# Uso de Información Contextual en la Interfaz de Entrada de Un Sistema de Diálogo

**R. López-Cózar**

Dpto. Lenguajes y Sistemas Informáticos, E.T.S. Ingeniería Informática
18071 Universidad de Granada, Tel.: +34 958 240579, FAX: +34 958 243179
E-mail: rlopezc@ugr.es

**Resumen:** Este artículo propone una nueva técnica cuya finalidad es mejorar el funcionamiento de la interfaz de entrada de los sistemas de diálogo combinando las ventajas que presentan las gramáticas dependientes e independientes de los *prompts* generados por dichos sistemas. La técnica propone la creación de un reconocedor del habla denominado *a dos niveles* ya que consta de dos módulos. El primer módulo es un reconocedor estándar de habla continua que recibe la voz del usuario y produce como salida un grafo de palabras. El segundo módulo es un analizador de grafos cuya función es procesar la salida del primer módulo teniendo en cuenta el prompt generado por el sistema de diálogo. Aplicando la técnica se ha evaluado el funcionamiento de la interfaz de entrada de un sistema de diálogo. Los resultados experimentales muestran que cuando se procesan frases *en-contexto*, las tasas de exactitud de palabras y comprensión de frases se incrementan en un 4,09% y 4,19%, respectivamente, mientras que cuando se procesan frases *fuera-contexto*, las tasas de exactitud y comprensión se incrementan en un 87,74% y 82,04%, respectivamente.
**Palabras clave:** Sistema de diálogo, reconocimiento del habla, procesamiento de lenguaje natural, comprensión del habla, gestión del diálogo, generación de lenguaje natural, síntesis de voz.

**Abstract**: This paper proposes a new technique that aims to enhance the performance of the input interface of spoken dialogue systems by means of combining the advantages of the prompt-dependent and prompt-independent grammars used by these systems. The technique proposes creating a *two-level speech recogniser* since it contains two modules. The first one is a continuous-speech standard recogniser that receives the user voice and produces as output a graph of words. The second one is a word-graph analyser that processes the output of the first module considering the prompt generated by the dialogue system. The input interface of a spoken dialogue system has been evaluated using the technique. The experimental results show that when the *in-context* utterances are analysed, the speech recognition and understanding rates increase in 4.09% and 4.19%, respectively. They also show that when the *out-of-context* utterances are analysed, the speech recognition and understanding rates increase in 87.74% and 82.04%, respectively.
**Keywords**: Spoken dialogue systems, speech recognition, natural language processing, speech understanding, dialogue management, natural language generation, speech synthesis.

## *1 Introducción*

A fin de optimizar el proceso de reconocimiento del habla, un gran número de sistemas de diálogo procesan cada frase del usuario mediante una gramática asociada a cada tipo de prompt del sistema. Si bien esta técnica es útil en determinadas ocasiones, no es adecuada si los usuarios pronuncian frases no previstas por la gramática activa en cada momento (la asociada al prompt del sistema). Por ejemplo, si el sistema genera el prompt *"Por favor, diga su número de teléfono"* y el usuario dice un número de teléfono, la frase del usuario puede ser correctamente reconocida; pero si el usuario pronuncia otro tipo de frase (una dirección, p.e.), la salida del reconocedor será cualquiera de los números de teléfono posibles, y la

dirección nunca podrá ser reconocida. Como consecuencia, el usuario se sentirá incómodo usando el sistema pues notará que cualquier desviación respecto de las indicaciones del sistema provoca el funcionamiento incorrecto del mismo.

A fin de intentar solventar este problema, existen sistemas de diálogo que usan una gramática general durante todo el diálogo, que llamaremos *G-gramática* en este artículo. Esta gramática está activa para reconocer cualquier frase del usuario, independientemente del prompt generado por el sistema en cada momento. Sin embargo, la G-gramática tiende a ser bastante compleja y el vocabulario permitido en cada momento suele ser notablemente mayor, lo que tiende a incrementar el número de errores de reconocimiento y, como consecuencia, provocar el funcionamiento erróneo del sistema.

La finalidad de la técnica propuesta en este artículo es combinar las ventajas de ambas aproximaciones. Así, para permitir la libertad de interacción del usuario se usa una G-gramática, y para mejorar el proceso de reconocimiento del habla se usa un módulo adicional que denominamos *G-analizador* (analizador de grafos). Este analizador usa tuplas de clases de palabras que proporcionan información relacionada con el contexto actual del diálogo durante el reconocimiento de cada frase.

## *2 Reconocimiento a dos niveles*

La técnica propuesta en este artículo se basa en el uso de un reconocedor del habla que denominamos *a dos niveles* por estar formado por dos módulos (Figura 1).

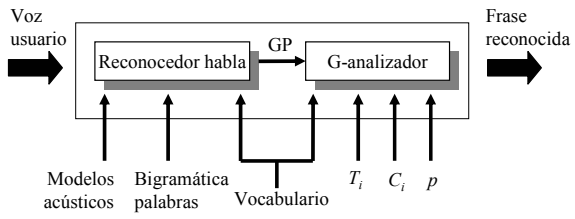


Figura 1. Reconocedor a dos niveles

El primer módulo es un reconocedor de habla continua estándar que recibe la voz del usuario y produce como salida un grafo de palabras (GP), usando modelos acústicos previamente entrenados y una bigramática de palabras, compilada a partir de un corpus de frases de entrenamiento pertenecientes al dominio de aplicación del sistema. Un GP es una red formada por un conjunto de nodos y un conjunto de arcos. Los nodos representan palabras y los arcos transiciones entre las palabras. La Figura 2 muestra un GP simple que representa una gramática que permite reconocer frases de la forma *"sí pero sí pero ..."*. Este GP contiene un nodo inicial, un nodo final, las palabras *"pero"* y *"sí"* y dos nodos nulos. Cada arco tiene asociada una determinada probabilidad de transición $p_i$ que el reconocedor del habla estima usando medidas acústicas y modelos de lenguaje.

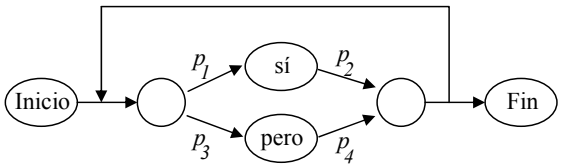


Figura 2. Ejemplo de grafo de palabras

El segundo componente del reconocedor a dos niveles es un módulo que denominamos G-analizador, cuya función es recibir el GP correspondiente a cada frase del usuario y producir como frase reconocida aquélla que se corresponda con el camino de mayor probabilidad en el GP. El análisis del GP se realiza usando tres parámetros: el prompt actual ($T_i$) del sistema de diálogo, un conjunto ($C_i$) de tuplas de clases de palabras creado previamente a partir de frases de usuarios destinadas a responder dicho prompt, y un parámetro denominado *incremento de probabilidad* ($\Delta_p$) cuya finalidad es incrementar la probabilidad de determinadas transiciones en el GP.

Ambos módulos del reconocedor a dos niveles reciben como entrada el conjunto de palabras que constituye el vocabulario del sistema de diálogo, es decir, el conjunto de palabras que pueden ser reconocidas.

### 2.1 Creación del asociaciones prompts-conjuntos de tuplas

La técnica presentada propone crear un conjunto que denominamos $\Omega$, constituido por asociaciones entre prompts generados por el sistema de diálogo y conjuntos formados por tuplas de clases de palabras:

$$\Omega = \{ T_i , C_i \}, \quad i = 1 \ldots n$$

donde $T_i$ representa un tipo de prompt generado por el sistema de diálogo y $C_i$ representa un conjunto formado por tuplas de clases de palabras ($W_m$,$W_n$) obtenido mediante el análisis de las frases que los usuarios usan para responder a dicho prompt del sistema. Por ejemplo, si $T_1$ fuera el prompt: *"Por favor, diga su número de teléfono"*, entonces $C_1$ sería un conjunto formado por tuplas de clases de palabras creado mediante el análisis de frases pronunciadas por los usuarios para proporcionar números de teléfono, p.e. *"9 1 1 2 3 4 5 6 7"*.

La creación de los conjuntos $C_i$ se puede realizar en tres etapas, las dos primeras de forma manual y la tercera de forma automática. La primera etapa consiste en analizar un corpus de frases pertenecientes al dominio de aplicación del sistema para averiguar qué frases pueden ser pronunciadas por los usuarios para responder a los prompts $T_i$ generados por el sistema de diálogo. Dichas frases son clasificadas formando grupos $F_1$ , $F_2$ , $F_3$ , ... (p. e., $F_1$ = pedidos de comida, $F_2$ = números de teléfono, $F_3$ = direcciones, $F_4$ = confirmaciones, etc.). Cada $F_i$ es el grupo de frases asociado a $T_i$.

La segunda etapa consiste en estudiar las frases para averiguar qué palabras son necesarias para obtener el contenido semántico de las mismas (llamadas usualmente *palabras-clave*) y agruparlas en clases de palabras $W_1$ , $W_2$, $W_3$ ... $W_n$.

La tercera etapa se puede dividir en dos fases. En la primera, cada frase *f* del usuario se analiza y se transforma en una frase *f'* en la que cada *palabra-clave* se sustituye por el nombre de la clase de palabras $W_k$ a la que pertenece. En la segunda fase, se analiza cada frase *f'* de cada uno de los tipos de frases $F_i$, y se crea una tupla de clases de palabras ($W_m$, $W_n$) por cada dos clases de palabras que aparecen de forma consecutiva en la frase *f'*. Cada tupla encontrada se añade a un conjunto $C_i$ que se asocia al prompt $T_i$. El objetivo de crear las tuplas de clases de palabras es explotar la información gramatical correspondiente a los patrones sintácticos de las frases (Hidaki y Kobayashi, 1999).

Denominamos *expansión* de un conjunto de tuplas al conjunto de todos los posibles pares de palabras que se pueden obtener sustituyendo los nombres de las clases de palabras por las palabras contenidas en las clases.

## 2.2 Procedimiento para analizar grafos de palabras

El procedimiento para analizar GPs usa una G-gramática para procesar las frases de los usuarios, tanto las que se producen en *en-contexto* (p. e., el sistema solicita un número de teléfono y el usuario proporciona un número de teléfono) como las que se producen *fuera-contexto* (p. e., el sistema solicita un número de teléfono y el usuario proporciona una dirección). La finalidad de este procedimiento es incrementar las probabilidades de transición de las frases en-contexto, ya que son éstas las que por lo general pronunciará el usuario en un momento dado del diálogo. El procedimiento se puede expresar algorítmicamente de la siguiente forma:

**analizaGP** ( **Input**: *GP*, $T_i$, $\Delta_p$ ; **Output**: *f* )

// *GP* = grafo de palabras
// $T_i$ = prompt actual del sistema de diálogo
// $\Delta_p$ = incremento de probabilidad
// *f* = frase reconocida (camino de mayor probabilidad)
{
/* La siguiente función procesa el GP y crea un conjunto con todas las transiciones que contiene. Cada transición $t_i$ tiene dos probabilidades: una acústica y otra del lenguaje */

transiciones = procesa(GP)

/* Para poder crear el mejor camino en el GP, la siguiente función crea en cada nodo del grafo dos campos adicionales (*prob* y *palabra_anterior*), con los valores iniciales indicados a continuación: prob = MIN_VALOR y palabra_anterior = NULL */

iniciarEspacioBusqueda ()

/* El siguiente bucle *for all* crea el mejor camino en el GP, recorriendo todas las transiciones entre las palabras */

**For all** $t_i \in$ transiciones **in** GP **do**

$w_S$ = nodoInicioTransicion($t_i$) . palabra
$w_E$ = nodoFinTransicion($t_i$) . palabra
$p_A$ = probAcustica($t_i$)
$p_L$ = probLenguaje($t_i$)
$p_S$ = nodoIncioTransicion($t_i$) . prob
incremento = 0.0

$C_i$ = Ω( $T_i$ ) // obtiene $C_i$ asociado a $T_i$

∏ = clasesPalabras( $w_S$ )
∑ = clasesPalabras( $w_E$ )

/* El siguiente bucle *for each* comprueba todos los posibles pares de clases de palabras a los que pertenecen $w_S$ y $w_E$ . Si se encuentra un par en $C_i$, se

```
  usa Δp para incrementar la probabilidad de la transición */

  for each Wm ∈∏
    for each Wn ∈ ∑
        U = (Wm,Wn)
        if (U ∈ Ci) then incremento = Δp
                         break
        endif
  endfor

  if (nodoFinTransicion(ti) . prob == MIN_VALOR )
  then  nueva_prob = pA + pL + incremento
        nodoFinTransicion(ti) . prob = nueva_prob
        nodoFinTransicion(ti) . palabra_anterior = wS

  else  nueva_prob = pA + pL + pS + incremento
        if (nueva_prob > nodoFinTransicion(ti) . prob)
        then nodoFinTransicion(ti) . prob = nueva_prob
             nodoFinTransicion(ti).palabra_anterior = wS
        endif
  endif
 enddo

  /* A continuación se recorre el camino de mayor
  probabilidad hacia atrás introduciendo las palabras de los
  nodos en una pila, comenzando por la palabra de mayor
  probabilidad acumulada */

 w = palabraDeMayorProbabilidadAcumulada()

 while (w <> NULL) do
   meteEnPila(w)
   w = palabraAnteriorEnCamino(w)
 enddo

  /* Finalmente se imprimen las palabras de la pila, que
  constituyen la frase reconocida */

 while ( not pilaVacia() ) do
   sacaDePila(w)
   write(w, " ")
 enddo
}
```

Figura 3. Algoritmo para analizar GPs

Al analizar un GP, el G-analizador usa el parámetro $\Delta_p$ para incrementar la probabilidad de la transición $w_S \rightarrow w_E$ si el par de palabras ($w_S$ , $w_E$) aparece en la expansión del conjunto de tuplas de clases de palabras $C_i$ asociado al prompt $T_i$.

## *3 Experimentos*

La finalidad de los experimentos es evaluar el funcionamiento de una nueva versión de la interfaz de entrada del sistema de diálogo SAPLEN (López-Cózar et al. 2000), en la que el reconocedor[1] de habla continua basado en HTK (Hain et al., 1999) usado por el sistema constituye ahora el primer módulo del reconocedor del habla a dos niveles (ver Figura 1). El segundo módulo de la nueva interfaz, es decir el analizador semántico, sigue siendo el mismo que se utiliza en la interfaz original.

En la evaluación se han usado las dos medidas siguientes: exactitud de palabras (WA, *Word Accuracy*) y comprensión de frases (SU, *Sentence Understanding*) (Huang et al. 2001). WA se calcula de la siguiente forma: $WA = ( w_t - w_i - w_s - w_d ) / w_t$, donde $w_t$ es el número total de palabras en las frases, y $w_i$, $w_s$ y $w_d$ es el número de palabras insertadas, sustituidas y borradas erróneamente durante el reconocimiento. SU se calcula como sigue: $SU = S_u / S_t$, donde $S_u$ es el número de frases para las cuales el analizador semántico obtiene una representación semántica correcta, y $S_t$ es el número total de frases analizadas.

Para realizar el reconocimiento del habla, la interfaz de entrada original del sistema SAPLEN puede utilizar un total de 17 bigramáticas de palabras, $BG_i$, compiladas a partir de corpora de frases de los tipos $F_i$ mostrados en la Tabla 1. La selección de la gramática que se usa en cada momento se realiza en función del parámetro *prompt_sistema*, que informa acerca del prompt actual generado por el sistema. En cambio, la nueva interfaz usa una G-gramática y emplea tres parámetros para reconocer cada frase: *tipo_frase*[2] (indica el tipo de frase que se va a procesar, ver Tabla 1), *prompt_sistema* (ahora indica el tipo de prompt supuestamente generado por el sistema de diálogo cuando la interfaz de entrada recibe la frase de tipo *tipo_frase*) e *incremento_probabilidad* ($\Delta_p$, valor que indica cuánto se han de incrementar las transiciones entre palabras durante el análisis de los GPs).

El incremento de probabilidad ($\Delta_p$) permite evaluar el efecto de la técnica propuesta en el análisis de las frases. Así, si $\Delta_p$=0 la información relacionada con las tuplas de clases de palabras no se usa durante el análisis de los GPs correspondientes a las frases, en cuyo caso, el G-analizador se comporta igual que un reconocedor de Viterbi (Rabiner y Juang, 1993). En cambio, cuando $\Delta_p$>0 las transiciones de los GPs se incrementan de acuerdo con el

[1] Dicho reconocedor se configura ahora para que genere un GP a partir de la frase de entrada (voz del usuario), en lugar de una frase reconocida.

[2] Este parámetro sólo se usa a efectos de evaluación, ya que en un diálogo real no se puede saber a priori qué tipo de frase pronuncia el usuario.

procedimiento descrito en la Figura 3. En los experimentos realizados se han usado 19 valores distintos del parámetro $\Delta_p$: 0, 1, 2, 3, 4, ..., 18, a fin de obtener diferencias apreciables en los valores de las medidas de evaluación que permitan extraer conclusiones.

| Tipo | Frase |
|---|---|
| $F_1$ | Pedido producto |
| $F_2$ | Nº teléfono |
| $F_3$ | Código postal |
| $F_4$ | Dirección |
| $F_5$ | Consulta |
| $F_6$ | Confirmación |
| $F_7$ | Cantidad pedido |
| $F_8$ | Nombre comida |
| $F_9$ | Contenido comida |

| Tipo | Frase |
|---|---|
| $F_{10}$ | Nombre bebida |
| $F_{11}$ | Tamaño |
| $F_{12}$ | Sabor |
| $F_{13}$ | Temperatura |
| $F_{14}$ | Nombre direcc. |
| $F_{15}$ | Número edificio |
| $F_{16}$ | Planta edificio |
| $F_{17}$ | Letra edificio |

Tabla 1. Tipos de frases usadas en los experimentos

## 3.1 Descripción de los corpora usados en los experimentos

Siguiendo el procedimiento descrito en la Sección 2.1, se ha creado un corpus de frases y un conjunto Ω de asociaciones prompts-conjuntos de tuplas. El corpus de frases ha sido creado a partir de un corpus de diálogos previamente grabado en un restaurante de comida rápida, que contiene unos 500 diálogos entre clientes y empleados. Dichos diálogos fueron transcritos, analizados semánticamente y etiquetados mediante trabajos previos, incluyendo etiquetas referentes al locutor y función pragmática de las frases.

Para crear el corpus de frases se han seleccionado aleatoriamente 1.700 grabaciones de frases de clientes, repartidas entre los 17 tipos de frases mostrados en la Tabla 1. Además, se han añadido al corpus las 1.700 transcripciones fonéticas (frases en formato de texto) y las 1.700 representaciones semánticas de las frases. A partir de las 1.700 transcripciones fonéticas se ha compilado una G-gramática (bigramática de palabras, concretamente) que es usada por el analizador a dos niveles para producir la frase reconocida.

Las clases de palabras han sido obtenidas analizando manualmente las 1.700 transcripciones fonéticas. La Tabla 2 muestra algunas de las clases de palabras obtenidas a partir de este análisis.

| Clase palabras | Palabras distintas en clase palabras | Palabras de ejemplo |
|---|---|---|
| NÚMERO | 103 | *uno, dos, tres, …* |
| COMIDA | 6 | *bocadillo, tarta, …* |
| INGREDIENTE | 28 | *jamón, queso, …* |
| BEBIDA | 16 | *cerveza, vino, …* |
| TAMAÑO | 4 | *pequeño, grande, …* |
| SABOR | 6 | *naranja, limón* |
| TEMPERATURA | 6 | *frío, caliente, ...* |
| TIPO_DIR | 5 | *calle, plaza, ...* |
| NOMBRE_DIR | 162 | *zacatín, alhóndiga, ...* |
| PLANTA | 20 | *primero, segundo, ...* |
| LETRA | 28 | *a, b, c, d, e, ...* |

Tabla 2. Ejemplos de clases de palabras

Para construir el conjunto Ω siguiendo el procedimiento descrito en la Sección 2.1, se ha analizado automáticamente cada uno de los 17 tipos de frases mostrados en la Tabla 1, utilizando las clases de palabras obtenidas del análisis del corpus de frases. Como resultado de este análisis, para cada conjunto $F_i$ se ha obtenido un conjunto $C_i$, que ha sido asociado a uno de los 17 tipos de prompts $T_i$ que el sistema SAPLEN puede generar.

Las Tablas 3 y 4 muestran, respectivamente, algunos ejemplos de conjuntos de tuplas y asociaciones prompt-conjunto de tuplas.

| Conjuntos de tuplas | Descripción de cada conjunto |
|---|---|
| $C_1$ | *(NÚMERO, COMIDA)*<br>*(COMIDA, INGREDIENTE)*<br>*(NÚMERO, INGREDIENTE)*<br>*(NÚMERO, BEBIDA)*<br>*(NÚMERO, TAMAÑO)*<br>*(NÚMERO, SABOR)*<br>*(SABOR, TAMAÑO)*<br>*(TAMAÑO, SABOR)*<br>*(SABOR, TEMPERATURA)*<br>*(TAMAÑO, TEMPERATURA)* |
| $C_2$ | *(NÚMERO, NÚMERO)* |
| $C_3$ | *(NÚMERO, NÚMERO)* |
| $C_4$ | *(TIPO_DIR, NOMBRE_DIR)*<br>*(NOMBRE_DIR, NUMERO)*<br>*(NUMERO, PLANTA)*<br>*(PLANTA, LETRA)* |

Tabla 3. Ejemplos de tuplas de clases de palabras

| $T_i$ | $C_i$ | Frase de ejemplo |
|---|---|---|
| Pedido producto | $C_1$ | *Un bocadillo de jamón* |
| Nº teléfono | $C_2$ | *9 5 8 1 3 2 4 1 5* |
| Código postal | $C_3$ | *1 8 0 1 4* |
| Dirección | $C_4$ | *Calle Alhóndiga 23 primero A* |

Tabla 4. Ejemplos de asociaciones prompt-conjunto de tuplas de clases de palabras

## 3.2 Resultados experimentales

*3.2.1 Gramáticas dependientes de los prompts*

En primer lugar se han realizado experimentos usando la interfaz de entrada original del sistema SAPLEN. Como se puede observar en la Tabla 5, el funcionamiento de esta interfaz es aceptable cuando se analizan frases en-contexto (es decir, cuando el tipo de frase se corresponde con el prompt). Sin embargo, su funcionamiento es totalmente inaceptable cuando se analizan frases fuera-contexto (es decir, cuando el tipo de frase no se corresponde con el prompt), dado que las frases no se reconocen correctamente, y como consecuencia, no se comprenden. Las salidas del reconocedor basado en HTK son frases permitidas por la bigramática activa en cada momento; por consiguiente, si la bigramática solo permite reconocer números de teléfono, por ejemplo, sólo se reconocen números de teléfono, independientemente del tipo de frase que realmente pronuncie el usuario.

| Tipo frase | Prompt | WA | SU |
|---|---|---|---|
| Pedido producto | Pedido producto | 93,39 | 94,36 |
| | Nº teléfono | 0.1 | 0 |
| Nº teléfono | Nº teléfono | 94,36 | 92,61 |
| | Confirmación | -0,33 | 0 |
| Código postal | Código postal | 94,82 | 91,49 |
| | Dirección | -0,5 | 0 |
| Dirección | Dirección | 96,3 | 75,66 |
| | Código postal | -0,03 | 0 |

Tabla 5. Análisis de frases usando gramáticas dependientes de los prompts del sistema

*3.2.2 G-analizador con frases en-contexto*

La Tabla 6 muestra los resultados medios obtenidos al analizar las 1.700 frases del corpus mediante la técnica propuesta, usando cada uno de los 19 valores del parámetro $\Delta_p$.

| $\Delta_P$ | WA | SU |
|---|---|---|
| 0 | 88,57 | 83,43 |
| 1 | 88,97 | 83,75 |
| 2 | 89,45 | 84,53 |
| 3 | 89,82 | 85,09 |
| 4 | 90,22 | 85,60 |
| 5 | 99,54 | 85,95 |
| 6 | 90,87 | 86,34 |
| 7 | 91,16 | 86,60 |
| 8 | 91,39 | 86,83 |
| 9 | 91,61 | 86,85 |

| $\Delta_P$ | WA | SU |
|---|---|---|
| 10 | 91,89 | 87,17 |
| 11 | 92,19 | 87,41 |
| 12 | 92,42 | 87,50 |
| 13 | 92,66 | 87,62 |
| 14 | 92,37 | 87,19 |
| 15 | 92,03 | 86,74 |
| 16 | 91,75 | 86,38 |
| 17 | 91,35 | 85,59 |
| 18 | 91,05 | 85,24 |

Tabla 6. Análisis de frases en-contexto

Se puede observar que los resultados más bajos se obtienen cuando $\Delta_p$=0, dado que en este caso no se incrementan las probabilidades de transición en los GPs, lo cual equivale a no usar la información proporcionada por las tuplas de clases de palabras. Los resultados se incrementan conforme aumenta el valor de $\Delta_p$ hasta llegar al valor $\Delta_p$=13, a partir del cual disminuyen. Por consiguiente, este es el valor óptimo del parámetro $\Delta_p$ para el corpus de frases usado. Cuando $\Delta_p$<13, el G-analizador no obtiene suficiente beneficio de la información proporcionada por las tuplas de clases de palabras. Este hecho se deduce claramente de los ficheros de traza creados durante el análisis de las frases, en los que se observa que se produce un gran número de sustituciones de palabras (p. e., *"sí"* se sustituye por *"seis"* o *"sin"*; *"veintitrés"* se sustituye por *"verde tres"*; *"cero"* se sustituye por *"pero"* o *"acera"*; etc.). En cambio, cuando $\Delta_p$>13, el G-analizador incrementa excesivamente las probabilidades de transición en los GPs. Como consecuencia, aumenta el número de inserciones de palabras siguiendo la estructura sintáctica de las tuplas de clases de palabras. Asimismo, aumenta el número de sustituciones de palabras no significativas (no incluidas en las clases de palabras) por palabras incluidas en dichas clases (p. e., en los ficheros de traza se observa que a menudo *"pero"* se sustituye por *"queso"*, etc.). En la situación óptima ($\Delta_p$=13), la técnica permite incrementar WA en 4,09% (desde 88,57% para $\Delta_p$=0 hasta 92.66% para $\Delta_p$=13) y SU en 4,19% (desde 83,43% para $\Delta_p$=0 hasta 87.62% para $\Delta_p$=13).

*3.2.3 G-analizador con frases fuera-contexto*

Para concluir los experimentos, se han estudiado las seis combinaciones $T_i$,$F_i$ siguientes, representativas de casos en los que los usuarios pueden pronunciar frases fuera-contexto:

*i)* $T_i$ = solicitar número de teléfono, $F_i$ = realizar pedido
*ii)* $T_i$ = solicitar código postal, $F_i$ = indicar número de teléfono
*iii)* $T_i$ = solicitar confirmación explícita del número de teléfono, $F_i$ = indicar número de teléfono
*iv)* $T_i$ = solicitar dirección, $F_i$ = indicar código postal
*v)* $T_i$ = solicitar confirmación explícita del código postal, $F_i$ = indicar código postal
*vi)* $T_i$ = solicitar confirmación explícita de la dirección, $F_i$ = indicar dirección

En estos seis casos, si se usaran gramáticas dependientes de los prompts, el reconocedor original del sistema proporcionaría frases completamente erróneas; sin embargo, usando el G-analizador las frases fuera-contexto también pueden ser correctamente reconocidas. La Tabla 7 muestra los resultados medios obtenidos al analizar los seis tipos de frases fuera-contexto indicados anteriormente. Se puede observar que la tendencia es distinta a la expuesta anteriormente; los mejores resultados se obtienen cuando $\Delta_p$=0 y disminuyen ligeramente conforme aumenta el valor de $\Delta_p$. Este efecto se produce porque el porcentaje de probabilidades que el G-analizador incrementa es muy reducido, pues en todas las situaciones fuera-contexto, excepto en la ii), las secuencias de palabras en la expansión del $C_i$ usado no suelen aparecer en los GPs correspondientes a las frases analizadas. En el caso ii) en cambio, el G-analizador usa el conjunto de tuplas $C_3$ mostrado en la Tabla 3, y por consiguiente, incrementa las probabilidades de las transiciones entre números, que sí existen en los GPs correspondientes a números de teléfonos.

| *p* | WA | SU |
|---|---|---|
| 0 | 89,21 | 78,27 |
| 1 | 89,19 | 78,27 |
| 2 | 89,18 | 78,27 |
| 3 | 89,17 | 78,22 |
| 4 | 89,17 | 78,17 |
| 5 | 89,16 | 78,13 |
| 6 | 89,13 | 78,12 |
| 7 | 89,08 | 78,09 |
| 8 | 89,07 | 78,08 |
| 9 | 89,06 | 78,08 |

| *P* | WA | SU |
|---|---|---|
| 10 | 89,03 | 77,95 |
| 11 | 88,97 | 77,95 |
| 12 | 88,91 | 77,70 |
| 13 | 88,90 | 77,42 |
| 14 | 88,89 | 77,32 |
| 15 | 88,80 | 77,13 |
| 16 | 88,72 | 77,09 |
| 17 | 88,64 | 76,85 |
| 18 | 88,55 | 76,53 |

Tabla 7. Análisis de frases fuera-contexto

En una situación real no se puede saber a priori si una frase dada del usuario estará en-contexto o fuera-contexto; no obstante, es lógico suponer que los usuarios pronunciarán frases en-contexto usualmente, y frases fuera-contexto en contadas ocasiones. De esta premisa, y de los resultados correspondientes al análisis de frases en-contexto mostrados en la Tabla 6, se deduce que se debe usar la técnica propuesta con el valor $\Delta_p$=13, ya que es el que permite obtener mejores resultados.

Cabe preguntarse pues, qué resultados se obtendrían si se usara siempre dicho valor, incluso para analizar frases fuera-contexto. Para responder a esta pregunta, la Tabla 8 muestra los resultados medios obtenidos cuando $\Delta_p$=13 para las seis situaciones fuera-contexto indicadas anteriormente (Tipo frase ≠ Prompt), así como los resultados medios obtenidos cuando los cuatro tipos de frases (pedidos de productos, números de teléfono, códigos postales y direcciones) se analizan en-contexto (Tipo frase = Prompt).

Como se observa en dicha Tabla, se obtienen mejores resultados cuando las frases se analizan en-contexto, pues cuando el análisis se realiza fuera-contexto el G-analizador incrementa indebidamente algunas transiciones de los GPs. A pesar de que los resultados obtenidos cuando el análisis se realiza fuera-contexto no son muy altos, son mucho mejores que los obtenidos cuando se usan las gramáticas dependientes de los prompts (comparar con resultados fuera-contexto mostrados en Tabla 5).

| Tipo frase | Prompt | WA | SU |
|---|---|---|---|
| Pedido producto | Pedido producto | 89,94 | 88,88 |
| | Nº teléfono | 86,44 | 82,64 |
| Nº teléfono | Nº teléfono | 94,11 | 89,92 |
| | Código postal | 94,11 | 89,92 |
| | Confirmación | 88,4 | 82,39 |
| Código postal | Código postal | 94,12 | 89,32 |
| | Dirección | 89,94 | 80 |
| | Confirmación | 90,16 | 80 |
| Dirección | Dirección | 88,37 | 71,34 |
| | Confirmación | 83,16 | 83,16 |

Tabla 8. Análisis de frases en-contexto y fuera-contexto ($p$=13)

Comparando los resultados medios correspondientes al mismo tipo de frases en las Tablas 5 y 8, se deduce que cuando el G-analizador procesa frases fuera-contexto, WA se incrementa en 87,74% (desde -0,27% hasta 86,98%) y SU se incrementa en 82,04% (desde 0% hasta 82,04%), es decir, la técnica propuesta permite que el sistema comprenda correctamente 8 de cada 10 frases fuera-contexto.

## *4 Conclusiones y trabajo futuro*

Los resultados experimentales muestran que el uso de la técnica propuesta mejora notablemente el funcionamiento de la nueva interfaz de entrada del sistema SAPLEN, en la que se usa una G-gramática y una reconocedor del habla a dos niveles. La técnica posibilita el procesamiento aceptable de frases fuera-contexto. Dicha conclusión es clara al comparar los resultados de los análisis fuera-contexto mostrados en las Tablas 5 y 8. La información proporcionada por las tuplas de clases de palabras es muy importante durante el análisis de los GPs. La Tabla 6 muestra que los valores de WA y SU se incrementan conforme aumenta el valor de $\Delta_p$; no obstante, un valor excesivo para este parámetro (mayor que 13 para el corpus de frases usado) provoca una distorsión en el análisis de los GPs que conlleva la disminución de los resultados obtenidos. La Tabla 7 muestra que esta tendencia es distinta cuando se analizan frases fuera-contexto, dado que en este caso, el porcentaje de transiciones incrementadas es muy reducido.

Las líneas de trabajo futuro están relacionadas con estudiar formas alternativas que mejoren la forma en que el G-analizador incrementa las probabilidades de transición de los GPs. El uso de bigramáticas de clases de palabras facilita el procedimiento empleado; no obstante, posibilita que en ocasiones se incremente indebidamente la probabilidad de determinadas transiciones. El procedimiento empleado se podría mejorar incluyendo reglas sintácticas y semánticas para decidir si se debe incrementar o no las probabilidades. Por ejemplo, una regla sintáctica indicaría no incrementar la probabilidad de la transición *"dos"* → *"bocadillo"* pues no existe concordancia de número entre ambas palabras, y una regla semántica indicaría no incrementar la probabilidad de la transición *"cerveza"* → *"tinto"* pues el producto *"cerveza tinto"* no tiene sentido en el dominio de aplicación del sistema. No obstante, las reglas semánticas serían dependientes del dominio de aplicación y, por consiguiente, deberían ser adaptadas si éste cambia.

## *Referencias*


López-Cózar R., Rubio A. J., Benítez M. C., Milone D. H. Restricciones de funcionamiento en tiempo real de un sistema de diálogo. Procesamiento del Lenguaje Natural, nº 24, pág. 169-174

Hain T., Woodland P.C., Niesler T.R., Whittaker E.W.D.; 1999; The 1998 HTK System for Transcription of Conversational Telephone Speech; Proc. of International Conference on Acoustics, Speech and Signal Processing

Hidaki Y., Kobayashi T. 1999. Combination of word-bigram and class-bigram for topic-robust language modeling". IPSJ SIG-SLP Note, vol. 98, nº 12, pág. 25-32

Huang X., Acero A., Hon H. 2001. Spoken Language Processing. A Guide to Theory, Algorithm and System Development, Prentice Hall

Rabiner L. R., Juang B. H. 1993. Fundamentals of Speech Recognition. Prentice-Hall