UNIVERSIDAD POLITÉCNICA DE VALENCIA

UNIVERSIDAD
POLITECNICA
DE VALENCIA

Departamento de Sistemas Informáticos y Computación

Tesis de Máster:

Aportaciones al etiquetado y segmentación automática de diálogos en el corpus DIHANA[1]

Autor:
Vicent Tamarit Ballester

Director:

Carlos D. Martínez Hinarejos

13 de noviembre de 2008

[1] Trabajo realizado bajo las becas FPI del proyecto subvencionado por el Ministerio de Educación y Ciencia Español TIN2006-15694-C02-01.

II

Resumen

El siguiente trabajo estudia varios métodos para el etiquetado automático de segmentos en los sistemas de diálogo hablados. Concretamente se centra la experimentación en el corpus de diálogo DIHANA. El estudio aborda la eficacia de la prosodia (información extraída de la señal, que caracteriza el habla) por sí misma para identificar actos de diálogo y su combinación con las transcripciones de las intervenciones. También se presenta un método de etiquetado basado en la transcripción que utiliza HMMs. Este modelo se presenta en distintas versiones, fruto de realizar distintas asunciones en el desarrollo del planteamiento por máxima verosimilitud. Se presenta también otro método basado en la transcripción que utiliza técnicas de alineamiento típicas de la traducción automática.

Abstract

The present work studies some methods for the automatic labeling of segments in spoken dialog systems. Specifically, the experiments are focused on the DIHANA dialog corpus. The work tests the prosody (information extracted from the signal, that characterizes the speech) as a feature for the dialog act identification. The experiments explore the combination of the prosody with the transcriptions of the turns. We also present a labeling method based on the transcriptions that use HMMs. This model is presented in different versions, as a result of making different assumptions in the development of the maximum likelihood approach. Furthermore, we present other method based on the transcription which is inspired by the alignments usually used in machine translation.

Índice general

Capítulo 1

Introducción

En este capítulo se incluye una descripción de los sistemas de diálogo y las partes que lo componen. Se presenta el área de trabajo de los sistemas de diálogo abordada en esta tesis y se fijan los objetivos del presente trabajo.

1.1. Sistemas de diálogo

Los sistemas de diálogo son un área de aplicación de las tecnologías del lenguaje en el que un humano interactúa con una máquina dentro del marco de una tarea concreta (reserva de trenes, centralita telefónica, servicio técnico,...) empleando el diálogo (generalmente hablado) para conseguir su objetivo. Un sistema de diálogo implica interacción hombre-máquina mediante el lenguaje.

Desde un punto de vista teórico, un sistema de diálogo consta de cuatro partes fundamentales [23]:

- Primero necesita un módulo de entrada de datos que recoja las interacciones del usuario. La interacción puede realizarse mediante entrada de texto, mediante voz o una combinación de estas características y otras interacciones.

- Un módulo semántico que extraiga información útil para el sistema, es decir, que capte las intenciones del usuario y los datos que aporta.

- Un gestor de diálogo que procesa la información de entrada para generar una respuesta coherente.

- Un sistema de salida que, una vez se obtienen los datos para generar la respuesta, la transmite al usuario, utilizando para ello el mismo sistema de entrada de datos o varios a la vez (por ejemplo, voz mientras se muestra algo en pantalla).

Como se puede ver, en un sistema de diálogo intervienen varias disciplinas de tecnología del lenguaje (procesamiento de lenguaje natural, reconocimiento del habla, ...) y la inteligencia artificial. Cada una de estas partes presenta problemas propios y suponen, independientemente, un reto para los investigadores.

1.2. Arquitectura básica

Los sistemas de diálogo se desarrollan sobre cuatro áreas de trabajo que incorporan técnicas de inteligencia artificial o procesado del lenguaje. Estas áreas se detallan en los siguientes apartados.

1.2.1. Interacción con el usuario

En un sistema de diálogo, las maneras más comunes de interactuar entre el usuario y el sistema es utilizando la voz, de la misma manera que cuando dialogamos con otro humano. También es posible hacerlo mediante la escritura, tecleando las interacciones al igual que haríamos en uno de los populares programas de mensajería instantánea.

Si la entrada es simplemente texto ésta puede ser procesada directamente por el módulo semántico. Sin embargo, si la entrada es mediante voz, hace falta realizar primero un reconocimiento del habla. El reconocimiento del habla tiene en la actualidad una limitación muy importante: el área de aplicación (dominio). No existe un reconocedor universal, y aunque el software básico utilizado para el reconocimiento no se modifica, para cada tarea es necesario diseñar un conjunto de elementos que acoten el sistema, en especial, el vocabulario y las frases del dominio que el sistema reconocerá.

Además de la limitación temática, los sistemas de reconocimiento de la voz están también sujetos a las condiciones acústicas en las que van a ponerse en producción. Por ejemplo, hay que cuidar que no haya ruido ambiente, que no se solapen los locutores o que no haya reverberación, algo no siempre posible en sistemas reales.

1.2.2. Extracción de información semántica

En los sistemas de diálogo, la semántica juega un papel muy importante y debemos ser capaces de extraer intenciones comunicativas y los datos de las intervenciones. Se trata, en definitiva, de entender al usuario para poder satisfacer sus peticiones y generar una respuesta coherente. Generalmente la salida de este módulo es un conjunto de pares de atributos junto con su valor correspondiente y una etiqueta de intencionalidad.

1.2.3. Gestor del diálogo

El cuerpo central de un sistema de diálogo es el gestor del diálogo. Éste módulo es el encargado de procesar la información de entrada, suministrada por el módulo semántico, y preparar la respuesta. Se trata de modelar la estructura del discurso, es decir, representar los posibles estados del diálogo y definir lo que el sistema debe hacer en cada uno de ellos.

1.2.4. Generador de respuestas

Cuando el gestor determina cuál es la respuesta adecuada hay que presentarle ésta de nuevo al usuario. Si la respuesta se limita a mostrar algo por pantalla, ya sea un dibujo o un texto, el proceso es sencillo. Sin embargo, si el sistema debe dar respuesta oral es necesario un sintetizador de voz [9] que devuelva la respuesta al usuario.

1.3. Métodos para la gestión de diálogo

La estructura del discurso es un problema multidisciplinar que no tiene actualmente una solución definitiva. Según [26], hay seis tipos de aproximaciones al análisis del discurso: sociolingüística interactiva, análisis de la conversación, la etnografía de la comunicación, análisis de variaciones, pragmática y teoría de los actos comunicativos [3, 27]. Esta última teoría del discurso se centra en actos comunicativos llevados a cabo mediante el habla y es el marco en el que muchos autores tratan de modelar la estructura del diálogo.

Más allá de los estudios teóricos, han habido varios estudios en implementaciones prácticas de sistemas de diálogo. Estas aplicaciones utilizan diferentes métodos para modelar el discurso y permitir al sistema interactuar con el usuario, es decir, gestionar adecuadamente el diálogo. Estos métodos pueden ser basados en reglas o en datos.

Los métodos basados en reglas utilizan un conjunto de reglas definidas a mano, por ejemplo mediante gramáticas de diálogo [20] o agentes de gestión del diálogo [6], que describen la reacción del sistema a las diferentes entradas del sistema dependiendo de la situación del diálogo [1]. Esta política de reacción del sistema se conoce comúnmente como estrategia del diálogo. Las reglas pueden ser definidas desde cero utilizando conocimiento previo de la tarea, pero es habitual adquirir diálogos (generalmente diálogos entre humanos) para comprobar la naturaleza de la interacción de los diálogos en la tarea correspondiente. A veces, se emula una máquina, algo conocido como la técnica del Mago de Oz [10]. Esto se debe a que los usuarios se comportan de manera diferente cuando se dan cuenta de que la otra parte es un sistema automático, con lo que el modelado de diálogos entre humanos no resulta adecuado. El desarrollo de las reglas lleva mucho tiempo y, por lo tanto, sólo sistemas simples y dirigidos por *slots* han implementado esta aproximación [14].

Los métodos basados en datos tratan de elaborar la estrategia de diálogo utilizando modelos estadísticos que pueden ser aprendidos de diálogos anotados. Para ello emplean técnicas de aprendizaje automático. Hay muchos modelos de diálogos disponibles. Los modelos de aprendizaje por refuerzo [31] y las redes bayesianas [21] son buenos ejemplos de los diálogos basados en datos. Muchas de estas aproximaciones se basan en diálogos anotados con actos comunicativos, conocidos como actos de diálogo.

1.4. El etiquetado semántico

Como se ha comentado, muchos autores modelan la estructura del diálogo con métodos basados en los datos. Estos métodos se apoyan en corpus anotados con actos de diálogo para aprender los parámetros del modelo. El uso de actos de diálogo asume que la información necesaria para elegir la reacción del sistema se encuentra en la anotación de los actos de diálogo de los turnos de diálogo previos, especialmente el último turno de usuario. El marco estadístico ha sido aplicado con éxito en la anotación y modelado de diálogo [30, 33].

En la literatura, los diálogos se describen como una sucesión de intervenciones, llamadas turnos. Cada turno, a su vez, se descompone en otras unidades más pequeñas: los segmentos, unidades mínimas de contenido semántico. Estos segmentos se etiquetan con un acto de diálogo, diseñado para transferir esa información semántica al sistema. Un acto de diálogo etiqueta un segmento de conversación, en función de cuál sea la intención del segmento. Cada uno de estos segmentos que refleja una intención o contiene información útil para el sistema se denomina en la literatura *utterance*[30],

aunque en este trabajo también nos referiremos a ellos simplemente como ”segmentos”.

Para poder aplicar los métodos basados en datos a la inferencia de la estrategia del diálogo, los corpus de diálogo deben ser anotados utilizando actos de diálogo. Esto implica la definición de un esquema de actos de diálogo para etiquetar los turnos. El etiquetado depende del tipo de diálogos. En los diálogos entre humanos no orientados a ninguna tarea, el esquema de actos de diálogo se dirige normalmente a estudios posteriores en la estructura del diálogo, funciones comunicativas o resolución de anáforas. En los diálogos orientados a una tarea (por ejemplo, la reserva de trenes), el esquema de etiquetado debe tener en cuenta las funciones comunicativas y la información de la tarea (ciudad de salida, horas de salida, ...).

El esquema de etiquetado más popular es el Dialogue Act Mark-up in Several Layers (DAMSL) [7]. DAMSL define varias capas, cada una de ellas referida a un aspecto concreto del diálogo: estado de la comunicación, nivel de información, función de búsqueda adelante y atrás. Esta estructura permite gran flexibilidad en la definición del esquema de actos de diálogo, pero no ofrece la posibilidad de reflejar los datos de la tarea en el etiquetado.

El esquema DAMSL presenta una gran complejidad, con un gran número de combinaciones que apenas aparecen en diálogos de otros corpora. Por lo tanto, aunque muchos diálogos se han anotado con DAMSL, se ha hecho con una redefinición del mismo que presenta un conjunto más reducido de etiquetas. El conjunto más popular derivado de DAMSL es SWBD-DAMSL [15], definido para etiquetar el corpus SwitchBoard [13].

Una buena propuesta para la anotación de corpus orientados a la tarea es el Interchange Format (IF), definido en principio para traducción automática en el proyecto C-Star [16]. El formato IF define tres niveles diferentes en cada etiqueta: el acto comunicativo, el concepto y el argumento. Este sistema de etiquetado es el empleado en el corpus DIHANA, empleado en los experimentos de esta tesis.

1.5. Objetivos del proyecto

La finalidad de este trabajo es estudiar nuevas técnicas de etiquetado para sistemas de diálogo. Entre la propuestas para el etiquetado destaca la inclusión de características de la señal vocal en el proceso de etiquetado. La idea es poder contar con la señal más allá de la transcripción y que ésta sea útil en la fase de extracción semántica. Para ello buscaremos las características de la señal vocal que puedan aportar información al etiquetado. Algunos trabajos [30] han planteado la posibilidad de incorporar información de prosodia, obtenida directamente de la señal, para mejorar el etiquetado de *utterances*.

Además, se han estudiado dos métodos de etiquetado basados únicamente en las transcripciones. Uno de ellos basa su funcionamiento en HMMs y el otro deriva de una técnica de alineamiento utilizada en traducción automática.

El trabajo presenta la siguiente estructura. En el Capítulo 2 se describe el corpus de diálogo utilizado, DIHANA. En el Capítulo 3 se incluye un estudio acerca de las características de la señal que pueden ser útiles para la identificación de actos de diálogo. El Capítulo 4 presenta una técnica de etiquetado basada en la señal combinada con la transcripción, que hace uso de un clasificador basado en k-NN para las características de prosodia y otro basado en HMM para las transcripciones. En el Capítulo 5 se describe un método de clasificación de actos que únicamente se vale de la señal

acústica y de un clasificador basado en HMM. En el Capítulo 6 se estudia un modelo de etiquetado basado en la transcripción y en HMMs. El Capítulo 7 recoge un modelo de segmentación jerárquico basado en NGT, una técnica de alineamiento pensada para traducción automática, pero adaptada a diálogo. Para finalizar, el Capítulo 8 recoge las conclusiones más importantes del trabajo y propone nuevas tareas.

Capítulo 2

Descripción y procesado del corpus DIHANA

Previamente a la exposición de los métodos de etiquetado propuestos, se presenta una descripción del corpus utilizado para las pruebas y que servirá de referencia. Es un recorrido por su creación, características más importantes y procesado específico para los objetivos de este trabajo.

2.1. Características del corpus

El corpus de diálogo que se ha utilizado en este trabajo es el corpus DIHANA [4]. Este corpus de diálogo de habla espontánea en castellano ha sido creado por la Universidad del País Vasco, la Universidad de Zaragoza y la Universidad Politécnica de Valencia.

DIHANA está orientado a una tarea concreta. Está compuesto por 900 diálogos usuario-máquina sobre la obtención de información de horarios, precios y servicios de trenes en España. Puede ser considerado como un corpus de tamaño medio por su número de diálogos y como un corpus pequeño si atendemos al tamaño de su vocabulario (823 palabras).

La adquisición de DIHANA se llevó a cabo utilizando la técnica del Mago de Oz [10]. Esta adquisición sólo está restringida a nivel semántico (los usuarios solicitan siempre información sobre trenes), pero no tiene ningún tipo de restricción léxica ni sintáctica. Para limitar la semántica se definieron situaciones en las que el usuario debía desenvolverse (escenarios).

En total, en la adquisición del corpus DIHANA participaron 225 locutores diferentes (153 hombres y 72 mujeres), con pequeñas variantes dialectales. Los 900 diálogos que se grabaron comprenden un total de 6.280 turnos de usuario y 9.133 turnos de sistema. En promedio, cada diálogo consta de siete turnos de usuario y diez de sistema, con una media de 7,7 palabras por turno de usuario. Hay cinco horas y media de grabación.

Los diferentes turnos se segmentaron en *utterances*. Por supuesto, en un turno pueden aparecer más de una *utterance*. De hecho cada turno tiene, de media, 1,5 *utterances*. Cada *utterance* se identifica con un acto de diálogo y se anota con una etiqueta.

Antes de realizar el etiquetado, es necesario definir el conjunto de etiquetas. Éstas deben dar cuenta de las interacciones del usuario y del sistema, recogiendo aspectos de las *utterances* como su intención o los datos que demanda o suministra al sistema.

Un buen esquema de etiquetado es el Interchange Format (IF) definido en el proyecto C-STAR [16]. A pesar de estar definido para una tarea de Traducción Automática, puede ser utilizado para anotación de diálogos [11].

Basado en el formato IF, se definió en [2] un esquema de tres niveles para las *utterances* de DIHANA. Este conjunto de actos de diálogo representa la intención general del segmento (primer nivel), así como información semántica más precisa específica de la tarea (segundo y tercer niveles). El segundo nivel contiene información implícita en el segmento (datos usados o modificados según la intención determinada por el primer nivel). El tercer nivel representa información específica presente en el segmento. En la Tabla 2.1 puede verse un ejemplo de un diálogo anotado. Cada diálogo se divide en turnos y cada turno en *utterances*. Cada *utterance* se marca con una etiqueta de tres niveles.

Tras el etiquetado semiautomático [2], quedaron 248 etiquetas de tres niveles diferentes (153 para segmentos de usuario y 95 para sistema). Considerando primer y segundo nivel hay 72 etiquetas (45 para usuario y 27 para sistema). Teniendo en cuenta sólo primer nivel existen 16 etiquetas (7 para usuario y 9 para sistema).

2.2. Preprocesado del corpus transcrito

Antes de utilizar el corpus se realizó un preprocesamiento para reducir la complejidad del corpus y sus estructuras. Esto es necesario para obtener mejores modelos, dado que los datos como tal tienen una gran variabilidad y dispersión. Este preproceso incluye:

- Categorización de algunas palabras, reuniendo bajo una misma notación nombres de ciudades, horas, fechas, tipos de tren, ...
- Todas las palabras se reescribieron en minúsculas.
- Los signos de puntuación se separaron de las palabras.
- Todas las palabras están marcadas según su locutor (U para usuario, M para el sistema)

2.3. Procesamiento de la señal

Cada uno de los turnos de diálogo grabados contiene a su vez uno o varios actos de diálogo. Por ello, antes de empezar a trabajar sobre ellos, la primera tarea fue partir los turnos en segmentos. Este procesado es necesario ya que algunas de las técnicas que se emplean en el trabajo se realizan directamente sobre segmentos. Para realizar la segmentación se ha utilizado una versión modificada de el reconocedor de habla ATROS [17] que realiza alineamiento forzado entre la señal y la transcripción.

2.3.1. Modelos acústicos

Para poder garantizar un buen alineamiento se entrenaron modelos acústicos adaptados a la tarea; es decir, las propias grabaciones de los turnos se utilizaron para entrenar los modelos. Se utilizó HTK [32] y se entrenaron modelos de Markov de 6 estados con 16 gaussianas en cada estado.

Tabla 2.1: Ejemplo de un diálogo anotado del corpus DIHANA. *Nil* marca la ausencia de información.

Locutor	Segmento	Transcripción		
		Nivel 1	Nivel 2	Nivel 3
M	M1	Bienvenido al sistema de información de trenes. ¿En qué puedo ayudarle?		
		Apertura	Nil	Nil
U	U1	Quiero saber los horarios de salida desde Valencia		
		Pregunta	Hora_salida	Origen
	U2	a Madrid		
		Pregunta	Hora_salida	Destino
	U3	para el 15 de mayo de 2004.		
		Pregunta	Hora_salida	Día
M	M2	¿Quiere salir el sábado, 15 de mayo de 2004?		
		Confirmación	Día	Día
U	U4	Sí.		
		Afirmación	Día	Nil
M	M3	Le consulto horarios para trenes desde Valencia a Madrid el sábado 15 de mayo de 2004.		
		Confirmación	Hora_salida	Destino, Día, Origen
	M4	Un momento, por favor.		
		Espera	Nil	Nil
	M5	Hay varios trenes. El primero sale a las 7:45 y llega a las 11:14, el último sale a las 18:45 y llega a las 22:18.		
		Respuesta	Hora_salida	Hora_llegada, Hora_salida Numero_relativo_orden Numero_trenes
	S6	¿Desea algo más?		
		Nueva_consulta	Nil	Nil
U	U5	Sí, quiero saber el precio del tren que sale a las 7:45.		
		Pregunta	Precio	Hora_salida
M	M7	Ese tren en clase turista cuesta 35.50 euros.		
		Respuesta	Precio	Clase, Precio
	M8	¿Desea algo más?		
		Nueva_consulta	Nil	Nil
U	U6	No, gracias.		
		Cierre	Nil	Nil
M	M9	Gracias por utilizar este servicio. Feliz viaje.		
		Cierre	Nil	Nil

2.3.2. Segmentación en actos de diálogo

Utilizando los modelos acústicos entrenados en ATROS, se realizó un alineamiento forzado de cada transcripción con la secuencia acústica. En la Figura 2.1 se puede ver un extracto de la salida del alineamiento. De la salida del alineamiento nos interesa la línea *Segmentación*, que contiene valores de frames (1 frame $= 10ms$). El primer valor indica el comienzo de la frase y a partir de ahí cada frame indica el final de una palabra.

Para poder realizar la segmentación hace falta también información sobre los actos de diálogo. Primero hay que definir si queremos dividir el corpus en actos de primer, segundo o tercer nivel. Luego es necesario conocer qué turnos contienen más de un

```
Frase:  11  /home/vtamarit/dihana/audio_CC/B101_BB3c0_T04u.CC
TransOrt:  <micro>
RecOut:  <si , me puede decir el precio del billete .>
Traduccion:  <    >
Score: 215736.8438
Segmentacion: 29 47 50 59 84 102 117 147 161 202 214 219
Scores Parciales: 38791.20 53708.48 57331.96 66601.71 84666.18
102398.34 123912.66 147679.41 155996.67 182525.16 193091.38 215736.34
Segundos:  0.02  Frames:  220
Segundos/Frame:  0.000091
AciertoReco:  NO
```

Figura 2.1: Salida del alineamiento

acto de diálogo (según el nivel que hayamos definido) y por dónde hay que partirlos. Esta información se puede obtener de las transcripciones de los diálogos y se ordenó en un fichero de texto tal y como aparece en la Figura 2.2. Al mismo tiempo se parten las transcripciones generando ya los ficheros con las etiquetas para cada acto de diálogo.

```
B101_BB3c0_2  2 Negacion 6 Respuesta
B101_BB5c4_2  2 Negacion 6 Respuesta
B101_BC0a0_2  2 Negacion 6 Respuesta
B103_BA0c2_2  2 Afirmacion 16 Pregunta
B103_BB3c2_2  2 Afirmacion 11 Pregunta
B104_BB5c7_2  2 Afirmacion 26 Pregunta
B104_BC1a1_7  2 Afirmacion 23 Respuesta
B105_BB3c4_5  2 Afirmacion 6 Respuesta
```

Figura 2.2: Extracto del fichero de actos de diálogos. Cada línea representa aquellos turnos de usuario que contienen dos o más actos de diálogo, indicando en cada caso por qué palabra hay que cortar.

Con la información extraída de las transcripciones y los alineamientos se partieron las grabaciones de aquellos actos de diálogo que lo necesitaban. Así se obtuvo como resultado todo el corpus segmentado a nivel de señal para todos los niveles de actos de diálogo.

Capítulo 3

Selección de características de la señal de audio

Antes de empezar a trabajar con el corpus se realizó un estudio sobre las características prosódicas que podrían aportar información para la distinción entre actos de diálogo. En las siguientes páginas se presentan las características extraídas de la señal acústica y una comparación entre actos de diálogo a través de tres de ellas: pitch, media de la señal y duración de la *utterance*.

3.1. Extracción de características

A partir de la señal acústica se han extraído diferentes características que pueden ser útiles para diferenciar actos de diálogo. Estas características están propuestas en [30] y son las siguientes:

- *Energía*: Es la amplitud de señal acústica.
- *Energía de la voz*: La amplitud señal cuando el locutor habla.
- *Energía de los silencios*: Amplitud cuando se detectan silencios. Se consideran silencios aquellos tramos de más de $800ms$ con un valor de señal (amplitud) por debajo de 700.
- *Duración*: Duración del segmento.
- *Duración silencios*: Duración de los silencios del segmento.
- *Número de silencios*: Cuántos silencios hay en el segmento.
- *Pitch y media del pitch*: El pitch (o frecuencia fundamental -F0-) captura la entonación del locutor. Se utiliza en síntesis de voz para simular la entonación de las frases. Se dan más detalles en el punto siguiente.

Los valores de energía y pitch se calculan a lo largo de toda la señal desplazando una ventana sobre ella.

3.2. Cálculo del pitch

Una señal sinusoidal perfecta puede descomponerse como una suma de senos y cosenos. Si se realiza la conversión al dominio de la frecuencia se puede observar que existe una primera frecuencia de la cual las demás son múltiplos. A esta frecuencia se la conoce como *frecuencia fundamental*. En habla las señales no son perfectas y no se calcula una única F0, sino que, típicamente, se desplaza una ventana por la señal (en nuestro caso de 2048 muestras, con solapamiento de 512) a la cuál se aplica una ventana de Hamming.

El cálculo del pitch es un tema complicado del procesamiento de la señal acústica. Es complicado porque existen diferentes aproximaciones ligadas en gran medida al tipo de señal que se quiera procesar. Hay estimadores de pitch que funcionan bien en detección de notas musicales y en análisis del habla. Sin embargo es difícil encontrar un método que funcione bien en ambos campos. Normalmente un buen estimador para una aplicación concreta no da buenos resultados al llevarlo a otro dominio. En [12] hay un completo resumen de métodos para la estimación del pitch.

Nuestra elección para extraer el pitch ha sido el software Wavesurfer [29]. Este software está desarrollado por el Centro de Tecnología del Lenguaje de la KTH, en Suecia, y se distribuye como software libre bajo una licencia de tipo BSD. El programa está muy bien modularizado y se basa en la librería Snack [1], también del mismo grupo. Esta librería está pensada para poder crear herramientas de sonido de una manera cómoda y rápida en combinación con los lenguajes de script Tcl/Tk o Python.

3.3. Influencia de la prosodia en la clasificación

De todas las características calculadas, el estudio se ha centrado en tres: evolución del pitch, energía y duración. Para evaluar la influencia se han utilizado gráficas que promedian los valores en los dos primeros casos y un histograma en el tercero. El estudio se ha fijado para comprobar cómo se comportan estos valores según la clasificación de los segmentos en cuatro actos de diálogo que consideramos más sencillos de distinguir con las características elegidas. Estos son: afirmaciones, negaciones, preguntas y respuestas.

3.3.1. Pitch

Estudiar el pitch se justifica puesto que es una forma de capturar la entonación de la frase (incluso se puede utilizar para extraer información de grabaciones musicales). En la Figura 3.1 se representa el pitch y la señal de la frase ”Se puede ir desde Santurce a Bilbao” pronunciada como una afirmación, mientras que en la Figura 3.2 se muestra la misma información gráfica para la misma frase pronunciada como una pregunta. En dichas figuras se puede apreciar al inicio de la señal puntos que permiten diferenciar una entonación de otra.

Utilizando las conexiones de la librería Snack con Python se extrajeron los valores de pitch de todas las señales; se obtiene un valor de pitch cada 10 ms. Con estos valores se han elaborado gráficas de evolución. Se ha estudiado la diferenciación preguntas/respuestas y afirmación/negación.

[1] http://www.speech.kth.se/snack/

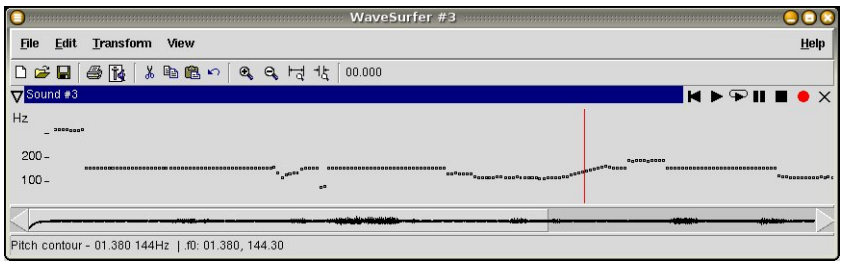

Figura 3.1: Pitch para la frase ”Se puede ir desde Santurce a Bilbao” pronunciada como una afirmación.

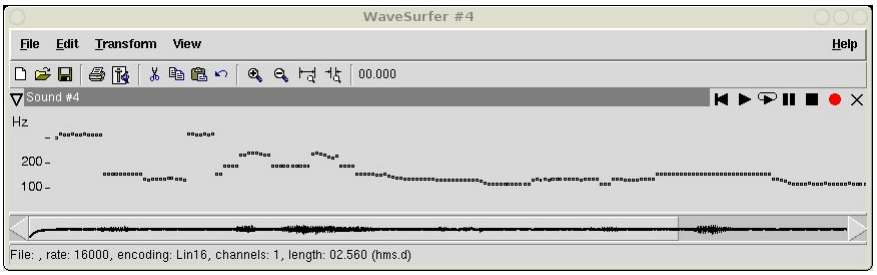

Figura 3.2: Pitch para la frase ”Se puede ir desde Santurce a Bilbao” pronunciada como una pregunta.

En la Figura 3.3 se pueden ver los resultados de la comparación entre preguntas y respuestas. Como se aprecia, las respuestas tienen, en promedio, unos valores mayores de pitch que las preguntas.

La Figura 3.4 muestra las evoluciones de pitch de afirmaciones y negaciones. Las diferencias entre ambos actos de diálogo parecen ser significativas. En promedio, el pitch de las negaciones supera al de las afirmaciones.

Las gráficas muestran la evolución del pitch a lo largo del tiempo. Para poder comparar los actos de diálogo se han normalizado las gráficas para ajustarlas a la que menos valores de pitch tenía y, además, se han promediado todos los valores de cada punto; se consigue así sacar el contorno medio de pitch para cada acto de diálogo.

3.3.2. Media de la señal

Otra característica de las grabaciones a estudiar fue la media de la señal. Esta medida puede ser un buen indicativo de la ”fuerza” con la que se habla. Cabría esperar que mientras que las preguntas se realizan en un tono más bajo, las respuestas, sobretodo las afirmaciones o negaciones, son más enérgicas.

Igual que en el punto anterior, se comparan las preguntas y respuestas por un lado y las afirmaciones y negaciones por otro. En la Figura 3.5 se muestran las gráficas de evolución de la amplitud en preguntas y respuestas, mientras que la Figura 3.6 recoge la evolución de la amplitud en afirmaciones y negaciones. Para obtener la amplitud se han considerado ventanas de 10 ms, para las que se ha calculado la amplitud media.

Aunque las gráficas no muestran grandes diferencias entre los actos comparados,

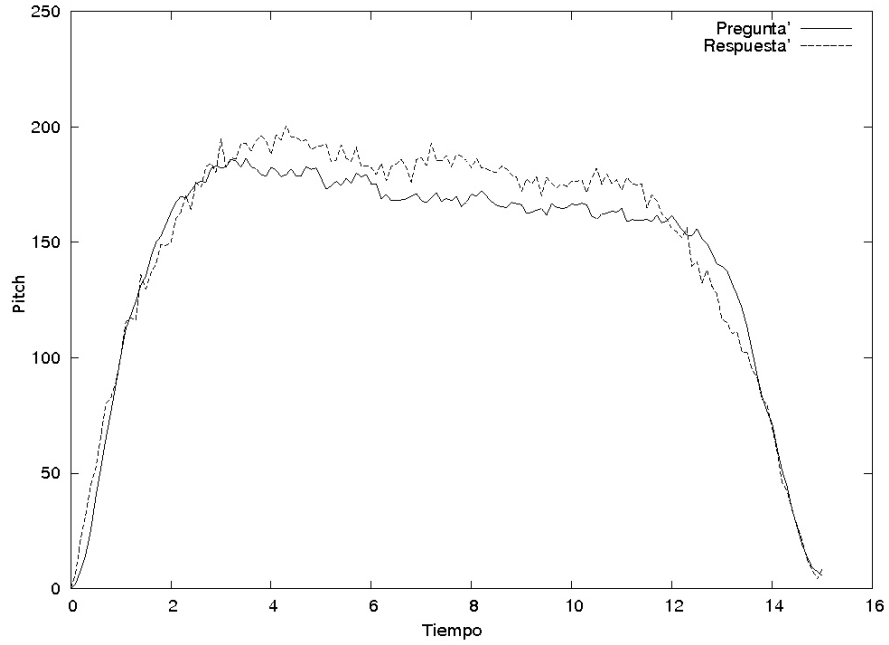

Figura 3.3: Evolución de los valores de pitch de las preguntas y respuestas.

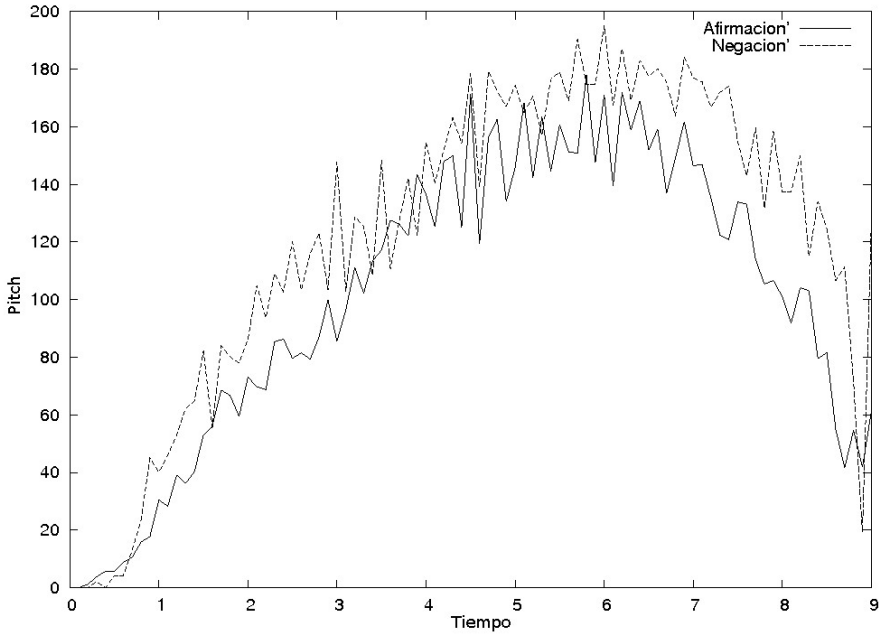

Figura 3.4: Evolución de los valores de pitch de las afirmaciones y negaciones.

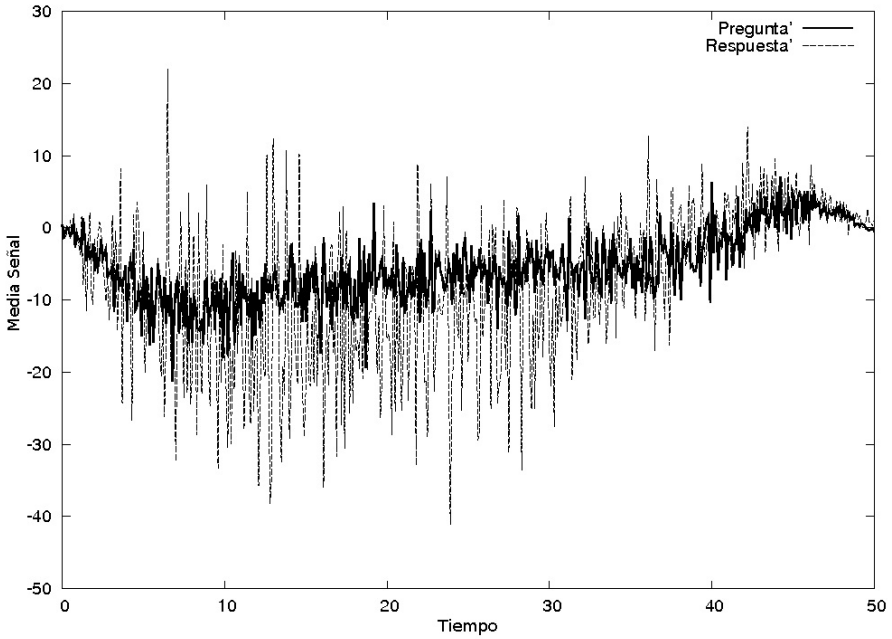

Figura 3.5: Evolución de la energía de la señal en preguntas y respuestas.

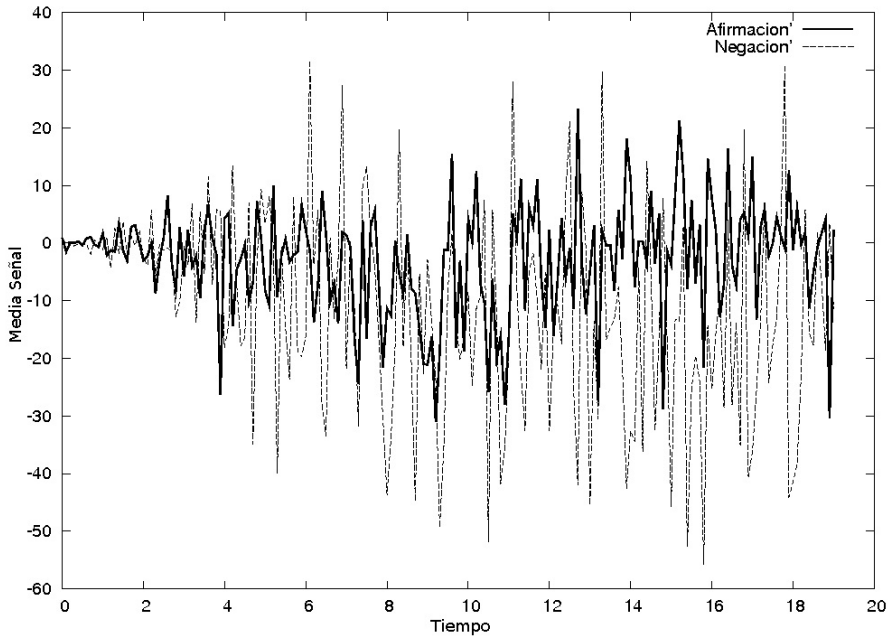

Figura 3.6: Evolución de la energía de la señal en afirmaciones y negaciones.

sí que se aprecia una importante diferencia que ya señalábamos como hipótesis: las afirmaciones y negaciones muestran amplitudes mayores que las preguntas y respuestas. En cualquier caso, no hay que olvidar que esta medida tiene una gran dependencia de las condiciones de grabación, como la distancia del micrófono, ruido ambiente, etcétera.

3.3.3. Duración

Sin duda algo que bien puede utilizarse para distinguir actos de diálogo es la duración del mismo. Una simple negación o afirmación no puede competir en duración a una pregunta o una respuesta. La Figura 3.7 representa los histogramas de los cuatro actos de diálogo estudiados.

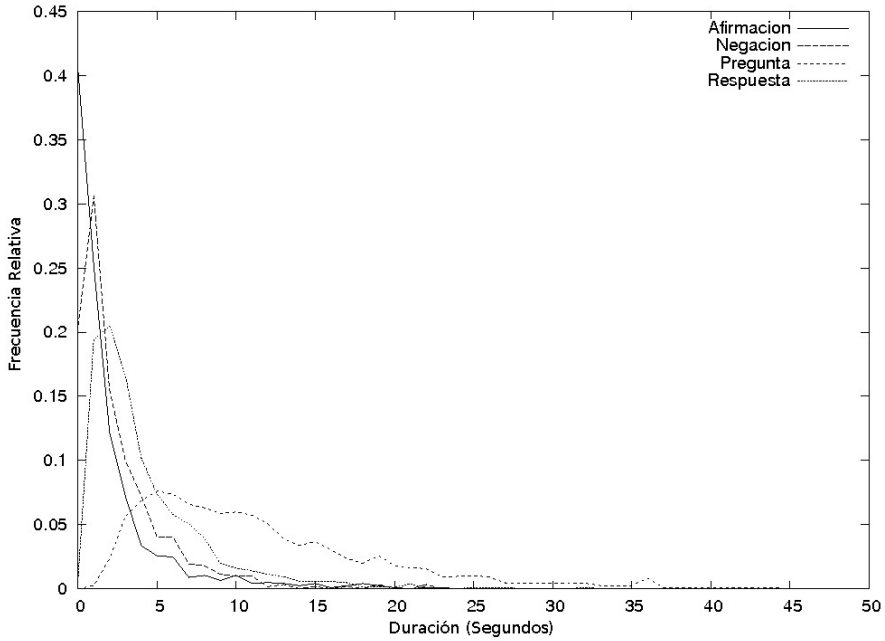

Figura 3.7: Histograma de la duraciones de los actos de diálogo.

Se pueden apreciar importantes diferencias de duración. Las afirmaciones y negaciones, obviamente, son más cortas, mientras que las respuestas y, sobre todo las preguntas, muestran duraciones más variables.

3.4. Conclusiones

Los resultados obtenidos son esperanzadores. Se aprecian diferencias a la hora de comparar actos como preguntas/respuestas o afirmación/negación. Obviamente el pitch, por sí solo, no es una buena medida discriminatoria, pero parece que si se sabe incorporar a un sistema de reconocimiento puede ayudar en la distinción de actos de diálogo.

Las otras dos características elegidas (media de la señal y duración) aportan poco. Tan sólo la duración se presenta como variable diferenciadora, pero si tenemos en cuenta que los actos de diálogo se encuentran ”encerrados” en turnos donde hay más actos, esta variable pierde prácticamente todo su poder. La energía, por su parte, se intuía como una buena manera de distinguir ciertos actos de diálogo (al afirmar o negar solemos hacerlo con más fuerza). Quizás para la energía habría que cambiar el modo de calcularla y analizarla; habría que estudiar el comportamiento de la energía desplazando una ventana sobre la señal, de igual forma que se hace con el pitch.

Capítulo 4

Técnicas basadas en la señal y el reconocimiento

El método de etiquetado en este capítulo es un modelo híbrido que reúne conocimiento obtenido directamente de la señal con el obtenido a partir de las transcripciones. Para cada tipo de características se desarrolla un clasificador diferente que luego se combina.

4.1. Objetivos

La finalidad de este estudio es comprobar el funcionamiento de las características acústicas para el etiquetado de segmentos, así como estimar las posibles mejoras que un sistema basado en señal podría proporcionar a un etiquetador basado en la transcripción.

La ventaja de utilizar la prosodia es doble. Por una parte, para extraer esas características no es necesario decodificar la señal (no es necesario pasarla por ningún reconocedor), evitando los posibles errores que pudiera generar y reduciendo el tiempo de reconocimiento. Por otro lado, derivado de lo anterior, estas características son aplicables a cualquier idioma; quizás para cada lengua haya que adaptarlas mínimamente, pero no es necesario pensar en nuevos modelos acústicos o modelos de lenguaje.

Para el etiquetado basado en las transcripciones se ha optado por un sistema de clasificación basado en Modelos de Markov, utilizado en la literatura para esta tarea. Sin embargo, para la clasificación basada en señal el clasificador elegido está basado en k-NN [8].

4.2. Preparación de las pruebas

Para probar ambos clasificadores se han utilizado únicamente dos particiones: el $60\,\%$ de las muestras para entrenamiento y el resto para pruebas, respetando la proporción para cada una de las clases. En total, de las 6.679 muestras (es decir, segmentos de usuario), 4.006 para entrenamiento y 2.673 para test.

La clase con más apariciones es "Pregunta", que aglutina el $42\,\%$ de los actos, lo que podemos tomar como un *baseline* para nuestra tarea. En total hay cinco clases:

Afirmación, Negación, Pregunta, Respuesta y Cierre. El problema de etiquetado se aborda como un problema de clasificación.

4.3. Clasificación basada en prosodia: k-NN

Dado que la información prosódica puede verse como un vector de datos, se ha optado por probar una clasificación basada en k-vecinos. Esta técnica necesita un conjunto de datos representativo de cada una de las clases que le sirvan de referencia. Para realizar la clasificación se define una medida de distancia entre los vectores de datos que representan cada muestra y se calcula esta distancia entre la muestra a clasificar y las de referencia. La muestra se clasifica dentro de la clase a la que pertenece la muestra más próxima. Dado un conjunto de entrenamiento de n muestras, cada una de ellas asociada a una clase $c(n_i)$ y un vector x de características a clasificar entre C clases, la clase a la que pertenece será la del prototipo más cercano:

$$c(x) = \arg \min_{c(n_i)} d(n_i, x) \tag{4.1}$$

La función de distancia d puede variar según el tipo de datos o la tarea de clasificación. La distancia euclídea es una medida habitual:

$$d(y, x) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2} \tag{4.2}$$

donde x e y son los puntos en el espacio y m su dimensión.

Esta distancia asume que todos las muestras de referencia son igual de importantes para determinar la clase. Sin embargo, cabe pensar que algunas muestras definen mejor una clase que otra, incluso que alguna clase sea más probable que otra. Por ello, la distancia se calcula considerando un peso w que puede estar asociado a la clase w_c a las muestras w_i o a cada una de las muestras de una clase w_{ci}.

$$d(y, x) = \sqrt{\sum_{i=1}^{m} w^2 (x_i - y_i)^2} \tag{4.3}$$

Para estimar los pesos se utiliza una formulación basada en la estimación del error por *Leaving One Out* para k-NN, definida como:

$$J_T(W) = \frac{1}{n} \sum_{x \in T} step \left(\frac{d(x, x^{=})}{d(x, x^{\neq})} \right) \tag{4.4}$$

Donde W es el conjunto de pesos para ser aprendido y $x^{=}$ y $x^{\neq}$ se refiere a los vecinos de la misma clase que la muestra actual y a los de las clases diferentes respectivamente. Si x está más cerca de un prototipo de su propia clase que de cualquier otro prototipo de una clase diferente, la muestra x se clasifica correctamente. En este caso $d(x, x^{=}) < d(x, x^{\neq})$ y el argumento de la función *step* es menor que 1. Si x está más cerca de otro prototipo de una clase diferente a la suya, se clasifica erróneamente y el argumento de la función *step* es mayor que uno. La función *step* se define como:

$$step(x) = \begin{cases} 0 & si \quad x <= 1 \\ 1 & si \quad x > 1 \end{cases} \tag{4.5}$$

A partir de la Fórmula 4.4 se construye una optimización basada en descenso por gradiente. Por esta razón la función $step$ se substituye por una función sigmoide S_β que sí es derivable:

$$S_\beta(z) = \frac{1}{1 + e^{\beta(1-z)}} \tag{4.6}$$

Esta función introduce un nuevo parámetro en el entrenamiento, la β de la función que varía su pendiente. A ésta hay que sumar dos más: μ y ν, que son valores de aprendizaje de los pesos de las clases y los prototipos, respectivamente.

Más detalles sobre este método y sobre la estimación de los pesos pueden consultarse en [22].

4.4. Experimentación con k-NN

Para evaluar esta técnica se ha partido de las características presentadas en la Sección 3.1. Se han incluido en el vector de datos toda la información extraída de cada segmento. La energía y el pitch, que son valores calculados cada 10 ms, se han reducido a 4 valores, así se iguala el número de valores de pitch y energía para todos los segmentos. Se han probado diversas combinaciones de parámetros en la estimación de los pesos. Básicamente se ha variado el número de iteraciones de entrenamiento y se han establecido diferentes valores de aprendizaje de los pesos de las clases μ y los prototipos ν. El valor de la β de la sigmoide se ha mantenido a 8. Así mismo, se han clasificado las muestras de test haciendo uso de diferentes combinaciones de los pesos estimados. Se han probado también las dos maneras en las que el software CPW[1] permite inicializar los pesos: mediante la distancia euclídea y mediante la distancia de Mahalanobis [18] dependiente de la clase (Class-dependent Mahalanobis - CDM).

En la Tabla 4.1 aparecen los resultados más destacados de la batería de experimentos. La primera columna de error se refiere a los experimentos en los que se ajustan los pesos de las clases y las muestras; la segunda columna a los experimentos donde sólo se utilizan los pesos de las clases. Todos los resultados mostrados incluyen inicialización de los pesos mediante CDM. Lo más llamativo es el empeoramiento que se produce al utilizar únicamente los pesos asociados a las clases y que el error varía muy poco al alterar los parámetros de entrenamiento.

Tabla 4.1: Resumen de los resultados más significativos para la clasificación de segmentos utilizando k-vecinos. Los errores mostrados corresponden a entrenamientos de 1.000 iteraciones con una sigmoide de $\beta = 8{,}0$.

μ	ν	Error (peso w_{ci})	Error (peso w_c)
0,0005	0,0	37,09 %	37,09 %
0,0005	0,001	**35,85** %	41,82 %
0,0005	0,005	36,69 %	39,07 %
0,001	0,0	37,06 %	37,06 %
0,001	0,001	36,65 %	42,04 %
0,001	0,005	37,28 %	39,15 %

[1] http://www.dsic.upv.es/ rparedes/research/CPW/index.html

4.5. Clasificación basada en transcripción: HMM discretos

La otra forma de detección de actos de diálogo ha sido la utilización de HMM discretos. Para este caso ha sido necesario procesar la transcripción de los actos de diálogo para, por una parte, categorizarlo (etiquetando con un mismo símbolo todos los nombres de ciudades, tipos de trenes, ...) y además reescribirlo como etiquetas numéricas, asociando un número a cada una de las 709 palabras resultantes tras la categorización.

Utilizar la transcripción de los actos, o su reconocimiento en el caso de un sistema en producción, puede ser de utilidad para el módulo semántico, que trata de una forma u otra al acto en función de la estimación que se haya hecho de la clase a la que pertenece.

4.6. Experimentación con HMM discretos

Los parámetros a tener en cuenta en esta técnica son, además de la topología de los modelos de Markov, el tipo de entrenamiento que se realiza (Viterbi o Baum-Welch), el suavizado que se realiza en cada iteración sobre las probabilidades de emisión, o el umbral de parada, que detiene el entrenamiento cuando la variación de las estimaciones no lo supera.

Los modelos utilizados tienen 2 estados, salvo en el caso de las clases "Respuesta" e "Indefinida" que tienen 4 y la topología utilizada es la de un HMM lineal con bucles en los estados y transiciones a los dos estados siguientes. En la Tabla 4.2 se muestran los resultados de clasificación para diversos parámetros de entrenamiento.

Tabla 4.2: Resumen de resultados para la clasificación de segmentos utilizando HMM discretos. V indica entrenamiento por Viterbi y BW por Baum-Welch.

	Suavizado			
	0,00001		0,0001	
Umbral	V	BW	V	BW
0,001	11,55 %	11,84 %	12,21 %	12,87 %
0,005	11,58 %	11,58 %	12,13 %	12,46 %
0,01	12,17 %	11,58 %	12,76 %	12,50 %
0,1	12,02 %	**10,30 %**	12,65 %	10,88 %

El entrenamiento con Baum-Welch es el que produce los mejores resultados, resultados que son considerablemente mejores que los obtenidos con k-NN.

4.7. Combinación de clasificadores

En principio, dados los buenos resultados que se consiguen con la clasificación de las transcripciones, buscar una combinación de herramientas no parece tener mucho sentido. Sin embargo hay que tener en cuenta que en un sistema en producción el reconocimiento puede fallar y ese error se propagará, con total seguridad, a la detección de actos de diálogo.

Para poder combinar ambos clasificadores primero ha sido necesario establecer puntuaciones compatibles, es decir: el software de HMM discretos trabaja con log probabilidades enteras, mientras que el clasificador de k-vecinos en verdad sólo devuelve el vecino más cercano.

La puntuación de los modelos de Markov se ha establecido de una forma un tanto artesanal. Dado que los valores de log probabilidad son muy pequeños para representarse como reales, simplemente se han normalizado. Así la puntuación dada a la clase c es:

$$\Pr_h(c|x) = -\frac{\log(\Pr(c|x))}{\sum_{\forall c'} \log(\Pr(c'|x))} \tag{4.7}$$

Para puntuar las clases del clasificador de k-vecinos se ha utilizado un estimador típico que se suele utilizar como medida de confianza:

$$\Pr_k(c|x) = \frac{\frac{1}{d(x,x_c)}}{\sum_{c'=1}^{C} \frac{1}{d(x,x_{c'})}} \tag{4.8}$$

Ambas puntuaciones se encuentran entre 0 y 1 y se combinan linealmente regulando el aporte de cada una mediante un parámetro α:

$$\Pr(c|x) = \alpha \Pr_k(c|x) + (1-\alpha) \Pr_h(c|x) \tag{4.9}$$

En la Tabla 4.3 se muestran los errores para diferentes valores de α combinando los resultados del mejor experimento con k-NN y el mejor con HMMs. Se puede ver como el error evoluciona desde el obtenido con k-vecinos hasta el obtenido con HMM discretos. En este caso, dado que la diferencia entre ambos es tan acusada, la combinación no aporta nada a la clasificación.

Tabla 4.3: Resultados de la combinación de ambos clasificadores.

α	Error
1	35,85 %
0,7	25,73 %
0,6	22,10 %
0,5	18,55 %
0,4	15,25 %
0,3	13,27 %
0,2	12,21 %
0,1	11,36 %
0,05	10,78 %
0,02	10,48 %
0,01	10,48 %
0,001	10,37 %
0	10,30 %

4.8. Conclusiones

Hemos visto dos métodos de clasificación de actos de diálogo. El primero de ellos, basado en prosodia, es bastante prometedor, debido fundamentalmente a su invarianza entre idiomas y a la posibilidad de extraer información sin tan siquiera pasar la señal por un reconocedor. Es un método que puede aportar ventajas combinándose con un reconocedor del habla o, como en este trabajo, con otro reconocedor de actos basado en otras características.

Los resultados de la combinación en este caso no resultan espectaculares, debido a que se trabaja con transcripciones y esto provoca que el clasificador basado en palabras funcione muy bien, dejando poco margen para mejorar. Es de esperar una mejoría en la combinación al utilizar la salida de un reconocedor en vez de las transcripciones, pues esto elevará el error del clasificador basado en HMM discretos; probar y ratificar esta suposición es el siguiente paso. También se podría plantear la combinación de reconocedores diferentes, de manera que la opinión del clasificador basado en prosodia ayude al reconocimiento del habla.

Capítulo 5

Técnicas basadas en la señal

Partiendo de nuevo de la señal acústica, se han extraído características continuas de ella, como son la media de la señal y el contorno (el pitch). Estas dos características se explican con más detalle en el Capítulo 3. Para modelar estos datos se han utilizado modelos ocultos de Markov continuos, usando gaussianas como distribuciones de probabilidad de emisión. La técnica es similar a la utilizada en reconocimiento del habla con la salvedad de que, en vez de proporcionar vectores de características basados en cepstrales, se suministran vectores de pitch y energía de la señal.

5.1. Objetivos

En las pruebas de etiquetado de actos de diálogo, realizadas en el capítulo anterior, las características que se tomaban eran independientes del tiempo: el cálculo de la energía de la señal o del pitch se limitaba a cuatro números que promediaban todos los valores obtenidos. Para una clasificación basada en k-vecinos era adecuada, pero al hacer esto estamos perdiendo la información de evolución que podría ser determinante.

En el caso de la entonación, algunos lingüistas [24] proponen clasificaciones de enunciados en español según esta característica. Partiendo de esa idea, nuestra intención es comprobar si la variación de la frecuencia fundamental en el tiempo, acompañada de la energía, nos es útil a la hora de clasificar los actos de diálogo de primer nivel del usuario. Estos actos son: afirmación, negación, pregunta, respuesta, indefinida y cierre.

Además de utilizar estas características en solitario se busca poder combinarlas con n-gramas de actos de diálogo para tratar de conseguir mejoras significativas en el etiquetado.

5.2. Extracción de características y entrenamiento

Para obtener los vectores de características se procesa la señal en ventanas de 10 ms, obteniendo para cada una de ellas un valor de energía (el promedio) y la estimación de la frecuencia fundamental. Se calcula también la primera y segunda derivadas de estos valores, obteniendo vectores de 6 elementos. El valor de la energía sólo es útil en este caso particular, ya que todas las grabaciones se realizaron en condiciones similares, pero es una medida sujeta a fuertes variaciones en función de la distancia del

micrófono, calidad del equipo de grabación o ruido ambiental. Esta debilidad puede corregirse con normalización por la media, pero su utilización *online* es complicada y los resultados obtenidos en la experimentación *offline* no serían extrapolables a un sistema de diálogo real.

Con esos vectores de datos se entrenaron modelos de Markov de tres estados con distribuciones de emisión gaussianas en los estados. El número de estados vino determinado por la cantidad de ventanas que se obtienen de la señal más corta. Del mismo modo que en reconocimiento del habla se entrenan fonemas, aquí se entrenan actos de diálogo de primer nivel. El entrenamiento se llevó a cabo con HTK, hasta conseguir modelos con 128 gaussianas en los estados. Se realizaron experimentos parciales con menos gaussianas pero los resultados mostraron un mejor comportamiento del modelo con 128 gaussianas.

5.3. Reconocedor iATROS

Para la evaluación de los modelos se ha empleado el reconocedor iATROS. Es un sistema de reconocimiento automático del habla y de escritura basado en modelos morfológicos estocásticos (para voz y texto), modelos léxicos y modelos sintácticos. Ha sido desarrollado dentro del grupo Pattern Recognition and Human Language Technology, adscrito al Instituto Tecnológico de Informática de la Universidad Politécnica de Valencia.

El software iATROS incluye:

- Extracción de características acústicas: Obtención de vectores de características (coeficientes cepstrales) a partir de una señal grabada o desde un dispositivo de entrada.
- Preproceso de imágenes de texto manuscrito: Para poder realizar clasificación de texto es necesario extraer aquellas características de la imagen que permitan discriminar lo mejor posible los símbolos, las cuales se codifican finalmente como vectores de características.
- Decodificación: A partir de los modelos entrenados que definen el lenguaje a reconocer, se obtiene la secuencia de palabras más probable para los vectores de características dados.

El proceso de decodificación en habla se apoya en tres modelos:

- Modelos acústicos: cada uno de los fonemas o sonidos modelados son descritos con HMM continuos, donde la densidad de probabilidad de emisión de los vectores de características se asocia con los estados del HMM. La versión actual de iATROS trabaja con HMM compuestos de mixturas de gaussianas.
- Modelos léxicos: cada palabra se describe con un autómata de estados finitos, con modelos acústicos asociados en los arcos. Además, se soportan múltiples pronunciaciones en las palabras del léxico.
- Modelo de lenguaje: define las relaciones entre las palabras. Los modelos de lenguaje pueden ser autómatas de estados finitos o n-gramas de cualquier orden, lo que da la posibilidad de que los sistemas sean más flexibles ante las frases de la tarea que pueden ser aceptadas. iATROS permite decodificación parcial, de modo que el reconocimiento pueda finalizar antes de alcanzar un estado final en el modelo de lenguaje.

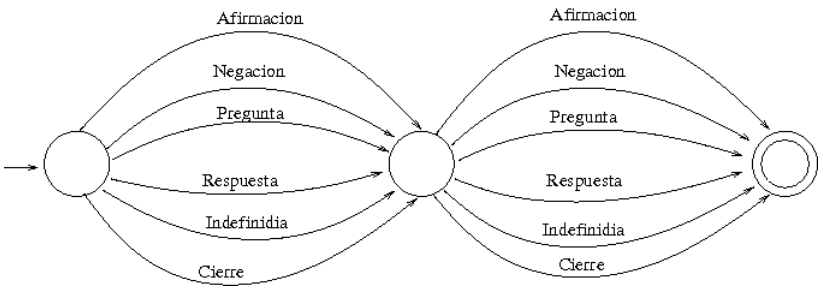

Figura 5.1: Autómata de estados finitos utilizado como modelo de lenguaje para la tarea.

La decodificación en iATROS se basa en el algoritmo de Viterbi, en el que se aplican técnicas de ”beam search” para reducir el campo de búsqueda de la hipótesis más probable.

Adaptación de la tarea

El software de reconocimiento está enfocado principalmente hacia el habla y la escritura; sin embargo, es sencillo adaptarlo a otras tareas. En este caso se han simplificado los modelos anteriores para ajustarlos a las necesidades del problema.

Ahora cada ”fonema” del modelo acústico es uno de los seis posibles actos de diálogo. Los modelos acústicos se han entrenado utilizando el software HTK.

Los modelos léxicos no son más que una ampliación de los acústicos, una capa por encima de ellos. Cada palabra del modelo (cada acto) remite directamente a su modelo acústico.

El modelo de lenguaje se diseñó a mano y es un autómata de estados finitos con tres estados. Las transiciones entre estados representan los posibles actos de diálogo; todos los actos tiene la misma probabilidad. En la Figura 5.1 puede verse el modelo empleado. Sólo se permiten dos actos de diálogo, ya que no existe ningún turno que contenga más de dos actos de primer nivel. Para permitir que el reconocedor ofrezca como respuesta un solo acto se activó la opción de decodificación parcial.

Grafos de palabras

Utilizamos el grafo de palabras para poder añadir al reconocimiento información extra que no podemos integrar directamente en el proceso de decodificación. Un grafo de palabras G es un grafo dirigido, acíclico y con pesos. Los nodos del grafo corresponden a puntos discretos en el tiempo. Las aristas del grafo son tripletas $[w, s, e]$ donde w es la palabra hipotética del nodo s al nodo e. Los pesos son puntuaciones asociadas a las aristas del grafo de palabras. El mejor camino que se forma desde el estado inicial hasta el estado final es la hipótesis más probable [25].

La construcción del grafo de palabras se realiza una vez finalizado el reconocimiento. Para generarlo se parte de las n mejores palabras finales (las n más probables) y se reconstruyen los caminos que las han generado, así como otros alternativos. De esta forma no sólo se recuperan los n mejores caminos, sino que aparecen otros alternativos que durante el proceso de decodificación se han descartado. El grafo de palabras

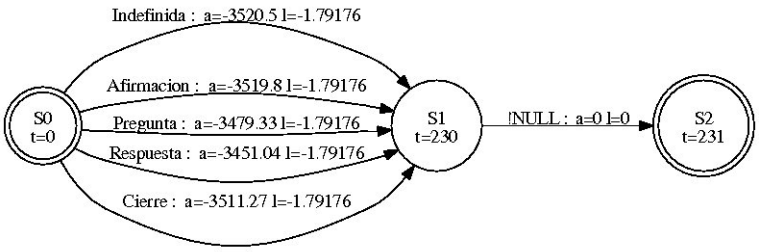

Figura 5.2: Ejemplo de grafo de palabras para la tarea. Cada arista va acompañada de su etiqueta y de la log-probabilidad del modelo acústico (a) y del modelo de lenguaje (l).

puede ser tan completo como el trellis del reconocimiento, aunque iATROS permite especificar mediante parámetros cuánta información queremos mantener.

En definitiva, el grafo de palabras es como una ”foto” del reconocimiento. En la Figura 5.2 hay un ejemplo de grafo de palabras de la tarea. En este caso sólo tenemos un acto de diálogo y el grafo nos muestra las probabilidades de cada clase.

5.4. Resultados y conclusiones

Las pruebas de clasificación de actos de diálogo se dividen en tres etapas: en la primera se prueba la clasificación basada únicamente en los parámetros propuestos (pitch y energía); la segunda es una estimación de la clasificación utilizando sólo trigramas inferidos a partir del conjunto de entrenamiento, teniendo en cuenta para este caso los actos de diálogo del sistema; la última etapa es la combinación de la clasificación basada en prosodia junto con el trigrama de actos de diálogo.

La clasificación mediante prosodia se realizó utilizando el autómata de estados finitos ya presentado y activando la decodificación parcial del reconocedor. Esta decodificación generó los grafos de palabra correspondientes que sirvieron de base para la clasificación utilizando n-gramas. Debido al pequeño tamaño del modelo de lenguaje, el grafo de palabras puede recoger todo el trellis del reconocimiento.

Partiendo de los grafos y aprovechando los cálculos parciales de probabilidades que adjuntan, se realizó una búsqueda del camino óptimo sobre cada uno de ellos, pero aplicando la probabilidad del trigrama. El n-grama no se integró en el reconocimiento original puesto que no se podían entrenar modelos para los actos de diálogo del sistema e incluirlos dentro de los modelos de iATROS; a pesar de eso, se pretendía aprovechar la información que ofrece conocer el acto anterior aunque sea del sistema, algo que en un sistema de diálogo siempre se conoce.

La experimentación sigue un modelo de validación cruzada sobre cinco particiones. La Tabla 5.1 muestra un promedio sobre estas particiones de los conjuntos de entrenamiento y test.

Se ha estimado el error a nivel de actos de diálogo (DAER, Dialog Act Error Rate) y el error a nivel del turno completo (TER, Turn Error Rate). El DAER es similar al Word Error Rate, pero con actos de diálogo; la secuencia de actos obtenida se compara con la secuencia de referencia, contando el número de inserciones (I), borrados (B) y substituciones (S), necesarios para transformar la primera en la segunda, incluyendo

Tabla 5.1: Estadísticas del corpus DIHANA (media de las cinco particiones para validación cruzada).

	Entrenamiento			Test		
	Usuario	Sistema	Total	Usuario	Sistema	Total
Diálogos	720			180		
Turnos	5,024	7,206	12,330	1,256	1,827	3,083
Palabras totales	42,806	119,807	162,613	10,815	29,950	40,765
Vocabulario	762	208	832	417	174	485

los actos correctos (C). El DAER se calcula como $\frac{I+B+S}{C+D+S}$. El TER indica el número de turnos que han sido etiquetados correctamente.

Los resultados de todos los experimentos se encuentran en la Tabla 5.2. Como se puede ver, los resultados de clasificación utilizando únicamente la información de prosodia son muy pobres. Utilizar únicamente el trigrama da resultados diez puntos mejores. Combinar ambos produce una pequeña ganancia de un punto en el DAER.

Tabla 5.2: Resultados de decodificación con prosodia y n-gramas. La primera columna muestra resultados sólo con prosodia, la segunda sólo utilizando n-gramas y la última la combinación de ambos clasificadores.

DAER/TER	Prosodia	3-grama	Combinado
Partición 1	50,9/48,4	41,4/40,2	40,5/39,3
Partición 2	59,7/59,6	43,5/42,2	42,8/41,0
Partición 3	56,0/55,2	40,8/39,0	39,8/37,6
Partición 4	53,4/52,3	40,9/39,5	40,6/39,4
Partición 5	43,4/41,6	34,5/33,2	32,4/30,7
Total	52,8/51,5	40,3/38,9	**39,3/37,6**

Los resultados de etiquetado utilizando únicamente prosodia mejoran el *baseline* (recordemos que para etiquetado se sitúa en torno al $60\,\%$ de error), pero no de manera significativa. La combinación con el trigrama, que da mejores resultados que sólo la prosodia, no produce tampoco diferencias notables en el error. No se han probado con actos de segundo o tercer nivel dado que éstos no presentan, a priori, entonaciones propias. Como trabajo futuro queda la búsqueda de un corpus que esté anotado teniendo en cuenta rasgos lingüísticos como la entonación. Esto nos permitiría evaluar mejor este método y comprobar hasta qué punto la información de prosodia aquí utilizada puede ayudar a los sistemas de diálogo.

Capítulo 6

Modelo del número de segmentos

En este capítulo se presenta un modelo de etiquetado de segmentos basado en la transcripción. La novedad del modelo radica en la inclusión de una estimación del número de segmentos que contiene el turno a etiquetar. Se presenta un modelo de etiquetado general que se concreta en otros dos: uno no tienen en cuenta el número de segmentos estimado y otro sí. Se incluye también una propuesta para la estimación del número de segmentos del turno.

6.1. Etiquetado de diálogos basado en HMM

Dada una secuencia de palabras $\mathcal{W}$ obtenida a partir del reconocimiento de la voz de un sistema de diálogo, el objetivo es el de obtener la secuencia óptima de actos de diálogo $\widehat{\mathcal{U}}$ que maximice la probabilidad a posteriori $\Pr(\mathcal{U}|\mathcal{W})$. Expresado formalmente:

$$\widehat{\mathcal{U}} = \underset{\mathcal{U}}{\operatorname{argmax}} \Pr(\mathcal{U}|\mathcal{W}) = \underset{\mathcal{U}}{\operatorname{argmax}} \Pr(\mathcal{U}) \Pr(\mathcal{W}|\mathcal{U}) \tag{6.1}$$

donde $\Pr(\mathcal{U})$ representa la probabilidad a priori de una secuencia de actos de diálogo y $\Pr(\mathcal{W}|\mathcal{U})$ es la probabilidad de la secuencias de palabras dados los actos de diálogo

La secuencia de actos de diálogo $\mathcal{U}$ del diálogo completo puede restringirse hasta el turno actual t, donde tendremos $U_1^{t-1} = U_1 \cdot U_2 \cdots U_{t-1}$, que representa la secuencia de actos de diálogo hasta el turno actual t. La secuencias de palabras del turno actual se expresa como $W = W_1^l = w_1 \cdot w_2 \cdots w_l$, donde l es el número de palabras de W. Por lo tanto, se puede reformular el problema introduciendo una nueva probabilidad a posteriori $\Pr(U|W_1^l, U_1^{t-1})$, que representa la probabilidad de la secuencia de actos de diálogo U que está asociada al turno de usuario actual, dadas la secuencia de palabras del turno actual W_1^l y la historia de las secuencias anteriores de actos de diálogo U_1^{t-1}. El objetivo ahora es encontrar la mejor secuencia de actos de diálogo para cada turno, de forma que en cada t buscamos:

$$\widehat{U} = \underset{U}{\operatorname{argmax}} \Pr(U|W_1^l, U_1^{t-1}) \tag{6.2}$$

Ahora se pueden introducir dos variables *ocultas*: el número de segmentos r y la segmentación del turno, que se puede describir como $s = (s_0, s_1, \ldots, s_r)$. De esta forma, U se puede expresar como $U = u_1^r$, y W como $W_1^l = W_{s_0+1}^{s_1} W_{s_1+1}^{s_2} \ldots W_{s_{r-1}+1}^{s_r}$.

A partir de la Ecuación (6.2) se pueden derivar dos modelos. La asunción usual es que la segmentación s y el número de segmentos r son desconocidos (permanecen ocultos) y no tienen influencia en la asignación de actos de diálogo. En este caso se puede expresar la probabilidad de la secuencia de actos de diálogo como:

$$\Pr(U|W_1^l, U_1^{t-1}) = \Pr(U|U_1^{t-1}) \Pr(W_1^l|U, U_1^{t-1}) = \sum_{r,s_1^r} \prod_{k=1}^{r} \Pr(u_k|u_1^{k-1}, U_1^{t-1}) \Pr(W_{s_{k-1}+1}^{s_k}|u_1^k, U_1^{t-1}) \quad (6.3)$$

Este modelo se simplifica con tres asunciones básicas: la probabilidad de las palabras del segmento actual depende sólo del acto de diálogo actual; la probabilidad del acto de diálogo depende únicamente de los n actos de diálogo previos; y la suma es reemplazada por una maximización. El modelo resultante es el siguiente:

$$\Pr(U|W_1^l, U_1^{t-1}) = \max_{r,s_1^r} \prod_{k=1}^{r} \Pr(u_k|u_{k-n-1}^{k-1}) \Pr(W_{s_{k-1}+1}^{s_k}|u_k) \quad (6.4)$$

Este modelo puede utilizarse cuando hay disponible una segmentación (y se conoce el número de segmentos r) eliminando la maximización y el productorio y fijando los valores s_k y r a aquellos provistos por la segmentación. Si no hay segmentación disponible, la búsqueda de la secuencia de actos de diálogo óptima produce una segmentación que permite obtener la máxima probabilidad. De este modo, se obtiene una segmentación derivada de este método. Este modelo puede ser considerado como el *baseline* de la tarea.

Se puede desarrollar otro modelo partiendo de la Ecuación (6.2) asumiendo que el número de segmentos r tiene influencia en el etiquetado. En este caso, la probabilidad de la secuencia U es:

$$\Pr(U|W_1^l, U_1^{t-1}) = \sum_r \Pr(U, r|W_1^l, U_1^{t-1}) =$$

$$= \sum_r \Pr(r|W_1^l, U_1^{t-1}) \prod_{k=1}^{r} \Pr(u_k|u_1^{k-1}, r, U_1^{t-1}) \Pr(W_{s_{k-1}+1}^{s_k}|u_1^k, r, U_1^{t-1}) \quad (6.5)$$

Esta expresión se puede simplificar utilizando las mismas asunciones que para obtener la Ecuación (6.4). Así, el nuevo modelo de etiquetado es:

$$\Pr(U|W_1^l, U_1^{t-1}) = \max_r \Pr(r|W_1^l, U_1^{t-1}) \max_{s_1^r} \prod_{k=1}^{r} \Pr(u_k|u_{k-n-1}^{k-1}) \Pr(W_{s_{k-1}+1}^{s_k}|u_k) \quad (6.6)$$

Igual que en el modelo anterior, se obtiene una segmentación de esta ecuación.

En las Ecuaciones (6.4) y (6.6), $\Pr(u_k|u_{k-n-1}^{k-1})$ puede ser modelada como una n-grama (de grado n) y $\Pr(W_{s_{k-1}+1}^{s_k}|u_k)$ se puede modelar como un HMM. Resaltar

que, en esta fórmula, u_{k-n-1}^{k-1} puede dar cuenta de los actos de diálogo de los turnos previos.

Se han derivado dos modelos de etiquetado a partir de la Ecuación (6.2). El modelo descrito en la Ecuación (6.4) no contiene ningún tipo de información sobre el número de *utterances* del turno, ni sobre la segmentación. El modelo presentado (6.6) incluye la estimación de la probabilidad del número de segmentos.

Para estimar la probabilidad $\Pr(r|W_1^l, U_1^{t-1})$, las dependencias de r se substituyen por una puntuación S_c definida sobre la secuencia de palabras W_1^l. La probabilidad es reformulada como $\Pr(r|S_c)$. La puntuación S_c se explica en la Sección 6.3.

6.2. Características usadas

A continuación se presentan diferentes características estudiadas y se discute su interés para el modelo en el cálculo de S_c Para el estudio se han tomado todos los diálogos del corpus. Cada segmento representa un acto de diálogo de tercer nivel.

6.2.1. Longitud del turno

No es descabellado pensar que a mayor duración del turno, más posibilidades hay de que éste contenga diferentes actos de diálogo. Para tratar de corroborar esta intuición se han medido las longitudes de los turnos de sistema y usuario en palabras.

La Figura 6.1 muestra las distribuciones de probabilidad de la longitud de los turnos. Cada distribución normal representa un número de segmentos desde uno hasta ocho. Como se puede apreciar, sí que existe cierta relación entre el número de palabras y el número de segmentos. Aunque los solapamientos descartan descansar sobre esta característica el peso del modelo, sí que conviene tenerla en cuenta.

6.2.2. Número de intervención

Un factor que puede influir considerablemente en el número de actos de un turno es la posición del turno dentro del diálogo. Es posible que las primeras intervenciones de los usuarios tengan más segmentos que las finales, o viceversa.

La Figura 6.2 muestra la gráfica del número de segmentos del turno actual en función del orden del turno de usuario en el diálogo. No se aprecia ninguna relación clara. En todo caso, los turnos con sólo un segmento o dos segmentos son ligeramente más frecuentes en intervenciones no iniciales. También cabe destacar que aquellos segmentos finales de diálogos muy largos, con más de 15 intervenciones del usuario, contienen, en su mayoría, un único segmento; sin embargo únicamente 29 de los 6.280 turnos han sido realizados en esas condiciones.

6.2.3. Segmentos del turno anterior

Planteamos la posibilidad de que el número de segmentos del turno esté relacionado con los segmentos del turno anterior. De nuevo hemos centrado nuestra atención en los turnos de usuario, y en cómo los segmentos de éstos pueden depender del turno anterior, siendo éste también de usuario o del sistema. En la Figura 6.3 se incluye la gráfica comparativa de los segmentos del turno actual en comparación a los segmentos del turno anterior. La gráfica no muestra ninguna relación directa.

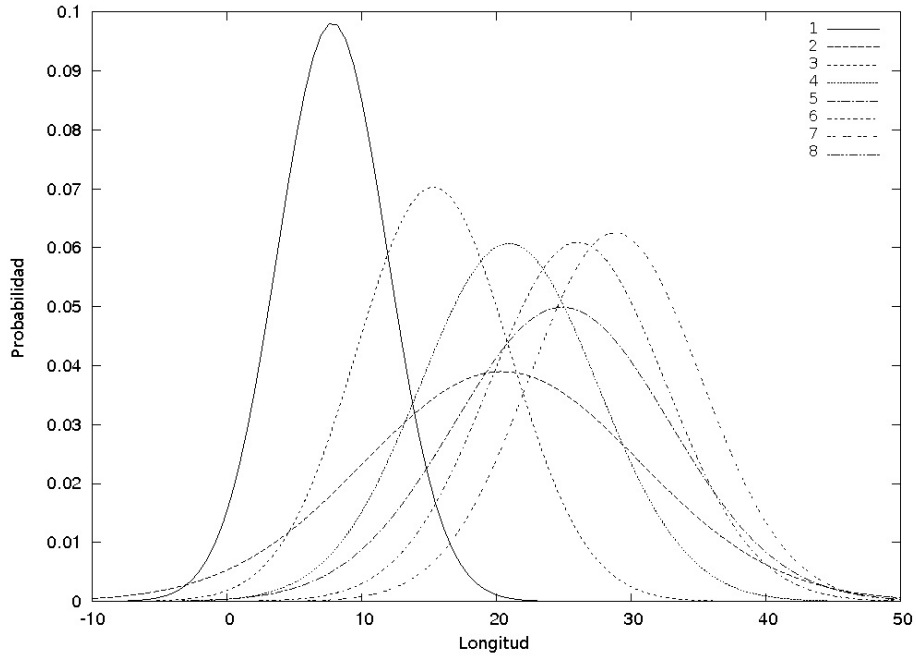

Figura 6.1: Distribuciones de probabilidad del número de segmentos de un turno en función de su longitud en palabras.

6.2.4. Palabras clave

Dado que la información estructural del diálogo no aporta mucha información, nos centramos ahora en la información léxica. La intuición nos dice que algunas palabras pueden ser en muchos casos claros indicadores de fin de segmento. En el corpus transcrito, por ejemplo, el punto final además del fin de frase indica también un fin de segmento.

Las intervenciones de usuario contienen en total 9.715 segmentos y 705 palabras de vocabulario. De ellas 75 indican, en al menos una ocasión, un fin de segmento. De las 53.462 palabras que componen las intervenciones de los usuarios, en 16.221 ocasiones una palabra fin de segmento no lo está marcando, esto es, en un $70\,\%$ de los casos las palabras fin de segmento marcan correctamente el final de un segmento. Con estos datos parece que el hecho de seleccionar un grupo de palabras que marquen el fin de segmento puede ser una buena característica con la que estimar el número de etiquetas de un turno.

6.3. Modelo para la estimación del número de segmentos

Para poder estimar la probabilidad del número de segmentos de un turno se asocia a cada turno una puntuación S_c, cuya obtención se muestra en el apartado 6.3.2. A continuación se presenta el modelo que utiliza S_c para obtener la probabilidad del

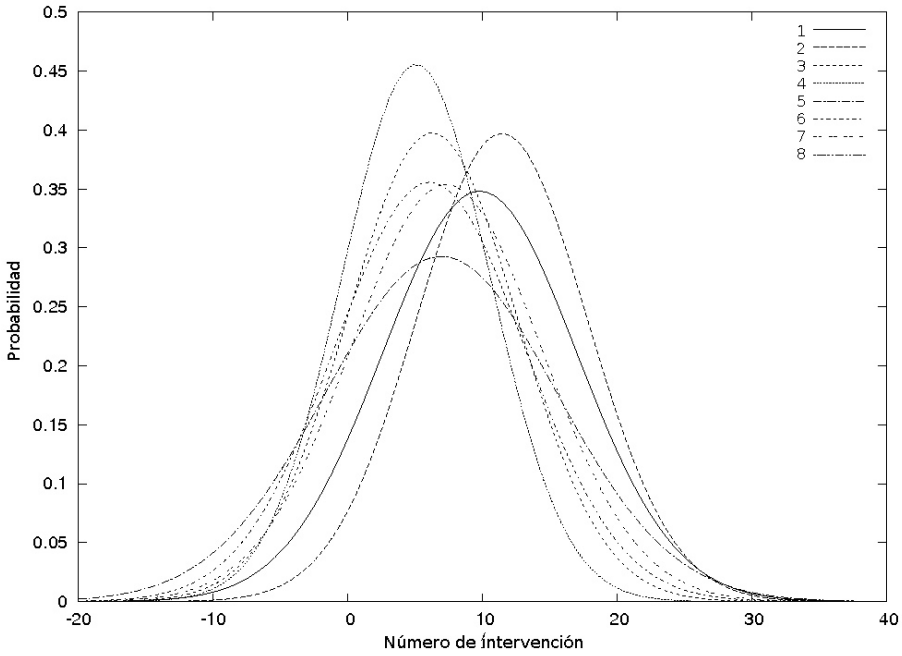

Figura 6.2: Segmentos de un turno en función del número de intervención.

número de segmentos.

6.3.1. Estimación de la probabilidad

En la Sección 6.1 se introduce una aproximación a la estimación del número de segmentos del turno; esto es, se define una puntuación S_c asociada con cada turno, que se calcula a partir de la transcripción. Para estimar la probabilidad del número de segmentos, se elige la aproximación $\Pr(r|W_1^l, U_1^{t-1}) = \Pr(r|S_c)$, donde S_c se calcula a partir de la secuencia de palabras W_1^l.

Esta nueva probabilidad que hemos definido se puede calcular aplicando Bayes:

$$\Pr(r|S_c) = \frac{p(S_c|r)p(r)}{p(S_c)} \tag{6.7}$$

La probabilidad a priori $p(r)$ se puede calcular fácilmente como el número turnos con r segmentos, N_{Tr}, dividido por el número total de turnos N_T:

$$p(r) = \frac{N_{Tr}}{N_T} \tag{6.8}$$

La probabilidad condicional $p(S_c|r)$ se estima mediante una distribución normal. Se calcula una distribución para cada r:

$$p(S_c|r) \sim \mathcal{N}(m_r, \sigma_r) \tag{6.9}$$

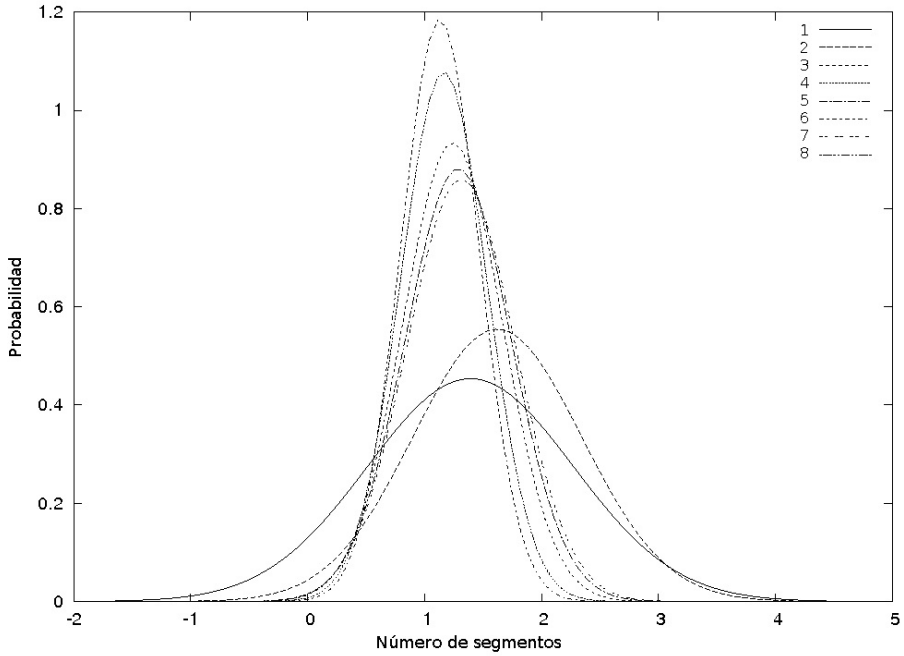

Figura 6.3: Segmentos de un turno actual en función del número de segmentos del turno anterior.

La media m_r y la varianza σ_r se calculan a partir de las puntuaciones asociadas con los turnos de r segmentos.

El último elemento es estimado utilizando otra distribución gaussiana, calculada a partir de todos los turnos:

$$p(S_c) \sim \mathcal{N}(m_{Sc}, \sigma_{Sc}) \tag{6.10}$$

La media y la varianza se calculan sobre todas las puntuaciones de los datos de entrenamiento.

6.3.2. Cálculo de la puntuación

Para puntuar cada turno y obtener su S_c asociado se pueden seguir diferentes estrategias, aunque todas ellas se apoyan en dos de las características vistas anteriormente: la longitud del turno y las palabras finales de segmento.

- Puntuación basada en el número de palabras

  Se puede estimar S_c como la longitud del turno:

$$S_c(W) = l \tag{6.11}$$

- Palabras frontera

  Se define la puntuación S_c de un turno W como:

$$S_c(W) = \sum_{i=1}^{l} p_f(w_i) \tag{6.12}$$

  donde $p_f(w_i)$ es la probabilidad de la palabra w_i de ser la última palabra de un segmento. Se estima contando el número de veces que la palabra es final de segmento dividida por el número de apariciones de la palabra. Este valor es cero para aquellas palabras que nunca aparecen como final de segmento.

  También es posible calcular S_c de la misma forma pero centrándonos en las palabras del inicio del segmento, en vez de en las finales.

- N-gramas frontera

  En vez de calcular la probabilidad de una palabra final, se puede extender el planteamiento inicial y estimar la probabilidad de que n palabras aparezcan al final de un segmento. En este caso, el método de estimación es el mismo que el utilizado en el supuesto anterior: el número de veces que un n-grama es final, dividido por el total de apariciones del n-grama. Se calcula S_c de esta manera con:

$$S_c(W) = \sum_{i=n}^{l} p_f(W_{i-(n-1)}^{i}) \tag{6.13}$$

  Igual que se propone para la estimación de palabras finales, la probabilidad de n-gramas iniciales puede ser calculada simplemente contando las veces que el n-grama es inicial.

- Puntuación combinada

  Las características utilizadas en la estimación de la puntuación pueden ser combinadas de dos maneras diferentes: creando una puntuación a partir de varias características o con una combinación naive-Bayes.

  En el primer método, la puntuación calculada para un turno está compuesta de varias características, por ejemplo, la puntuación puede ser vista como la suma de la probabilidad de que cada palabra sea final más la longitud del turno (añadiendo un número a por cada palabra):

$$S_c(W) = \sum_{i=1}^{l} (p_f(w_i) + a) \tag{6.14}$$

  Otra opción es combinar las palabras finales con los n-gramas, por ejemplo, combinando bigramas finales y palabras finales:

$$S_c(W) = \sum_{i=2}^{l} p_f(W_{i-1}^i) + \sum_{i=1}^{l} p_f(w_i) \tag{6.15}$$

En el segundo método, la probabilidad final del número de segmentos se calcula combinando las probabilidades de cada puntuación, por ejemplo, si consideramos:

$$\Pr(r|S_{c_1}, S_{c_2}, \cdots S_{c_n})$$

esta probabilidad puede simplificarse asumiendo que no hay dependencias entre puntuaciones (asunción naive-Bayes):

$$\begin{aligned} &\Pr(r|S_{c_1}, S_{c_2}, \cdots S_{c_n}) = \\ &\Pr(r|S_{c_1}) \Pr(r|S_{c_2}) \cdots \Pr(r|S_{c_n}) \end{aligned} \tag{6.16}$$

6.4. Experimentos y resultados

Los experimentos se han dividido en dos fases: la primera está dedicada a evaluar la estimación del número de segmentos y la segunda combina esta estimación con el modelo de etiquetado general. Las pruebas se realizan sobre las cinco particiones definidas para la experimentación desarrollada en el Capítulo 5.

Los experimentos de etiquetado se han realizado con actos de diálogo de tercer nivel y se han tenido en cuenta los diálogos completos, contando los turnos de sistema y usuario.

6.4.1. Estimación del número de segmentos

Las primeras pruebas realizadas tienen como finalidad probar la capacidad de los modelos propuestos para estimar el número de segmentos de un turno. La Tabla 6.1 muestra el porcentaje del número de turnos según los segmentos. El $68\,\%$ de los turnos tiene un único segmento, por lo que un estimador naive del número de segmentos que estimara para todos los turnos un único segmento estaría cometiendo un error del $32\,\%$.

Tabla 6.1: Número de turnos y su porcentaje en función del número de segmentos que contienen.

Número de segmentos	Turnos	Porcentaje
1	4274	68 %
2	1129	18 %
3	517	8,2 %
4	226	3,6 %
5	94	1,5 %
6	23	0,4 %
7	16	0,25 %
8	1	0,016 %

La Tabla 6.2 recoge los resultados de las estimaciones propuestas en la sección anterior, suponiendo que se toma como correcto el número de segmentos que consigue mayor probabilidad. También incluye un par de estimaciones basadas en los dos modelos de combinación de puntuaciones. Algunos métodos como las palabras o trigramas finales, así como las palabras iniciales, superan el error del estimador naive.

Tabla 6.2: Resultados de la estimación del número de segmentos. La columna "Estimación" indica el tipo de puntuación utilizada en la estimación de r. El error indica el porcentaje de turnos en los que el estimador ha fallado.

Estimación	Error
Longitud	17.9
Palabras finales	35.0
Bigramas finales	**9.5**
Trigramas finales	47.4
Palabras iniciales	39.2
Bigramas iniciales	13.4
Trigramas iniciales	14.1
Puntuación compuesta por longitud y palabras finales	17.7
Combinación Naive-Bayes de la longitud y las palabras finales	20.2

6.4.2. Etiquetado de los turnos

En esta sección se van a comparar los dos modelos de etiquetado definidos en la Sección 6.1. Primero se establece un *baseline* para la tarea basándonos en el etiquetado de la Ecuación (6.4). Estos resultados vienen avalados por lo publicado en [19]. Sobre el *baseline* se comparan los resultados obtenidos con el etiquetado de la Fórmula 6.6.

Baseline

En la Sección 6.1 se han definido dos modelos de etiquetado basados en HMMs. Los experimentos para medir el *baseline* utilizan el modelo representado por la Ecuación (6.4).

La Tabla 6.3 muestra los resultados utilizando 2-gramas y 3-gramas para la estimación de la probabilidad $\Pr(u_k|u_{k-n-1}^{k-1})$. También muestra una comparación del error de etiquetado entre las versiones segmentadas y no segmentadas del corpus. En la versión segmentada se conoce la segmentación correcta, pero en la no segmentada no se tiene ninguna información acerca de la segmentación o del número de segmentos.

Estos resultados establecen fronteras de errores. Los turnos segmentados ofrecen un error mínimo de los modelos basados en HMM. Los turnos no segmentados ofrecen un valor máximo de error, obtenido sin conocer la segmentación. Se considera que el resultado obtenido con la versión no segmentada y con un 3-grama es el error *baseline* (17.0 % de DAER).

Tabla 6.3: DAER para los modelos descritos mediante la Ecuación (6.4). La primera línea indica el uso de turnos segmentados en el etiquetado. La segunda línea indica el uso de turnos no segmentados. En negrita el resultado que se considera baseline de la tarea.

	2-gram	3-gram
Segmentado	10.8	10.3
No segmentado	17.8	**17.0**

Etiquetado con el número de segmentos

El último conjunto de experimentos muestra el etiquetado de turnos producido por el modelo matemático presentado en la Ecuación (6.6), donde se introduce una estimación de la probabilidad del número de segmentos. Debido a los resultados de la estimación de r, se han seleccionado diferentes estimaciones; en concreto se han utilizado unigramas, longitud del turno, bigramas fin de segmento y bigramas inicio de segmento. Se ha probado el etiquetado con 2-gramas y 3-gramas como estimadores de la probabilidad $\Pr(u_k|u_{k-n-1}^{k-1})$.

La Tabla 6.4 muestra una comparación de los errores obtenidos en los experimentos. El error con la estimación correcta de r está calculado a partir de la versión no segmentada del corpus, conociendo el número correcto de segmentos ($\Pr(r|S_c)$ es 1 para el r correcto y 0 para el resto). El resto de las líneas se refieren a las diferentes estimaciones del número de segmentos. Los resultados del etiquetado con r conocida suponen una nueva cota inferior, que substituye al error del etiquetado con el corpus segmentado.

Tabla 6.4: DAER y TER para el etiquetado utilizando la estimación de segmentos y diferentes n-gramas para estimar $\Pr(u_k|u_{k-n-1}^{k-1})$. Cada línea hace referencia una estimación diferente de la probabilidad del número de segmentos. Se incluye el error de etiquetado y un intervalo de confianza del 95 % para la diferencia entre el DAER y el DAER del baseline.

	2-gram		3-gram	
Estimación de r	DAER/TER	C.I.	DAER/TER	C.I.
r correcta	12.8/12.9	[3,1; 5,4]	12.1/12.2	[3,9; 6,0]
Longitud	15.5/14.9	[0,3; 2,9]	15.0/14.2	[0,86; 3,17]
Unigrama FDS	15.9/15.2	[−0,1; 2,4]	**14.9/14.4**	[0,9; 3,3]
Bigramas FDS	16.4/15.8	[−0,6; 1,9]	15.6/15.0	[0,3; 2,7]
Bigramas IDS	17.2/16.3	[−1,4; 1,1]	16.2/15.4	[−0,4; 2,0]

El mejor resultado se obtuvo con la estimación del número de segmentos basado en palabras finales y la probabilidad de los actos de diálogo dado por una 3-grama. El análisis estadístico del mejor resultado obtenido con este modelo y el resultado *baseline* (obtenido con el corpus sin segmentar y el modelo previo) muestra que la diferencia entre las medias de los errores de ambos experimentos se encuentra en el intervalo [0.9 ; 3.3] (con una confianza del 95 %). Por lo tanto, se puede concluir que el modelo

con la estimación de la probabilidad del número de segmentos produce una mejora significativa en el etiquetado.

Los errores de etiquetado muestran que no existe ninguna relación entre el error de la estimación del número de segmentos y los de etiquetado. Esto se debe a la dificultad de etiquetar correctamente algunos turnos que no son etiquetados correctamente en ninguno de los experimentos, incluso cuando se suministra el número correctos de segmentos. Esto es debido a que la estimación del número de segmentos no es suficiente para eliminar los errores del etiquetador. Para comprobar el origen de este comportamiento se han calculado los valores de precisión, recall y F-measure de los experimentos.

Tabla 6.5: *Precision*, *recall* y F-measure del etiquetado. Se incluyen los resultados del error de *baseline* de etiquetado (sin estimaciones), el error de etiquetado conociendo r y el error de etiquetado utilizando bigramas fin de segmento y unigramas fin de segmento.

	2-gram			3-gram		
Estimación de r	Precision	Recall	F	Precision	Recall	F
Sin estimación	0.84	0.89	0.86	0.85	0.90	0.87
r correct	0.88	0.88	0.88	0.88	0.88	0.88
Unigramas FDS	0.86	0.88	0.87	0.87	0.88	0.88
Bigramas FDS	0.86	0.87	0.86	0.87	0.88	0.87

La precisión se calcula dividiendo el número de segmentos bien etiquetados entre el total de etiquetas dadas por el etiquetador. El recall se calcula dividiendo el número de segmentos bien etiquetados entre el número correcto de segmentos. La F-measure se calcula como $F = 2 \cdot (\text{precision} \cdot \text{recall})/(\text{precision} + \text{recall})$.

La Tabla 6.5 muestra la precisión, el recall y la F-measure de algunos experimentos. La precisión indica el acierto del etiquetador, pero la posición de las etiquetas no se tienen en cuenta, por eso estos errores son un poco mejores que su correspondiente DAER. La precisión es similar para todos los experimentos, lo que significa que los errores están producidos por el etiquetador, incluso cuando se suministra el número correcto de segmentos. Los resultados también muestran la mejora producida por la inclusión de la probabilidad del número de segmentos en el etiquetado.

6.5. Conclusiones

Como ha quedado demostrado, la inclusión de una estimación del número de segmentos en la formulación del etiquetador produce mejoras significativas en el etiquetado, aunque los resultados están lejos de igualar al etiquetado conociendo la segmentación.

Aunque las estimaciones del número de segmentos son bastante buenas (la mejor estimación da un $9{,}5\,\%$ de error), esta bondad no se traslada en la misma medida al etiquetado. De hecho el estimador que produce el mejor etiquetado es precisamente el que peor estima el número de segmentos. Esta diferencia es comprensible dado que la estimación del número de segmentos se traduce en el modelo de etiquetado como una probabilidad, un elemento más de la fórmula, mientras que para estimar el número de

segmentos simplemente se devuelve el número de segmentos de mayor probabilidad.

El hecho de que los valores de precision y recall sean similares en todos los experimentos da a entender que el etiquetado está fallando en todos los casos. Conocer el número correcto de segmentos produce una mejora en el DAER (que tiene en cuenta la posición de la etiqueta en el turno), pero no influye en qué etiquetas se eligen. Con la inclusión del número de segmentos se ajustan mejor las etiquetas que hay por turnos, pero estas etiquetas no son necesariamente las correctas.

Capítulo 7

Segmentación y etiquetado basado en GIATI

En el Capítulo 6 se ha planteado en la Ecuación (6.4) un modelo matemático de etiquetado basado en HMMs que asumía que el número de segmentos del turno y su segmentación eran desconocidos. Sin embargo, este modelo podía fácilmente adaptarse si conocíamos la segmentación correcta.

En este capítulo se presenta una técnica de segmentación basada en n-gramas. La técnica se basa en la inferencia de Transductores Estocásticos de Estados Finitos, utilizada principalmente en Traducción Automática.

7.1. N-grama Transductora

La segmentación de turnos en segmentos puede verse desde la perspectiva de la traducción automática. El objetivo es ”traducir” una serie de palabras por una palabra que represente el corte del turno. Para abordar la segmentación desde este punto de vista se puede utilizar una búsqueda, apoyada en el algoritmo de Viterbi, basada en un n-grama inferido a partir de muestras de entrenamiento.

La inferencia del n-grama se realiza utilizando la técnica de inferencia de Transductores Estocásticos de Estados Finitos (TEEF) GIATI[1]. GIATI es una técnica de inferencia basada en un proceso de re-etiquetado sobre un par de frases, una de entrada (lenguaje origen) y otra de salida (lengua destino). El re-etiquetado depende de los alineamientos realizados entre los símbolos de entrada y los de salida, como se puede ver en la Figura 7.1. Partiendo del corpus re-etiquetado se infiere un n-grama suavizado que se transforma finalmente en un TEEF.

Al aplicar esta técnica a la segmentación de turnos no es necesario este último paso y podemos utilizar directamente el n-grama, ahorrándonos así los problemas derivados de la transformación de los n-gramas suavizados a TEEFs. A esta n-grama se le llama n-grama transductora (NGT). En la aplicación de GIATI a diálogo los símbolos de entrada son las palabras del turno y los símbolos de salida (la lengua destino) un signo que marca el fin del segmento. El alineamiento entre la entrada y la salida se realiza alineando la última palabra de cada segmento con la marca de fin de segmento y dejando el resto de las palabras en el turno con un alineamiento vacío. Esta estrate-

[1] GIATI es el acrónimo de Grammatical Inference and Alignments for Transducer Inference

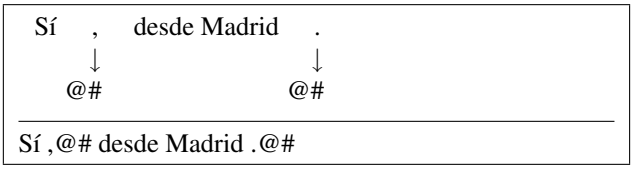

Figura 7.1: Ejemplo de re-etiquetado de un turno de la tarea. La parte de arriba muestra el alineamiento entre las palabras y los cortes del segmento. La parte inferior presenta el resultado del re-etiquetado.

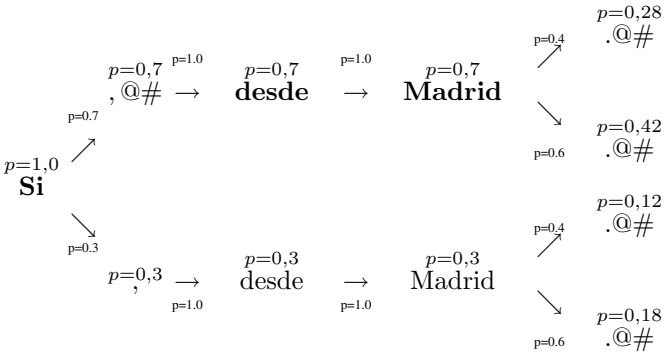

Figura 7.2: Ejemplo del árbol de búsqueda junto con la evolución de probabilidades. La rama de máxima probabilidad se muestra en negrita.

gia de alineamientos produce alineaciones sin cruces, lo que simplifica el proceso de re-etiquetado.

Para segmentar un turno concreto se emplea una implementación del algoritmo de Viterbi que utiliza el n-grama obtenido en el entrenamiento y realiza una búsqueda en árbol de la mejor anotación para la frase de entrada. En la Figura 7.2 se muestra un ejemplo de árbol de búsqueda. El nivel i en el árbol corresponde a la secuencia de las primeras i palabras del turno. Cada nodo del árbol incluye la palabra actual y la probabilidad de la secuencia. Los nodos a su vez se dividen en tantos nodos hijos como palabras puedan seguir a la palabra del nodo padre. La probabilidad de un nodo se calcula a partir de la probabilidad del padre y de la probabilidad de la nueva secuencia del n-grama, que resulta de concatenar el n-grama del padre con la palabra actual. El etiquetado final es aquel cuyo camino haya obtenido la probabilidad mayor.

7.2. Técnicas de segmentación jerárquica

La segmentación de los turnos de un nivel se puede realizar de dos maneras: de manera independiente o utilizando la información del nivel anterior. En el primer caso, en cada nivel, el lenguaje origen de los alineamientos se mantiene, es decir, las frases a segmentar son las mismas, y varía el lenguaje destino; cada nivel tiene sus propias frases con diferente segmentación. En el segundo caso, esto se modifica y se enlazan los niveles de forma que la segmentación del nivel anterior sirva para el segmentado

del siguiente. Las frases segmentadas en actos de primer nivel se utilizan como entrada para la segmentación del segundo nivel. De igual forma se repite el proceso para el tercer nivel con los resultados del segundo. En la Figura 7.3 se puede ver un esquema del proceso de etiquetado.

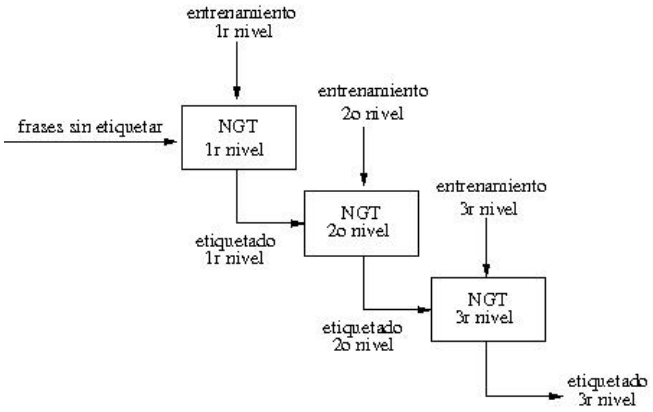

Figura 7.3: Esquema del segmentado jerárquico basado en GIATI.

7.3. Experimentos y conclusiones

Se han realizado dos baterías de experimentos. La primera comprende la segmentación y posterior etiquetado del corpus. El segundo conjunto de pruebas recoge la segmentación jerárquica y su etiquetado. Los experimentos se han realizado bajo un planteamiento de validación cruzada de 5 particiones presentado en la Sección 5.4. Se han diferenciado los resultados obtenidos con la segmentación independiente de los realizados mediante la segmentación jerárquica.

Segmentación independiente

La Tabla 7.1 muestra los resultados de segmentación utilizando GIATI con bigramas y trigramas. Se incluye segmentación para uno, dos y tres niveles. El error se mide contando las posiciones en las que el sistema establece los cortes de segmentos y comparándolas con las posiciones correctas de los cortes.

Tabla 7.1: Errores de segmentación utilizando GIATI. Se mide el error (WER/SER) de las posiciones de los cortes.

Nivel	2-gramas	3-gramas
1	1.85/2.38	1.61/2.08
2	2.75/3.53	2.77/3.53
3	4.10/5.54	3.47/4.75

Utilizando estas segmentaciones se realizó el etiquetado de segmentos con actos de diálogo mediante el modelo basado en HMMs desarrollado en la Sección 6.1, en

la Ecuación (6.4). Los resultados se muestran en la Tabla 7.2. Para la segmentación se usaron bigramas y trigramas para modelar la probabilidad de los actos de diálogo. El etiquetado se ha realizado turno a turno, por lo que el sistema no tiene información de los diálogos previos o posteriores.

Tabla 7.2: Etiquetado de segmentos obtenidos con NGT. Las columnas diferencian el modelo de lenguaje utilizado en la segmentación con NGT y el modelo de lenguaje utilizado posteriormente en el etiquetado.

	2-gramas NGT		3-gramas NGT	
DAER/TER	2-gramas	3-gramas	2-gramas	3-gramas
2 niveles	9.88/11.78	9.60/11.33	9.64/11.55	9.45/11.16
3 niveles	14.59/14.02	14.18/13.65	13.92/13.51	13.58/13.15

Segmentación jerarquizada

Los resultados de la segmentación jerarquizada se recogen en la Tabla 7.3. en este caso se ha eliminado la segmentación de primer nivel, porque es la misma que en el caso anterior.

Tabla 7.3: Errores de segmentación utilizando NGT jerarquizado.

Nivel	2-gramas	3-gramas
2 niveles	2.74/3.52	2.78/3.54
3 niveles	4.00/5.40	3.49/4.75

El etiquetado correspondiente a la segmentación jerarquizada se muestra en la Tabla 7.4.

Tabla 7.4: Etiquetado de segmentos obtenidos con NGT jerarquizado. Las columnas diferencian el modelo de lenguaje utilizado en la segmentación con NGT y el modelo de lenguaje utilizado posteriormente en el etiquetado.

	2-gramas NGT		3-gramas NGT	
DAER/TER	2-gramas	3-gramas	2-gramas	3-gramas
2 niveles	9.88/11.78	9.60/11.33	9.64/11.55	9.45/11.16
3 niveles	14.59/14.02	14.18/13.65	13.92/13.51	13.58/13.15

Conclusiones

Las diferencias entre los métodos de segmentación son inapreciables al igual que en el etiquetado derivado de ellas. La jerarquización de la segmentación con NGT no produce mejoras significativas sobre este corpus.

Capítulo 8

Conclusiones y trabajo futuro

En este capítulo se recogen las conclusiones del trabajo realizado, así como las propuestas para trabajos futuros.

8.1. Preprocesado del corpus

En el Capítulo 2 se ha presentado el corpus DIHANA. Sobre este corpus se han realizado una serie de preprocesos que facilitan la experimentación, como la categorización de las transcripciones. Además, se ha realizado la partición del corpus según los diferentes niveles de los actos de diálogo. Esta partición se ha llevado a cabo sobre las transcripciones y sobre el audio, lo que ha permitido realizar los experimentos con prosodia basados en etiquetas de primer nivel.

8.2. Influencia de la prosodia en la identificación de actos de diálogo

Los Capítulos 4 y 5 muestran dos técnicas diferentes para incorporar la prosodia en el etiquetado de actos de diálogo. En ambos casos los turnos se han procesado para que los segmentos se correspondan con un acto de diálogo de primer nivel, es decir: Pregunta, Respuesta, Afirmación, Negación, Cierre o Indefinida.

En el primer caso, se ha presentado un método mixto de etiquetado, basado en un clasificador por k-NN para la prosodia y en uno basado en HMMs discretos para la transcripción. Los resultados de ambos sistemas por separado muestran el buen funcionamiento de las transcripciones frente a un resultado muy pobre utilizando únicamente la prosodia. La combinación de ambos clasificadores no produce ninguna mejora significativa.

En el segundo caso se ha optado por trabajar únicamente con prosodia y, en vez de clasificar los segmentos con k-NN, se han utilizado modelos ocultos de Markov. Los HMMs tienen la ventaja de que pueden dar cuenta de la evolución de las características a lo largo del tiempo. El trabajo realizado muestra peores resultados de clasificación que utilizando únicamente k-NN y muy alejado de la clasificación basada en la transcripción.

A pesar de que los resultados de etiquetado basado en prosodia no muestran muy buenos comportamientos, este trabajo supone sólo una primera aproximación al pro-

blema desde esta perspectiva. Conviene seguir trabajando tanto en la extracción de características prosódicas como en métodos de clasificación. Las máquinas de soporte vectorial (SVM) [28] y las redes neuronales [5] son dos técnicas candidatas para futuras pruebas. Además, la clasificación con k-NN puede extenderse incluyendo en el vector de características la aparición o no de ciertas palabras que típicamente definen algunos actos de diálogo. Otro camino posible es reetiquetar el corpus con etiquetas más significativas de la información prosódica; por ejemplo, las preguntas se pueden diferenciar entre aquellas que muestran un comportamiento en el tono típico de una pregunta frente a las que no lo hacen.

8.3. Estimación del número de segmentos

En el Capítulo 6 se han presentado dos modelos de etiquetado de actos de diálogo basados en la transcripción. El primer modelo no incluye información sobre el número de posibles segmentos en el turno, mientras que el segundo añade una estimación de éste número. Se han presentado y comparado varias propuestas para estimar la probabilidad de que, dado un turno, éste tenga un cierto número de segmentos.

Los resultados demuestran que conocer el número de segmentos en un turno mejora el etiquetado de manera significativa. También se puede ver cómo las diferentes estimaciones de la probabilidad del número de segmentos producen resultados similares y no parece existir una relación directa entre lo bueno que es un estimador y una disminución del error de etiquetado.

Como trabajo futuro se plantea la posibilidad de exportar este modelo a otros corpus como SwitchBoard, que no es un corpus dirigido como el empleado. También se deben probar nuevos estimadores de probabilidad que aproximen el error de etiquetado al error obtenido cuando se conoce el número correcto de segmentos.

8.4. NGT jerarquizado

Esta técnica importada de la traducción da buenos resultados, como se ha visto en el Capítulo 7. De hecho los errores de etiquetado con la segmentación estimada son mejores que los obtenidos en el Capítulo 6. Sin embargo, la mejora propuesta, la jerarquización del etiquetado, no aporta mejoras significativas.

Como trabajo futuro queda la posibilidad de estimar con NGT, no la segmentación, sino el número de segmentos del turno para recurrir después al etiquetado del Capítulo 6. Además, NGT puede utilizarse para estimar etiquetados, no sólo segmentaciones como se ha visto en este trabajo. A partir de ese etiquetado se puede obtener también una estimación de la segmentación y del número de segmentos.

Bibliografía

[1] A. Abella, M.K. Brown, and B. Buntschuh. Development principles for dialog-based interfaces. In *Proceedings of the Workshop on Dialog Processing in Spoken Language Systems ECAI-96*, 1996.

[2] N. Alcácer, J. Benedí, F. Blat, R. Granell, C. D. Martínez, and F. Torres. Acquisition and labelling of a spontaneous speech dialogue corpus. In *Proceeding of 10th International Conference on Speech and Computer (SPECOM).*, pages 583–586, 2005.

[3] J.L. Austin. *How to Do Things with Words?* Oxford University Press, 1962.

[4] J.-M. Benedí, E. Lleida, A. Varona, M.-J. Castro, I. Galiano, R. Justo, I. López de Letona, and A. Miguel. Design and acquisition of a telephone spontaneous speech dialogue corpus in spanish: Dihana. *Fifth International Conference on Language Resources and Evaluation (LREC)*, pages 1636–1639, May 2006.

[5] Chris Bishop and Geoffrey Hinton. *Neural Networks for Pattern Recognition.* Clarendon Press. Oxford., 1995.

[6] Johan Bos, Ewan Klein, Oliver Lemon, and Tetsushi Oka. Dipper: Description and formalisation of an information-state update dialogue system architecture. In *4th SIGdial Workshop on Discourse and Dialogue*, pages 115–124, 2003.

[7] Mark G. Core and James F. Allen. Coding dialogues with the DAMSL annotation scheme. In David Traum, editor, *Working Notes: AAAI Fall Symposium on Communicative Action in Humans and Machines*, pages 28–35, Menlo Park, California, 1997. American Association for Artificial Intelligence.

[8] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience Publication, 2000.

[9] J. Flanagan. *Speech Analysis, Synthesis, and Perception*. Springer-Verlag, Berlin-Heidelberg-New York, 1972.

[10] M. Fraser and G. Gilbert. Simulating speech systems. *Computer Speech and Language*, 5(1):81–89, 1991.

[11] T. Fukada, D. Koll, A. Waibel, and K. Tanigaki. Probabilistic dialogue act extraction for concept based multilingual translation systems. *ICSLP 98*, pages 2771–2774, 1998.

[12] D. Gerhard. Pitch extraction and fundamental frequency: History and current techniques. Technical report 2003-06, Department of Computer Science, University of Regina, November 2003. ISBN 0-7731-0455-0.

[13] J. Godfrey, E. Holliman, and J. McDaniel. Switchboard: Telephone speech corpus for research and development. In *Proc. ICASSP-92*, pages 517–520, 1992.

[14] A. L. Gorin, B. A. Parker, R. M. Sachs, and J. G. Wilpon. How may i help you. *Speech Communication*, 23:113–127, 1997.

[15] D. Jurafsky, E. Shriberg, and D. Biasca. Switchboard swbd-damsl shallow-discourse-function annotation coders manual - draft 13. Technical Report 97-01, University of Colorado Institute of Cognitive Science, 1997.

[16] A. Lavie, L. Levin, P. Zhan, M. Taboada, D. Gates, M. Lapata, C. Clark, M. Broadhead, and A. Waibel. Expanding the domain of a multi-lingual speech-to-speech translation system. In *Proceedings of the Workshop on Spoken Language Translation, ACL/EACL-97*, 1997.

[17] D. Llorens, F. Casacuberta, E. Segarra, J. Sánchez, P. Aibar, and M. Castro. Acoustical and syntactical modeling in the atros system. In *International Conference on Acoustic, Speech and Signal Processing*, volume 2, pages 641–644. IEEE press, March 1999.

[18] P.C. Mahalanobis. On the generalised distance in statistics. In *Proceedings of the National Institute of Science of India*, pages 49–55, 1936.

[19] Carlos-D. Martinez-Hinarejos, Jose-Miguel Benedi, and Ramon Granell. Statistical framework for a spanish spoken dialogue corpus. *Speech Communication*, 50:992–1008, 2008.

[20] Michael F. Mctear, Susan Allen, Laura Clatworthy, Noelle Ellison, Colin Lavelle, and Helen Mccaffery. Integrating flexibility into a structured dialogue model: Some design considerations. In *Proceedings of International Conference on Speech and Language Processing*, pages 943–946, 2000.

[21] Helen M. Meng, Carmen Wai, and Roberto Pieraccini. The use of belief networks for mixed-initiative dialog modeling. In *Int Conference on Spoken Language Processing*, pages 757–773, 2000.

[22] R. Paredes and E. Vidal. Learning weighted metrics to minimize nearest-neighbor classification error. *IEEE Transaction on Pattern Analisys and Machine Intelligence*, 28(7), 2006.

[23] R. Pieraccini and J. M. Huerta. *Recent Trends inDiscourse and Dialogue*, volume 39 of *Text, Speech and Language Technology*, chapter Where do we go from here?, pages 1–24. Springer, 2008. ISBN 978-1-4020-6820-1.

[24] Antonio Quilis and Joseph A. Fernández. *Curso de fonética y fonología española*. CSIC, 1993. ISBN 84-00-07088-5.

[25] A. Sanchis, A. Juan, and E. Vidal. New features based on multiple word graphs for utterance verification. In *8th International Conference on Spoken Language Processing*, pages 2545–2548, October 2004.

[26] D. Schiffrin. *Approaches to Discourse*. Blackwell textbooks in linguistics, 1994. ISBN 0-631-16622.

[27] J.R. Searle. *Speech Acts*. Cambridge University Press, 1969.

[28] John Shawe-Taylor and Nello Cristianini. *Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, 2000.

[29] K. Sjölander and J. Beskow. Wavesurfer - an open source speech tool. *Proc of ICSLP*, pages 464–467, October 2000.

[30] A. Stolcke, N. Coccaro, R. Bates., P. Taylor, C. Van Ess-Dykema, K. Ries, E. Shriberg, D. Jurafsky, R. Martin, and M. Meteer. Dialogue act modelling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):1–34, 2000.

[31] J. Williams and S. Young. Partially observable markov decision processes for spoken dialog systems. *Computer Speech and Language*, 21(2):393–422, 2007.

[32] S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland. *The HTK Book*. CUED, UK, v3.2 edition, July, 2004.

[33] Steve Young. Probabilistic methods in spoken dialogue systems. *Philosophical Transactions of the Royal Society (Series A)*, 358:1389–1402, 2000.