero de mayor robustez.

Según la resolución de las medidas de confianza, podemos clasificarlas en 4 niveles diferentes:

- *Nivel de palabra:* en este caso el objetivo es detectar palabras mal reconocidas. Estos errores de reconocimiento se pueden haber producido por problemas del decodificador, o porque la palabra no está incluida en el vocabulario de reconocimiento (OOV: out of vocabulary). En el caso de un sistema de habla continua se pretenderá la detección de las posibles inserciones y substituciones de palabras.

- *Nivel de concepto:* en este caso se pretende detectar conceptos erróneos dentro de una frase determinada. Las medidas de confianza en este caso son muy importantes para la gestión de diálogo puesto que es la información semántica, la que se utiliza para realizar esta labor de gestión y decidir cuales van a ser las acciones del sistema en su interacción con el usuario.

- *Nivel de frase:* en este nivel, el objetivo es detectar por un lado frases fuera del dominio de la aplicación y por otro, frases del dominio con problemas en el reconocimiento que no tienen ninguna información semántica o concepto correcto. Se pretende por tanto, detectar frases que no van a ser correctamente reconocidas y comprendidas con nuestro sistema, evitando que se detecte algún concepto erróneamente que le haga al gestor realizar una mala interpretación y genere un diálogo divergente con las necesidades del usuario.

- *Nivel de interacción:* el principal objetivo a este nivel es medir la calidad de la interacción. Con estas medidas se pretende detectar situaciones problemáticas como las siguientes: que el diálogo emprenda un camino que diverge de las necesidades del usuario debido a un error de compresión, que la tasa de reconocimiento del sistema esté siendo muy baja por problemas de gran ruido ambiente, o situaciones en las que la respuesta del usuario no se ajusta a las preguntas del sistema por desconocimiento de la funcionalidad y/o las limitaciones del servicio. En estos casos es necesario dotar de mecanismos de corrección ágiles que permitan volver a puntos anteriores del diálogo, disponer de variedad de sistemas de reconocimiento que permitan mayor robustez en ambientes ruidosos aun a costa de perder flexibilidad, y también son necesarias estrategias de modelado de usuario que permitan adaptar las preguntas, informaciones o ayudas del sistema, a la destreza del usuario.

Al final de la década de los 90s, el interés en medidas de confianza ha incrementado considerablemente. El desarrollo de Servidores Vocales Interactivos como los de Ward y Pellom, Zue, y Lamel (Ward y Pellom, 1999; Zue, 1997b; Lamel et al, 2000) ha producido la necesidad de desarrollar mecanismos de análisis de fiabilidad de los sistemas con el fin de conseguir servicios más robustos. Como hemos comentado anteriormente se pueden considerar cuatro niveles de medidas de confianza. Al nivel de palabra, la gran mayoría de los parámetros analizados provienen del proceso de descodificación, considerando tanto modelos acústicos como modelos de lenguaje. En trabajos recientes (Chase, 1997a; Chase, 1997b; Kamppari y Hazen, 2000; Macías-Guarasa et al, 2000b; Hazen et al, 2002) se puede observar un análisis detallado de gran cantidad de parámetros acústicos y del modelo de lenguaje obtenidos en el proceso de descodificación. Por otro lado, Wessel (Wessel et al, 1999; Wessel et al, 2001) ha realizado comparaciones de parámetros obtenidos del grafo de palabras con parámetros obtenidos de las N mejores hipótesis, Moreau y Jouvet (Moreau y Jouvet, 1999) proponen medidas de confianza basadas en relaciones de verosimilitudes, y Bansal junto con Ravishankar, (Bansal y Ravishankar, 1998) analizan dos nuevas medidas: ”Likelihood Dependence (LD)” y “Neighborhood Dependence (ND)”. En MIT, Hazen y Bazzi (Hazen y Bazzi, 2001) proponen la combinación de modelos diferentes para la detección de palabras incorrectas y palabras fuera de vocabulario con el fin de mejorar la detección de errores de reconocimiento en general.

En otra línea de investigación, existe un conjunto de técnicas basadas en la introducción de modelos acústicos de relleno o modelos basura en el proceso de decodificación, para disponer de una referencia con la que comparar. En este caso, las medidas de confianza están basadas en medidas relativas referentes al funcionamiento de estos modelos en el proceso de decodificación (Jouvet et al, 1999a; Gunawardana et al, 1999; Dasmahapatra y Cox, 2000; Moreau et al, 2000; Charlet et al, 2001).

En los trabajos dirigidos por W. Ward (Uhrik y Ward, 1997; San-Segundo et al, 2000b) podemos ver el uso exclusivo de parámetros obtenidos del modelo de lenguaje para obtener medidas de confianza tanto al nivel de palabra como de frase. Pao (Pao, 1998) utiliza parámetros provenientes tanto del decodificador como del sistema de comprensión para la obtención de medidas de confianza al nivel de frase. Trabajos previos en análisis de confianza al nivel de concepto son los realizados por Bouwman, Souvignier, Hazen y San-Segundo (Bouwman et al, 1999; Souvignier et al 2000; Hazen et al, 2000a; Hazen et al, 2000b; San-Segundo et al, 2001a). En los estudios de confianza realizados sobre el sistema CU-Communicator (San-Segundo et al, 2001a; San-Segundo et al, 2001i) podemos ver un análisis detallado para los tres primeros niveles de estudio, y en el que se proponen y analizan gran cantidad de nuevos parámetros provenientes del módulo de comprensión. Zhang y Rudnicky en un trabajo posterior (Zhang y Rudnicky, 2001) amplían el anterior estudio sobre la misma tarea DARPA Communicator, haciendo una comparación de diferentes técnicas de clasificación: Análisis Lineal Discriminante, Redes Neuronales y Árboles de Decisión. Los últimos trabajos realizados en medidas de confianza al nivel de concepto semántico son los dirigidos por W. Ward sobre el sistema CU-Communicator (Hacioglu and Ward, 2002; Sameer and Ward, 2002).

Cuando se trabaja con varios parámetros diferentes, para poder obtener una única media de confianza, es necesario seleccionar los mejores parámetros y combinarlos. Las soluciones más utilizadas para la selección de parámetros son el LDA (Linear Discriminant Analysis)(Hazen et al, 2000a; Hazen et al, 2000b; Hazen y Bazzi, 2001) y los Árboles de Decisión (Breiman et al, 1984; Pao et al, 1998). Las Redes Neuronales junto con los Árboles de Decisión son los mecanismos más utilizados para la combinación de parámetros. Según los resultados presentados por San-Segundo (San-Segundo et al, 2000a), las Redes Neuronales consiguen mejor poder de clasificación que los Árboles de Decisión cuando en los nodos de decisión se utilizan únicamente decisiones estándar, es decir, se considera un único parámetro y un sólo umbral por decisión. En la tesis propuesta en este documento se ha utilizado únicamente Redes Neuronales como mecanismo de combinación y selección de parámetros.

En esta tesis se trabajará principalmente sobre el sistema CU Communicator (Ward y Pellom, 1999; Pellom et al, 2000) desarrollado en la Universidad de Colorado, Boulder. Este sistema es una implementación de la tarea propuesta en el proyecto DARPA Communicator (http://fofoca.mitre.org). El sistema combina reconocimiento de habla continua, comprensión de lenguaje natural y control flexible del flujo del diálogo para permitir una interacción natural a través del teléfono mediante la cual los usuarios pueden acceder a información de viajes de avión, reserva de hoteles y coches de alquiler. El sistema accede por web a información actualizada de forma dinámica. Este sistema utiliza el reconocedor de la Universidad de Carnegie Mellon SPHINX-II. Es un reconocedor basado en modelos HMMs semi-continuos con un modelo de lenguaje 3-gram basado en clases. Para la comprensión de lenguaje natural se usa una nueva versión del sistema Phoenix (Ward, 1994) que realiza la correspondencia entre la secuencia de palabras y la secuencia de conceptos. El diálogo es controlado por un gestor flexible guiado por eventos donde en cada momento se va tomando decisiones sobre el flujo del diálogo según la historia y el estado actual del mismo. Para ver una descripción más detallada del servicio se puede consultar el apéndice A. Algunos trabajos de evaluación de medidas de confianza en este dominio son los realizados en la Universidad Carnegie Mellon (Zhang y Rudnicky, 2001; Jiang et al 2001) y en la Universidad de Colorado (San-Segundo et al, 2000a; San-Segundo et al, 2001a; Hacioglu y Ward, 2002; Sameer y Ward, 2002) que proponen medidas al nivel de palabra, concepto y frase. Por último, cabe comentar el trabajo realizado por Carpenter (Carpenter et al, 2001), donde se proponen parámetros del diálogo para evaluar la calidad de la interacción.

Además de trabajar con el sistema CU-Communicator de la Universidad de Colorado, se estudiarán técnicas de medida de confianza para los sistemas de reconocimiento desarrollados en esta tesis, tanto para el sistema de reconocimiento de nombres deletreados como para el reconocedor de fechas y horas. Algunos trabajos previos realizados sobre reconocedores de nombres deletreados para la detección de errores o palabras fuera de vocabulario son los realizados en France Telecom y el GTH (Grupo de tecnología del Habla)(Jouvet y Monné, 1999b; San-Segundo et al, 2001c; Martín, 2001).

Por último, cabe comentar que en (Carpenter et al, 2001; San-Segundo et al, 2001e) podemos ver varias propuestas para el análisis automático on-line de la calidad de la interacción en los diálogos usuario-sistema.

2.5 Gestión de diálogo

El gestor del diálogo es la piedra angular de un Servidor Vocal Interactivo sobre la que giran el resto de módulos. Este módulo es el encargado de gestionar la interacción del sistema con el usuario, haciendo uso de la funcionalidad y prestaciones que le ofrecen el resto de los módulos. La importancia de este elemento, dentro de la estructura de un SVI, radica en que la interacción con el usuario, y en definitiva, la percepción que el usuario adquiere del servicio, depende fuertemente de la correcta gestión que este módulo realice de los recursos disponibles en el resto de los módulos.

Para poder comprender mejor el resto del apartado presentaremos la definición de algunos conceptos importantes:

- **Objetivo del diálogo:** denominaremos posibles objetivos del diálogo a cada una de las partes del servicio que pueden ser ofrecidos por el sistema automático de forma independiente. Por ejemplo, en un servicio de información y reserva de viajes de tren, se podrían considerar objetivos independientes la realización de una reserva concreta, la petición de información sobre horarios o la consulta de precios. En una misma consulta, el usuario podría querer satisfacer varios de los objetivos posibles, como por ejemplo solicitar información de horarios para posteriormente hacer la reserva.

- **Datos del usuario:** consideraremos datos del usuario a toda aquella información que el sistema debe conocer del usuario para poder satisfacer alguno de los objetivos solicitados por él. Por ejemplo, en el caso de consultar los horarios de tren, el sistema necesita conocer las ciudades origen y destino, así como la fecha y franja horaria en la que se quiere realizar el viaje.

- **Información del sistema:** conjunto de datos que el sistema ofrece al usuario para satisfacer cada uno de los objetivos. Por ejemplo, si el usuario solicita horarios de viajes en tren, el sistema podría informar de la hora de salida, la hora de llegada y el tipo de tren para realizar cada uno de los viajes posibles. En el caso de que el usuario quisiera hacer una reserva, el sistema debería informarle del código de la reserva así como de los detalles de la reserva realizada.

Atendiendo al agente, usuario o sistema, que lleva la iniciativa del diálogo, podemos clasificar los gestores de diálogo en:

- *Gestor de diálogo de iniciativa del sistema.* En este caso es el sistema el que tiene definidas una serie de acciones y una serie de objetivos que puede satisfacer, de forma que le va preguntando al usuario, siempre en el mismo orden, los datos necesarios para satisfacer cada uno de esos objetivos. El sistema, y por tanto el servicio ofrecido, no puede satisfacer los objetivos en el orden que prefiera el

usuario sino en uno preestablecido por el diseñador (Sutton et al, 1998; San-Segundo et al, 1999; Baggia et al, 2000; Córdoba et al, 2000; San-Segundo et al, 2000c; Córdoba et al, 2001).

- *Gestor de diálogo de iniciativa mixta.* En estos sistemas, el usuario puede tener cierta iniciativa y puede cambiar el flujo del diálogo con ciertas limitaciones. Según la dificultad en la implementación del gestor de diálogo, podemos clasificar las diferentes posibilidades en:

  a) Flexibilidad en la secuencia de los datos. El usuario puede responder con más datos de los solicitados en la pregunta, o incluso responder a un dato no preguntado, pero siempre perteneciente al objetivo activo en ese momento y que viene fijado por el sistema. En este caso, la secuencia de objetivos está predefinida.

  b) Flexibilidad en la secuencia de objetivos. El usuario puede elegir el objetivo o parte del servicio que quiere que el sistema le ofrezca. En ese caso, se debe desarrollar un módulo de clasificación de objetivos que permita, en función del análisis conceptual realizado por el sistema de comprensión, decidir la parte del servicio solicitada. Esta flexibilidad a su vez, dispone de dos grados de dificultad según se permita al usuario decidir el objetivo al comienzo de la interacción o se permita saltos entre objetivos sin haber completado el anterior. En este último caso habría que mantener activo de forma constante el módulo de clasificación de objetivos para detectar posibles cambios.

  Un detalle importante que se debe tener en cuenta es que aunque exista cierta flexibilidad en la secuencia de datos o de objetivos, debe haber siempre una secuencia de datos y objetivos por defecto. Secuencia que deberá ser seguida por el gestor del diálogo en el caso en el que el usuario no quiera tomar la iniciativa.

  Para permitir este tipo de flexibilidad es necesario disponer de un sistema de reconocimiento de habla continua que abarque varios dominios y un sistema de comprensión capaz de traducir la secuencia de palabras a secuencia de conceptos. Ejemplos de este tipo de gestores son los desarrollados en MIT, CMU, CSLR, LIMSI o Philips (Zue et al, 1997b; Ward y Pellom, 1999; Pellom et al, 2000; Rudnicky et al, 1999; Rudnicky et al, 2000; Rosset et al, 1999; Zue et al, 2000; Lamel et al, 2000; Schrämm et al, 2000).

- *Gestor de diálogo de iniciativa del usuario.* En este caso, es el usuario el que va definiendo el curso del diálogo y decidiendo el tipo de servicio que desea. Este tipo de gestores deben hacer frente a una mayor variedad de objetivos, lo que requiere de la gestión de grandes cantidades de conocimiento/información, y por tanto, de sistemas de reconocimiento y comprensión más potentes. El estado de la tecnología actual no permite disponer de estos sistemas con la suficiente robustez como para ser utilizados en servicios reales de atención al público.

Debido a la relevancia del módulo de gestión de diálogo, el proceso de diseño de este módulo es una tarea crítica dentro del desarrollo de un SVI. Atendiendo a la bibliografía

podemos considerar las siguientes estrategias de diseño (EAGLES, 1994; Eskenazi et al, 1999):

- *Intuición:* en este caso la definición del diálogo se realiza en base a la experiencia del desarrollador. Esta opción, aunque el coste de desarrollo es muy reducido, tiene el inconveniente de que la experiencia de una persona, por muy amplia que esta sea, no es suficiente para modelar los comportamientos de la gran variedad de usuarios existentes en servicios orientados al gran público. Además, la opinión del propio experto puede estar sesgada por un conocimiento parcial de la tecnología existente.

- *Observación:* este tipo de diseño se basa en el análisis de diálogos reales entre personas (usuario – operador) en servicios análogos. Este método permite el análisis de la variedad en los comportamientos de los usuarios pero tiene el inconveniente de que es bastante costoso en tiempo y recursos porque hay que escuchar, transcribir y analizar los diálogos grabados. Por otro lado, una interacción sistema-persona presenta matices diferentes respecto de una interacción persona-persona como son las diferencias en la velocidad de locución o la utilización de construcciones gramaticales diferentes (formal vs. informal).

- *Simulación:* en este caso se trabaja con la herramienta de Mago de Oz (Wizard of Oz, WOZ), a través de la cual se simula el comportamiento de un sistema completamente automático. Esta herramienta está manipulada por un operador humano que se encarga de realizar parte de las tareas, como la de reconocimiento o comprensión. El objetivo es engañar al usuario y hacerle pensar que está interactuando con un sistema realmente automático y de esta forma poder aprender el comportamiento que tendrá ante el sistema a desarrollar. Esta técnica tiene la ventaja de que permite analizar interacciones entre un sistema automático y un usuario pero los costes de desarrollo son bastante elevados, tanto por el desarrollo de la aplicación de Mago de Oz, como por la tarea de transcripción y análisis de las conversaciones.

- *Proceso iterativo*: este tipo de diseño propone el desarrollo de una primera versión del sistema (diseñada basada en la intuición) y la prueba directa con usuarios reales. A partir de este punto se genera un proceso iterativo de análisis y mejora del sistema hasta llegar a un punto de estabilidad. Esta solución requiere de grandes periodos de análisis de diálogos y de la necesidad de implementar un gestor fácilmente modificable que permita una adaptación constante.

En esta tesis se presenta una metodología para el diseño de gestores de diálogo en la que se combinan las técnicas descritas anteriormente (San-Segundo et al, 2001c; San-Segundo et al, 2001e; San-Segundo et al, 2001f; San-Segundo et al, 2001h; Ramos, 2001). Esta metodología se presenta a través del diseño de un servicio de información y reserva de billetes de tren por línea telefónica. Esta metodología propuesta, aunque se aplicará a un caso de diálogo guiado por el sistema, es directamente aplicable al diseño del diálogo por defecto que siempre debe existir en los gestores con iniciativa mixta. La metodología propuesta es similar al Life-Cycle Model descrito por el Prof. Bernsen (Bernsen et al, 1998; www.disc2.dk). En nuestro caso incorporamos una nueva etapa, diseño por *observación*, en la que se analizan diálogos entre personas en un servicio

similar al que se quiere automatizar. Además, presentaremos medidas de evaluación para comparar cada una de las posibles soluciones de diseño en las diferentes etapas. El Life-Cycle Model presentado por Bernsen está basado en las metodologías de diseño software. En el proyecto fin de carrera de Javier López (López, 2001), podemos ver una comparativa de diferentes metodologías para el desarrollo de programas software.

En otra línea de investigación llevada por Levin y Pieraccini (Levin et al, 1997; Levin et al, 2000) se trabaja en la definición de un modelo matemático del diálogo sistema–persona. Este modelo puede ser utilizado para definir algoritmos de aprendizaje que permitan calcular los parámetros del modelo de forma automática a partir de diálogos transcritos. En este trabajo se propone la posibilidad de describir formalmente el diálogo como una secuencia de procesos de decisión en términos de espacios de estados, conjunto de acciones a realizar en cada estado, y estrategia. Las técnicas de aprendizaje automático tienen el inconveniente de que el control por parte del desarrollador es menor, y cualquier cambio del servicio requiere reentrenar el modelo a partir de nuevos datos que se deben conseguir y etiquetar.

En todo sistema automático, debido a que los módulos de reconocimiento y comprensión pueden cometer errores, es imprescindible que el sistema realice la confirmación de los datos aportados por el usuario. Esta confirmación de los datos obliga inevitablemente a que se realicen más preguntas en el diálogo, dando lugar a un mayor número de interacciones. Por esta razón, es muy importante analizar diferentes mecanismos de confirmación, evaluar su carga en el diálogo y decidir cuál utilizar en cada caso. En este último punto tienen especial interés las medidas de confianza. Estas medidas permiten gestionar mejor el tipo de confirmación a realizar, pudiendo incluso sugerir la acción de no confirmar ante la alta confianza de un dato, con la reducción de preguntas que eso reporta (Lavelle et al, 1999; Bouwman et al, 1999; Sturm et al, 1999; San-Segundo et al, 2001f). Los principales tipos de confirmación son los siguientes:

- Confirmación explícita: para cada dato se le pregunta al usuario de forma directa si lo que ha reconocido/entendido el sistema es correcto. Esta solución asegura la confirmación del dato pero ralentiza enormemente el diálogo.

- Confirmación implícita: antes de continuar con la interacción se informa al usuario de los datos reconocidos/comprendidos durante el último paso del diálogo. En este caso se da por bueno el dato y no se pregunta al usuario sobre su validez, simplemente se va realimentando para que el usuario esté informado. Esta opción es la que menor carga supone pero sólo se puede realizar en condiciones de gran confianza. En esta tesis haremos una descripción detallada de las técnicas o estrategias de confirmación disponibles y describiremos la incorporación de medidas de confianza en el diseño de los mecanismos de confirmación de datos.

En todo servicio automático existe cierta funcionalidad de carácter general que debe estar activa en cualquier punto del diálogo. Las principales acciones son las siguientes:

- *Mecanismos de corrección o de recuperación de errores.* Cuando el sistema automático hace suposiciones acerca del valor de un dato (por ejemplo en la

confirmación implícita), debe establecer mecanismos de corrección que permitan al usuario corregir un posible error; volviendo hacia atrás en el diálogo o comenzando de nuevo la interacción (San-Segundo et al, 2001e).

- *Ayuda*. En este caso, el sistema debe detectar la necesidad de ayuda por parte del usuario, bien porque el usuario la solicite explícitamente o bien porque esté teniendo problemas en la interacción. Esta ayuda podrá depender del punto del diálogo, o ser general al servicio.

- *Pausar/Continuar*. Se debe permitir que el usuario, en cualquier punto del diálogo, pueda suspender la interacción para realizar alguna tarea y reanudarla después desde el mismo punto. De esta forma se evita que el usuario tenga que interrumpir la consulta/llamada y comenzarla de nuevo otra vez.

- *Repetir*. En cualquier punto, el usuario puede pedir que se le repita la última información dada, o la última pregunta realizada por el sistema.

Otro elemento importante en el desarrollo de los gestores de diálogo es su capacidad de adaptación al usuario con el que está interactuando: el modelado de usuario (Eckert e tal, 1997; Veldhuijzn van Zanten, 1999; San-Segundo et al, 2001g). Este tipo de adaptación es difícil en servicios por teléfono donde la interacción hombre-máquina dura relativamente poco (3 ó 4 minutos) y donde la diversidad de usuarios es elevada. Aun así, es posible y necesario definir mecanismos que permitan una generación de las preguntas del sistema acorde con la agilidad del usuario concreto que está realizando la interacción. De esta forma, ante situaciones de dificultad en las que el usuario tiene que repetir varias veces un dato, se puede ir teniendo en cuenta este hecho para que en posteriores interacciones dentro de una misma llamada, las preguntas tiendan a ser más claras y contengan información de cómo el usuario debe decir el dato solicitado.