

## *Un Paciente Simulado Virtual Multilingüe para la formación en medicina*

**Víctor López, Eduardo M. Eisman, María Navarro, Jose Manuel Zurita, Juan Luis Castro**

Departamento de Ciencias de la Computación e Inteligencia Artificial  
Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación  
Universidad de Granada  
C / Periodista Daniel Saucedo Aranda s/n. C.P. 18071. Granada, España  
{victor,eisman,marianj,zurita,castro}@decsai.ugr.es

**Resumen:** Muchas universidades instruyen a sus estudiantes para adquirir ciertas habilidades antes de que estos se enfrenten a un entorno profesional. En el caso de la medicina, se destina una importante cantidad de recursos económicos a instruir actores para que juegue el papel de pacientes que sufren una determinada enfermedad frente a los estudiantes. Además de ser costoso, este proceso requiere que el alumno y el actor coincidan físicamente. Para solventar estas y otras limitaciones, presentamos un Paciente Simulado Virtual Multilingüe, un agente conversacional que simula un paciente real que acude a una consulta de atención primaria. Los estudiantes pueden entrevistar al agente tratando de diagnosticar qué le ocurre, tal y como lo haría un médico en una consulta real. Este paciente virtual es capaz de comunicarse en múltiples idiomas y expresar diferentes estados de ánimo dependiendo de la enfermedad que padece y el comportamiento del estudiante.

**Palabras clave:** Agente conversacional, paciente virtual, ontología, generación de lenguaje natural, razonamiento basado en casos, sistema basado en reglas difusas.

**Abstract:** Many universities instruct their students to acquire certain abilities before they face a professional environment. In the case of medicine, an important amount of economic resources is dedicated to instruct actors to play the role of patients suffering a certain illness before the students. Besides being expensive, this process requires that the student and the actor meet physically. In order to solve this and other limitations, we present a Multilingual Virtual Simulated Patient, a conversational agent simulating a real patient who goes to a primary care consultation. Students can interview the agent trying to diagnose what the matter is, in the way that a doctor would normally do in a real consultation. This virtual patient is able to communicate using different languages and to express different moods depending on the illness he or she suffers from and the behaviour of the student.

**Key words:** Conversational agent, virtual patient, ontology, natural language generation, case-based reasoning, fuzzy rule-based system.

### **1. Introducción**

La simulación de las situaciones a las que una persona se enfrentará en el mundo real es un método para la mejora de sus habilidades cuya eficacia ha sido probada en diferentes ámbitos: sistemas de tutoría [Graesser 05], juegos conversacionales [Rehm 05], agentes que simulan diferentes roles en un

entorno profesional [Kopp 05], etc. En el campo de la medicina existen sistemas inteligentes que ayudan a los estudiantes a desarrollar sus habilidades deductivas y cooperativas [Woo 06], pero no existen sistemas para entrenar las habilidades necesarias para comunicarse de forma efectiva con los pacientes, las cuales resultan fundamentales en una consulta de atención primaria.

La importancia de esta formación previa se pone de manifiesto mediante el hecho de que muchas universidades y otros centros y compañías relacionadas con la formación destinan importantes recursos económicos a instruir actores que desempeñen el papel de pacientes que sufren una determinada enfermedad. Sin embargo, esta metodología, aunque es útil, presenta ciertos inconvenientes tanto para el estudiante como para el centro de aprendizaje. En primer lugar, los estudiantes sólo pueden practicar cuando se lleva a cabo alguna acción formativa concreta en el centro educativo, ya que está limitada a la disponibilidad de los actores. Además, la calidad del proceso de aprendizaje puede variar entre diferentes actores e incluso entre varias actuaciones de un mismo actor. Por este motivo resulta interesante el desarrollo de un sistema que permita a los estudiantes mejorar su formación en cualquier momento, a la vez que ahorrar gran cantidad de recursos y asegurar la calidad del proceso de aprendizaje.

Un sistema inteligente de este tipo debe tener una serie de características. En primer lugar debe estar representado gráficamente mediante un paciente que sea capaz de expresar diferentes estados de ánimo. Además, debe ser capaz de comunicarse en lenguaje natural con el alumno, reaccionando adecuadamente al transcurso de la conversación. También debe permitir la creación de pacientes con diferente edad, sexo, personalidad y enfermedades, y que cada uno se comporte de forma acorde a sus características.

La propuesta que presentamos en este artículo es la utilización de un agente conversacional basado en un modelo de paciente que permite la definición de síntomas y enfermedades y su asignación a pacientes virtuales. Mediante el módulo reconocedor de lenguaje natural, el agente es capaz de dotar de contenido semántico a la pregunta realizada por el alumno. Esta representación semántica de las preguntas es utilizada por el gestor de diálogo para controlar el comportamiento del agente y generar una respuesta, también semántica, utilizando un sistema basado en reglas. Esta respuesta semántica incluye el estado emocional del paciente, gestionado por el controlador del estado emocional. Finalmente, el generador de lenguaje natural traduce esta respuesta semántica a una respuesta en lenguaje natural que pueda ser entendida por el alumno.

El contenido del resto del artículo es el siguiente. En la sección 2 mencionaremos brevemente algunos de

los trabajos relacionados con el tema. Más tarde, en la sección 3, describiremos la arquitectura del Paciente Simulado Virtual, entrando en detalle en cada uno de los módulos que lo componen. A continuación, dedicaremos la sección 4 al aspecto de la localización, es decir, la adaptación del paciente virtual a diferentes idiomas y culturas. En la sección 5 mostraremos algunos resultados de las pruebas de pilotaje llevadas a cabo. Finalmente, en la sección 6 estableceremos las conclusiones de nuestro trabajo.

## 2. Trabajo relacionado

Los agentes conversacionales pueden ser utilizados en cualquier situación donde exista una interacción entre humanos, con el agente interpretando el papel de cualquiera de ellos. Como ya hemos comentado, los agentes conversacionales han sido aplicados a muy diferentes campos como un guía turístico para un museo [Kopp 05] o un tutor virtual para estudiantes de primer curso de física [Graesser 05]. Hasta donde nosotros conocemos, este tipo de sistemas ha sido tradicionalmente poco empleado en el campo de la simulación de pacientes para la formación en entornos médicos.

[Dickerson 05] presenta un agente conversacional cuyas respuestas y comportamiento está escrito en un sistema basado en scripts similar a AIML [Wallace 05]. Estos scripts capturan de antemano las típicas preguntas y movimientos de diálogo que un médico puede utilizar para pacientes con una enfermedad específica, así como las propias respuestas del agente. Presentan descriptores (frases del médico) y acciones que son típicas en cada situación (por ejemplo, señalar). El comportamiento del agente se lleva a cabo seleccionando el script con el descriptor más similar a la entrada del usuario previamente normalizada. Esta solución necesita escribir scripts para todas las situaciones específicas de una consulta con un paciente. Como podemos imaginar, el esfuerzo es muy alto ya que no permite especificar por separado el conocimiento sobre el modelo de los pacientes, los síntomas, el comportamiento cognitivo y emocional, y el conocimiento lingüístico.

[Kenny 07] describe un paciente virtual para el entrenamiento de las habilidades de terapia clínica. Posee un modelo separado de comportamiento no verbal y un esquema de selección estadístico de la respuesta del agente con algunas consideraciones

sobre posibles situaciones de diálogo, como preguntas sin relación al tema, peticiones realizadas por el usuario al agente como “¿Podría repetir eso?”, o respuestas alternativas a preguntas como “¿Tiene algo más que añadir?”. Aunque utiliza reglas para asignar el comportamiento no verbal al agente dependiendo de la respuesta seleccionada, al igual que antes, este trabajo carece de un modelo para relacionar los síntomas de las enfermedades con el comportamiento del paciente.

Tal y como se indica en publicaciones más recientes [Kenny 11] [Carpenter 12], existe un creciente interés en la aplicación de este tipo de sistemas para la formación en medicina. Esto demuestra que estamos ante un campo de investigación abierto en el que probablemente se produzcan importantes avances en los próximos años.

En relación al proceso de generación de lenguaje natural en sí mismo, en líneas generales existen tres aproximaciones, las cuales se sitúan en orden ascendente de complejidad y generalidad: texto enlatado, plantillas, y aproximaciones simbólicas que emplean representaciones del conocimiento a diferentes niveles lingüísticos y reglas para manipularlas.

El texto enlatado tiene la ventaja de ser una aproximación simple. Solamente necesita el texto final que se va a generar, pero tiene el inconveniente de no ser reutilizable.

Las plantillas poseen una visión más abstracta a la hora de generar el lenguaje natural, mezclando texto prefijado con texto variable. Un ejemplo de un sistema clásico que utiliza esta aproximación es ELIZA [Weizenbaum 66]. Este sistema inserta en sus respuestas parte de la entrada del usuario para simular el proceso de un psicoterapeuta haciendo terapia. Esta es una técnica de generación de lenguaje natural más general, ya que no necesita generar de antemano todas las respuestas del sistema. Sin embargo, estas plantillas no se podrían reutilizar en otras situaciones diferentes a aquellas para las que han sido creadas.

El último tipo de aproximación normalmente emplea conocimiento lingüístico como gramáticas u operadores retóricos [Mann 05] para describir la parte del lenguaje utilizada por el sistema. Esto lo hace más genérico, aunque eleva su complejidad.

Normalmente los agentes conversacionales centran su metodología de generación de lenguaje natural en el uso de plantillas. Este esquema es claramente insuficiente para establecer el contenido de la

respuesta, ya que el agente no puede decir lo que quiere, sino solamente la respuesta asociada a una entrada del usuario.

Existen alternativas como ProtoPropp [Gervas 05] que utilizan razonamiento basado en casos para generar historias a partir de una descripción inicial del argumento. Los casos son argumentos completos compuestos por movimientos relacionados. Un movimiento es una clase de procedimiento que permite seguir la historia hasta un cierto punto, es decir, hasta que se ha alcanzado un objetivo en el argumento. El módulo de razonamiento basado en casos recupera un caso similar a partir de la consulta de entrada, la cual contiene una descripción parcial de la historia. El caso recuperado es adaptado haciendo sustituciones con las restricciones de la consulta de entrada, buscando movimientos compatibles en una ontología para adaptar el caso. A continuación, se pasa la estructura resultante a un módulo de generación de lenguaje natural que selecciona el contenido a incluir en la historia descartando la información ya presentada. Además, el discurso se estructura estableciendo un orden de presentación entre los hechos de la historia en base a prioridades, y se realiza una agregación de los hechos que hablan del mismo tema o la misma acción. Por último, se seleccionan las palabras adecuadas para representar las entidades salientes de la historia, y se expresa el argumento en instancias de las plantillas en lenguaje natural.

Otra alternativa es PERSONAGE [Mairesse 10], un generador de lenguaje natural parametrizable capaz de producir textos con diferentes estilos asociados a aspectos de la personalidad. El sistema consta de varios módulos: un planificador de contenido que selecciona el contenido de las frases y estructura el discurso utilizando operadores retóricos, un planificador de frases que especifica cómo se tienen que organizar las frases, y un módulo de realización que produce el texto final. Cada uno de estos módulos tiene parámetros que controlan algunos aspectos del tipo de texto a generar. Por ejemplo, para el planificador de contenido, doce parámetros influyen en el tamaño del plan de contenido, el orden del contenido, las relaciones retóricas utilizadas, y la polaridad de las proposiciones expresadas.

A pesar de los diferentes enfoques existentes, una gran cantidad de autores coinciden en que uno de los objetivos deseables para la generación de lenguaje natural en un agente conversacional es mantener la

metodología empleada tan simple como sea posible. Normalmente, los expertos en lingüística computacional son recursos costosos o no están disponibles. Si la técnica empleada es lo suficientemente genérica, este módulo podría reutilizarse para otros agentes dedicados a diferentes dominios e idiomas sin tener que desarrollar otro sistema de generación de lenguaje natural específico para cada dominio e idioma.

Uno de nuestros objetivos dentro del proyecto del Paciente Simulado Virtual era desarrollar un sistema de generación de lenguaje natural que no necesitase un conocimiento lingüístico profundo y que pudiese reutilizarse en muchos dominios sin tener que cambiar la funcionalidad del sistema. Los detalles concretos sobre la arquitectura elegida para el sistema se muestran en la próxima sección.

### 3. Arquitectura del Paciente Simulado Virtual

El Paciente Simulado Virtual es un agente conversacional que utiliza una ontología que almacena el conocimiento asociado a las posibles características de un paciente: edad, sexo, tipo de personalidad, síntomas, intensidades, frecuencias, localizaciones del cuerpo humano, enfermedades, palabras en lenguaje natural que se utilizan para referirse a ellos, etc. Consta principalmente de cuatro módulos: el reconocedor de lenguaje natural, el controlador del estado emocional, el gestor de diálogo y el generador de lenguaje natural.

Adicionalmente, es posible conectar otros dos módulos disponibles comercialmente: un módulo ASR (*Automatic Speech Recognition*) para el reconocimiento del habla (de forma que no sea necesario escribir las preguntas para el paciente) y un módulo TTS (*Text To Speech*) para que el paciente sea capaz de reproducir de forma oral sus respuestas.

A lo largo de esta sección iremos analizando con detalle la estructura de la ontología y cada uno de los módulos que componen la arquitectura del sistema. Sin embargo, para ir introduciéndonos en el tipo de respuestas proporcionadas por el Paciente Simulado Virtual, a continuación mostramos un ejemplo de conversación real con el médico.

**Médico** | Hola Sr. García, entre y siéntese, por favor.

**Paciente** | Hola doctor.

**Médico** | ¿Qué le ocurre?

**Paciente** | Tengo un fuerte dolor de cabeza en la parte izquierda y a veces noto entumecidas la lengua y las manos.

**Médico** | ¿Desde cuándo ha sentido ese dolor?

**Paciente** | Las dos últimas semanas.

**Médico** | ¿Tiene fiebre?

**Paciente** | No, no tengo.

**Médico** | ¿Ha notado inflamación en las manos?

**Paciente** | No, aunque a veces se me hincha el ojo izquierdo.

**Médico** | De acuerdo... tómese esto durante cuatro días y si el dolor y la inflamación no disminuyen, vuelva de nuevo.

**Paciente** | Muchas gracias, doctor.

#### 3.1. Ontología

La ontología del Paciente Simulado Virtual integra, mediante una jerarquía de clases, el conocimiento del agente conversacional en relación a tres niveles: el dominio del problema y sus relaciones, las características del agente, y los procesos (reglas y casos utilizados por los módulos del sistema). Asimismo, la ontología determina el lenguaje empleado por el paciente, especificando plantillas de respuesta que son personalizadas mediante palabras asociadas a cada entidad del dominio. Entre las clases más importantes podemos destacar las siguientes:

**Agente.** Registra los agentes creados y su estado interno. Incluye propiedades como personalidad (ansioso, depresivo, hipocondríaco, maníaco, fóbico y normal), sexo (hombre o mujer), edad, estado emocional (alegría, desdén, ira, miedo, preocupación, sorpresa, tristeza y vergüenza) y memoria (para guardar la conversación hasta el momento). Puede heredarse para especificar el conocimiento de dominio de un agente conversacional concreto. Por ejemplo, la subclase Paciente contiene el modelo de paciente mientras que la subclase Usuario contiene el modelo de usuario. Los pacientes pueden presentar varios modelos psicológicos que muestran la intensidad normal de cada síntoma en un paciente

saludable. Existen modelos tanto para hombres como para mujeres, y tres rangos de edad diferentes (adulto, niño y anciano) que resultan en seis tipos de modelos de paciente. Cuando se crea un paciente se le asignan algunas enfermedades, y esto altera su modelo normal de paciente añadiendo las intensidades de los síntomas, incluyendo cierto ruido para hacer que el paciente sea más variable. Este modelo hace que nuestro sistema sea fácil de extender con nuevos pacientes y enfermedades. Actualmente el modelo de usuario sólo contiene las frases que este dice, pero podría contener otro conocimiento añadiendo las propiedades adecuadas.

**ContenidoSemantico.** Especifica el conocimiento del dominio. Contiene los términos léxicos de entrada, útiles para especificar algunas palabras en lenguaje natural o expresiones que se utilizan para reconocer las preguntas realizadas por el usuario, y los términos léxicos de salida, que contienen palabras o frases que describen una entidad a un nivel léxico y que se utilizan posteriormente en la generación en lenguaje natural de la respuesta. En el Paciente Simulado Virtual estas clases están relacionadas con enfermedades, síntomas (dolor, mareos, malestar general, signos vitales y características físicas del paciente como frecuencia cardíaca, pérdida de memoria, consciencia...), localizaciones (cabeza, ojo, brazo, pecho...), intensidades, frecuencias, duraciones y actividades. Estas clases son parte del modelo de enfermedad que podría sufrir un paciente. Cada síntoma tiene una intensidad normalizada relacionada, que puede ser positiva o negativa, desde dos semanas antes del día de la cita, y quizás alguna actividad o localización relacionadas. Los síntomas son los componentes de las enfermedades, por lo que en nuestra ontología hemos definido un conjunto de síntomas que caracterizan un gran conjunto de enfermedades, tal y como han indicado expertos de la Fundación IAVANTE [IAVANTE 13].

**Estado.** Conjunto que agrupa reglas de dos tipos: reglas de estado, que crean la respuesta semántica del agente, y reglas de transición, que establecen cuándo pasar de un estado a otro.

**Regla.** Entidad utilizada por el gestor de diálogo para controlar el comportamiento del agente (crear la respuesta semántica, cambiar de estado, evaluar alguna instancia de la ontología...).

**ObjetivoComunicativo.** Expresa la intención comunicativa de la respuesta, como por ejemplo “Saludar” o “Responder”. Posee una prioridad que indica el orden entre objetivos. Cada objetivo puede ser alcanzado por una o más AccionComunicativa.

**AccionComunicativa.** Representa diferentes tipos de acciones como “Informar” o “Quejarse”. Posee una prioridad que indica el orden entre acciones.

**RestriccionSemantica.** Indica el contenido semántico de una AccionComunicativa.

**Frase.** Plantilla simple que representa una frase en lenguaje natural. Contiene restricciones que indican cuándo puede ser elegida para formar parte de la respuesta final del agente. Se compone de una AccionComunicativa, una RestriccionSemantica, un campo de texto que contiene la plantilla de texto, y cero o más RestriccionDeFrase, utilizadas para rellenar los huecos de la plantilla.

**RestriccionDeFrase.** Representa una realización concreta en lenguaje natural de una entidad de ContenidoSemantico. Se utiliza para rellenar los huecos de una plantilla de texto en una Frase. Por ejemplo, para la entidad *Humano*, una restricción de frase contendrá *Humano* como su referencia a entidad de dominio, y “*persona*” como su realización en lenguaje natural. Otra podría ser *Humano* → “*homo sapiens*”, o incluso *Humano* → “*ser humano*”. Además del nivel léxico asociado a una entidad, cada restricción de frase puede tener algunas restricciones que indiquen el tipo de situación a la que se puede aplicar. Podría tener restricciones morfológicas (género, número...), restricciones lógicas relacionadas con propiedades de la entidad (por ejemplo, una instancia de *Humano* sólo puede ser etiquetada como “*anciano*” si la restricción “*Humano.edad > 70*” es verdadera), o restricciones ad hoc creadas específicamente para agrupar algunas restricciones de frase.

**Caso.** Plantilla compleja compuesta de un ObjetivoComunicativo y una o más instancias de Frase ordenadas.

Merece la pena recalcar que esta ontología es genérica y no está dedicada a un dominio específico.

Sin embargo, como hemos visto, las clases pueden ser heredades para crear entidades de algún dominio concreto que permitan crear un agente con el conocimiento necesario para mantener una conversación relacionada con ese dominio.

### 3.2. Reconocedor de lenguaje natural

El proceso de reconocimiento de una frase dentro de un diálogo general es una tarea bastante difícil. Sin embargo, puede ser simplificada si nos restringimos al lenguaje y las características de un dominio concreto. Si nos fijamos en una consulta entre un médico y un paciente, podemos observar que el diálogo es estructurado, y típicamente está formado por un saludo, algunas frases sobre la enfermedad del paciente, un diagnóstico y una despedida. De esta forma podemos establecer una serie de acciones en la ontología para representar los diferentes tipos de frases del usuario (ver Tabla 1). Esto permite analizar y caracterizar el lenguaje que se utiliza normalmente.

Acción*	Frase de ejemplo
PreguntaIntensidadSL	“¿Cómo es su dolor de brazo?”
PreguntaFrecuenciaSL	“¿Cuántas veces le duele?”
PreguntaDuracionSL	“¿Desde hace cuánto ha notado entumecimiento en las manos?”
PreguntaIntensidadS	“¿Se encuentra muy cansado?”
PreguntaFrecuenciaS	“¿Cuántas veces ha perdido la consciencia?”
PreguntaDuracionS	“¿Desde cuándo se siente enfermo?”
PreguntaMotivoConsulta	“¿Qué le ocurre?”
AccionSaludo	“Hola”
AccionDespedida	“Adiós”
AccionDiagnostico	“Tiene migrañas”
AccionSilencio	Utilizada cuando el médico no dice nada durante un tiempo

\* Las acciones se nombran según el contenido de su restricción semántica: S = Síntoma, L = Localización

**Tabla 1.** Ejemplos de acciones y preguntas

Con este conjunto de acciones, el reconocedor de lenguaje natural es capaz de hacer interpretaciones. Para ello, busca correspondencias entre las palabras de la pregunta y las categorías semánticas de la ontología, asignando una interpretación a las entidades descubiertas en el paso previo. Las palabras de las frases que no identifican ninguna clase o instancia de la ontología son ignoradas.

Así, la pregunta “¿Cómo es su dolor de pecho?”, se traduciría en las siguientes entidades semánticas:

“cómo es” = PreguntaIntensidadSL, “pecho” = pecho, “dolor” = dolor. La interpretación se construiría rellenando la restricción semántica de la acción con las entidades pecho y dolor. Por tanto, la salida del reconocedor sería básicamente la siguiente:

```
pregunta: PreguntaIntensidadSL (restSem:
RestriccionSemanticaSL (dolor: Sintoma,
pecho: Localizacion), 1)
```

En el caso de que se pueda seleccionar más de una acción posible, se calcula una puntuación entre 0 y 1 para cada una en base a la completitud de la interpretación con respecto a las entidades encontradas en la traducción de la frase del usuario. Así, el reconocedor de lenguaje elegirá la que tenga una mayor puntuación, siempre y cuando supere un umbral establecido. Por el contrario, si ninguna interpretación supera el umbral, el reconocedor devolverá un mensaje de *NoReconocimiento* al resto de módulos del sistema.

Una descripción más detallada de este proceso se puede encontrar en [López 08].

### 3.3. Controlador del estado emocional

Como ya hemos comentado, el Paciente Simulado Virtual debe ser capaz de reaccionar emocionalmente durante el transcurso de la consulta. De ello se encarga el controlador del estado emocional, que se basa en dos conceptos esenciales, el estado emocional y la personalidad. Siguiendo las aproximaciones [Ekman 84] y [Yanaru 10], el estado emocional se construye en base a un conjunto de emociones básicas, a las que llamamos atributos emocionales. Estos atributos representan las emociones más relevantes que puede sentir un paciente real. El valor de cada atributo es un número real comprendido entre 0 (ausencia total) y 1 (presencia total). La Tabla 2 muestra dos ejemplos de estado emocional, que pueden pertenecer a dos pacientes diferentes o al mismo paciente en diferente estado de tiempo.

	Atributo Emocional							
	Ale	Des	Ira	Mie	Pre	Sor	Tri	Ver
EE1	0.73	0.26	0.12	0.08	0.11	0.39	0.04	0.10
EE2	0.05	0.03	0.07	0.48	0.39	0.10	0.86	0.01

Leyenda: EE: Estado Emocional, Ale(gría), Des(dén), Mie(do), Pre(ocupación), Sor(presa), Tri(steza), Ver(güenza)

**Tabla 2.** Algunos estados emocionales

Por su parte, la personalidad es un conjunto de características estáticas definidas para cada uno de los ocho atributos emocionales. Al igual que estos, los tipos de personalidad (ver Tabla 3) han sido determinados con la ayuda de especialistas. Estos tipos cubren, a priori, todas las personalidades que se suponen adecuadas para un paciente.

	Atributo Emocional							
	Ale	Des	Ira	Mie	Pre	Sor	Tri	Ver
<b>Ansioso</b>	0.0	0.0	0.0	0.6	0.6	0.0	0.0	0.0
<b>Depresivo</b>	0.0	0.0	0.0	0.0	0.3	0.0	0.6	0.4
<b>Hipocond.</b>	0.0	0.0	0.5	0.7	0.0	0.0	0.0	0.0
<b>Maníaco</b>	0.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0
<b>Fóbico</b>	0.0	0.0	0.0	0.6	0.3	0.0	0.0	0.0
<b>Normal</b>	0.1	0.0	0.0	0.1	0.1	0.1	0.1	0.2

Leyenda: Hipocond(riaco), Ale(gría), Des(dén), Mie(do), Pre(ocupación), Sor(presa), Tri(steza), Ver(güenza)

**Tabla 3.** Tipos de personalidad

Cuando se asigna una personalidad a un paciente concreto, se lleva a cabo una pequeña variación en el valor de los atributos emocionales asociados a esa personalidad, de manera que dos pacientes que compartan la misma personalidad no tienen por qué ser exactamente iguales.

Cada atributo emocional se actualiza de forma independiente utilizando un sistema basado en reglas difusas. Las variables de entrada de estos sistemas se refieren al tipo de personalidad del paciente, su salud (si está enfermo o cansado en principio debería ser propenso a enfadarse con otros), el tipo de conversación (si la pregunta está relacionada con el motivo de la consulta o un síntoma en particular, es de índole sexual...), las características del síntoma sobre el que pregunte el médico (intensidad, frecuencia, duración, y peso), etc.

La salida de estos sistemas es un nivel específico de variación a producir en cada atributo emocional (negativo alto, negativo medio, negativo bajo, cero, positivo bajo, positivo medio, positivo alto, y personalidad). Las reglas con variación cero, cuando se disparan, reducen el impacto que tienen sobre el estado emocional del paciente el resto de reglas disparadas. Por otro lado, las reglas con variación personalidad son dinámicas, lo que significa que su nivel de variación específico cambia a lo largo de las diferentes iteraciones del sistema de acuerdo a la personalidad y el estado emocional actual.

De esta forma, una regla difusa tiene dos componentes, el antecedente con las variables de entrada (parte izquierda de la regla) y el consecuente con la variable de salida (parte derecha). A continuación se puede apreciar un ejemplo de regla para el atributo emocional *alegría*.

```
Personalidad:!Depresivo
TipoPregunta:ComentarioPositivoMedico
EnMemoria:Cero
→ Variacion:PositivaBaja
```

Esta regla se interpreta de la siguiente manera: SI el paciente *no* tiene una *personalidad depresiva*, Y el médico hace un *comentario positivo* por *primera vez*, ENTONCES se produce una *variación positiva baja* en el atributo emocional correspondiente (*alegría* en ese caso).

Para saber si una regla se dispara o no, se asignan los valores asociados a cada variable de entrada. A continuación, se evalúa cada expresión difusa del antecedente, y finalmente se agregan las salidas de todas las reglas disparadas para obtener una variación concreta del atributo emocional.

Además, cada regla tiene un valor de probabilidad asociado a ella que hace que el sistema sea no determinista. Esto resulta en pacientes más naturales y diferentes ya que ni todos van a ser iguales, ni el mismo paciente va a comportarse siempre igual ante la misma situación. Por otro lado, el sistema ajusta, en tiempo de ejecución, tanto las probabilidades asociadas a cada regla como los niveles de variación de los consecuentes de las reglas dinámicas, de manera que el comportamiento del paciente puede ser guiado por una determinada conducta.

Una descripción más detallada de este proceso se puede encontrar en [Eisman 09].

### 3.4. Gestor de diálogo

El gestor de diálogo es un sistema basado en reglas que utiliza los individuos de las clases Estado y Regla de la ontología para crear una máquina de estados finitos. De esta forma, una vez actualizada la memoria de trabajo con las entidades producidas por el reconocedor de lenguaje natural, el gestor de diálogo comprueba si existe alguna regla de transición que haga cambiar el estado actual. En ese caso, se elige uno de los estados que se han disparado y se realiza la transición.

A continuación, se utilizan las reglas de estado del estado actual para generar la respuesta semántica del paciente eligiendo entre los consecuentes de las reglas disparadas utilizando un mecanismo basado en votos. Cada regla tiene un número de votos que se utiliza para elegir de forma aleatoria entre ellas e introducir así cierta variabilidad en el sistema. Así, los pacientes generados son más naturales en el sentido de que responden con más variabilidad a las preguntas del usuario.

La Figura 1 muestra la estructura general del gestor de diálogo. Los arcos son las reglas de transición con las frases del médico como antecedentes. Además, cada uno de los cinco estados tiene reglas que dan una respuesta semántica apropiada en un momento determinado de la conversación.

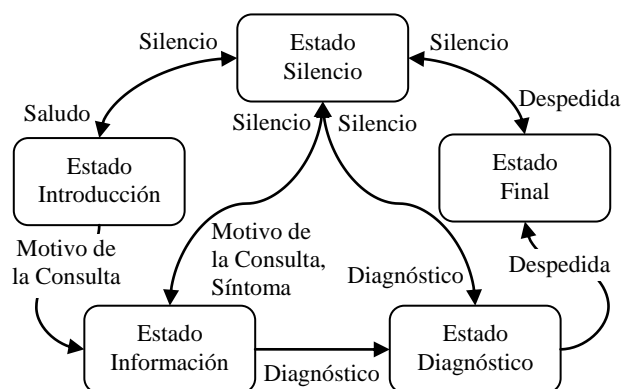


Figura 1. Estados de conversación del PSV

A modo de ejemplo, mostramos una regla simplificada que casaría con la interpretación de la pregunta “¿Dónde le duele?”:

```

ReglaRespuestaPLS:
[pregunta: PreguntaLocalizacionS (?restSem:
RestriccionSemanticaS (?sintoma: Sintoma))],
[encuentraLocalizacion(?sintoma, ?paciente,
?varL)]
-> dar_informacion: DarInformacion
(especificar: Especificar(rest_sem:
RestSemSL (?sintoma: Sintoma, ?varL:
Localizacion)))
    
```

Los símbolos que comienzan con el carácter ? son variables. Aquí podemos apreciar la acción *PreguntaLocalizacionS*, que es la interpretación de la pregunta, y un método *encuentraLocalizacion* que permite buscar la localización del síntoma *?sintoma* en el modelo de paciente *?paciente*,

almacenándola en *?varL*.

La respuesta semántica generada representaría lo que conceptualmente quiere decir el paciente, y podría ser traducida por el generador de lenguaje en la respuesta de texto “*Me duele el ojo izquierdo*”.

### 3.5. Generador de lenguaje natural

El generador de lenguaje natural transforma la respuesta semántica proporcionada por el gestor de diálogo en una respuesta textual que pueda ser entendida por el alumno. Se basa en un proceso de razonamiento basado en casos. Estos casos se almacenan en la ontología (como instancias de la clase Caso) y se procesan en tres etapas: indexación, recuperación y adaptación.

**Indexación.** Se realiza una sola vez al comienzo del proceso. Ordena los casos en base a tres índices, primero según su objetivo, después según el número de frases que contienen, y finalmente, para los casos con una sola frase, según la acción que contienen.

**Recuperación.** Utiliza la base de casos indexada para buscar un caso que cumpla con las restricciones de la respuesta semántica proporcionada por el gestor de diálogo. Para poder recuperar un caso, su objetivo, y las acciones y restricciones de sus frases, deben coincidir con los de la respuesta semántica. Si no existe ningún caso con el número de frases necesario, se parte del conjunto de casos con el mayor número de frases menor que el número de acciones. A continuación, de entre los casos posibles se selecciona uno aleatoriamente y se borran las restricciones correspondientes de la respuesta semántica. Esta fase de recuperación se repite hasta que no quedan restricciones en la respuesta semántica.

**Adaptación.** Una vez que se han recuperado los casos, estos pasan por un proceso de adaptación en el que se rellenan los huecos de la plantilla de texto de cada frase. Para ello, se utilizan las palabras asociadas al nivel léxico de salida de las restricciones semánticas incluidas en el caso.

Supongamos por ejemplo que el gestor de diálogo devuelve la siguiente respuesta semántica:



```
motivo: ObjetivoMotivoConsulta (  
    especificar: Especificar(rest_sem:  
        RestSemISL(alta: Intensidad, dolor:  
        Sintoma, ojoizquierdo: Localizacion)),  
    especificar1: Especificar(rest_seml:  
        RestSemFSSL(frecmed: Frecuencia,  
        entumecer: Sintoma, lengua: Localizacion,  
        manos: Localizacion))  
)
```

El generador de lenguaje buscará un caso que tenga el objetivo `ObjetivoMotivoConsulta` y dos frases para intentar satisfacer esas dos restricciones. Imaginemos que se recupera un caso `CasoMotivo1` cuya plantilla es `"#FraseFuerteDolorEnLoc#` y `#FraseFrecSint2Loc#"`. Si expandimos cada frase con el contenido de su plantilla de texto, tendríamos las frases `"Tengo un #RIntAlta# #RSintDolor# en #RLoc#"` y `"#RFrecMed# noto #RSintParaNoto# #RLoc1# y #RLoc2#"`. Sustituyendo cada restricción del caso por las entidades exactas especificadas en la respuesta semántica, obtendríamos la respuesta final del paciente, que en este caso sería *"Tengo un fuerte dolor en el ojo izquierdo y a veces noto entumecidas la lengua y las manos"*.

Si la respuesta semántica hubiese requerido recuperar otro caso adicional como por ejemplo `CasoEnfermedadResfriado`, cuya plantilla asociada podría ser *"Creo que he cogido un resfriado"*, la respuesta final del paciente se construiría uniendo las dos respuestas, es decir, *"Tengo un fuerte dolor en el ojo izquierdo y a veces noto entumecidas la lengua y las manos. Creo que he cogido un resfriado"*.

Una descripción más detallada de este proceso se puede encontrar en [López 12].

#### 4. Localización

Gracias al proyecto europeo *"Multilingual Virtual Simulated Patient (MVSP)"*, cuya referencia puede encontrarse en la sección Agradecimientos, se llevó a cabo un proceso de localización para adaptar el modelo de Paciente Simulado Virtual anteriormente comentado a diferentes idiomas y culturas.

Además del idioma español, empleado inicialmente para construir y probar el modelo, seis socios de seis países europeos diferentes rellenaron la base de casos

con su propio conocimiento lingüístico. Estos seis nuevos idiomas fueron el inglés, el alemán, el italiano, el húngaro, el búlgaro y el portugués. Para cada idioma se hicieron dos versiones, una para paciente nativo y otra para paciente no nativo representando a la minoría étnica de cada país (su forma de hablar, de expresarse...).

Aunque existía algún socio tecnológico, el perfil de la mayoría era fundamentalmente sanitario, ya que su tarea era crear el diálogo para el paciente utilizando las frases que ellos estimaban que serían más apropiadas y se emplearían con mayor naturalidad en cada situación prevista en el contexto de un diálogo entre un médico y un paciente.

Por este motivo se desarrolló un editor que ayudase a los socios a introducir, en la base de casos, todo el conocimiento lingüístico (casos, frases, restricciones de síntomas, localizaciones, intensidades, frecuencias, y duraciones).

Más detalles sobre este proyecto pueden encontrarse en [López 12] y en la página oficial [MVSP 13].

#### 5. Pruebas del sistema

Las pruebas de validación del sistema se realizaron en dos fases. La primera fase estuvo destinada a la validación del modelo de paciente en español. Posteriormente, en una segunda fase se validó cada uno de los pacientes de los diferentes idiomas creados dentro del proyecto MVSP.

La validación del modelo de paciente se llevó a cabo en nuestro grupo de investigación. El paciente era capaz de reconocer 6.592 tipos diferentes de preguntas relacionadas con síntomas. El gestor de diálogo tenía 45 reglas de estado (la mayoría en el Estado Información), y disparaba, de media, 1,57 reglas para cada pregunta. Para crear una respuesta, el generador de lenguaje natural tenía 279 casos y recuperaba, de media, 9,92 casos para cada respuesta semántica.

Se realizaron pruebas con siete diálogos con una media de 9,4 frases. El módulo de reconocimiento de lenguaje natural tuvo un 69,7% de interpretaciones correctas. Dentro de las incorrectas, el 16,7% de las frases no reconocidas se debía a que el alumno había utilizado palabras que no habían sido tenidas en cuenta para referirse a entidades del dominio. El resto eran errores de interpretación debido al uso de frases multi-objetivo (errores que se podrían reducir si el

reconocedor de lenguaje pudiese crear más de una acción a la vez), al uso de frases con palabras que designaban varias entidades, es decir, sinónimos (para lo que habría que añadir más reglas), o a palabras referidas a entidades de frases previas como anáfora o deixis (para lo cual el reconocedor de lenguaje debería utilizar el contexto del diálogo).

Con respecto al controlador del estado emocional, se compararon las variaciones de los atributos emocionales producidas por cada frase de los siete diálogos con las reacciones que la mayoría de la gente debería tener en esas situaciones. En el 71,8% de los casos, el comportamiento emocional generado fue el esperado, aunque los niveles de variación de las reglas deberían incrementarse de forma que se pudiera alcanzar el 100% de cada atributo emocional durante un diálogo completo (algunos atributos no llegaron a alcanzar el 65%). El 29,2% restante correspondía a pequeñas variaciones opuestas a las esperadas. Por ejemplo, el atributo ira experimentaba varias fluctuaciones cuando se preguntaba sobre varios síntomas que el paciente no sufría, algo que debería evitarse en el futuro.

Por todo ello, llegamos a la conclusión de que, a pesar del simple esquema de reconocimiento de lenguaje natural y de las pocas reglas de comportamiento, nuestro modelo de agente conversacional se comportaba razonablemente bien, con naturalidad y variabilidad, en las pruebas llevadas a cabo.

Durante la fase de validación del proyecto europeo, cada socio realizó una sesión de pilotaje en su propio idioma con una interfaz similar a la que aparece en la Figura 2.



Figura 2. Interfaz de comunicación con el MVSP

En la prueba participaron principalmente estudiantes de primer y segundo año de medicina. Su tarea consistía en realizar una entrevista clínica de diez minutos a un paciente virtual que padecía cefalea de Horton (hecho que desconocían). El objetivo de la entrevista era averiguar la enfermedad del paciente realizando preguntas en lenguaje natural. Después de la prueba, los alumnos tenían que rellenar un cuestionario sobre varios aspectos del MVSP como qué les había parecido la interacción con el paciente, su aspecto visual, o la calidad del lenguaje empleado (credibilidad, naturalidad y realismo). Este último aspecto fue valorado en una escala del 0 al 5, y el resultado para cada idioma es el que se muestra en la Figura 3.

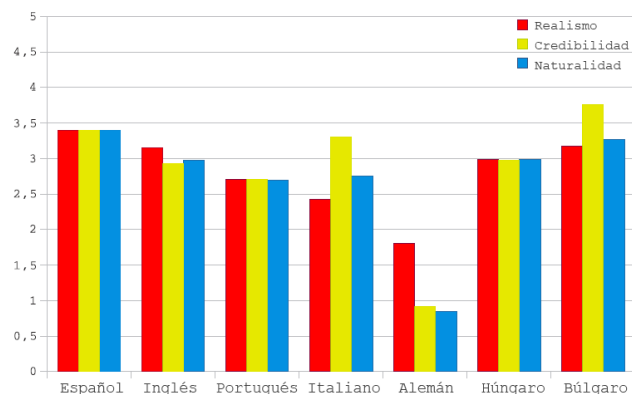


Figura 3. Resultados de las pruebas del MVSP

El caso más llamativo es el del idioma alemán. Algunas de las personas que probaron esta versión del paciente indicaron que a veces este proporcionaba o demasiada cantidad o muy poca información en un solo turno. Incluso, las respuestas proporcionadas, aunque eran gramaticalmente correctas y posibles, en ocasiones no resultaban demasiado usuales en la situación propuesta. Sin embargo, este no es problema del modelo de generación en sí mismo, ya que es incremental y permite la mejora de los resultados con sólo introducir más casos. Además, debido a las limitaciones de tiempo del proyecto, la mayoría de los casos introducidos por los socios eran de una sola frase y sin oraciones complejas.

## 6. Conclusiones

En este artículo hemos presentado el Paciente Simulado Virtual, un agente conversacional

específico para la formación de estudiantes de medicina en el entorno de una consulta de atención primaria, aunque el sistema es fácilmente extensible a otros dominios ya que todos los módulos han sido diseñados orientados al conocimiento y son controlados por la información contenida en una ontología.

El sistema consta principalmente de cuatro módulos. El primero de ellos es el reconocedor de lenguaje natural, que se encarga de crear una representación semántica de la pregunta realizada por el alumno. Esta representación semántica de la pregunta es utilizada por el gestor de diálogo para generar una respuesta semántica (abstracta) apropiada para el estado actual en el que se encuentra la conversación, el cual es controlado mediante una máquina de estados y una serie de reglas. Esta respuesta semántica incluye el estado emocional del paciente, gestionado por el controlador del estado emocional mediante un sistema basado en reglas difusas, el cual dota al paciente de una mayor naturalidad. Por último, el generador de lenguaje natural utiliza la técnica de razonamiento basado en casos para recuperar y adaptar casos de la base de casos que cumplan las restricciones especificadas por la respuesta semántica del gestor de diálogo con el objetivo de crear una respuesta textual entendible por el alumno.

Asimismo, gracias al proyecto europeo “*Multilingual Virtual Simulated Patient*”, ha sido posible la adaptación del Paciente Simulado Virtual a otros idiomas como el inglés, el alemán, el italiano, el húngaro, el búlgaro y el portugués, teniendo incluso en cuenta las características de las minorías étnicas de cada país.

Los resultados muestran que, a pesar del simple esquema de reconocimiento de lenguaje natural y de las pocas reglas de comportamiento incluidas inicialmente, nuestro modelo de paciente es capaz de comportarse razonablemente bien, con naturalidad y variabilidad.

### Agradecimientos

En primer lugar, nos gustaría agradecer a la Consejería de Innovación, Ciencia y Empresa de la Junta de Andalucía (TIC-P06-01424), al Ministerio de Ciencia y Tecnología de España (TIN2007-67984-C02-01), a la Unión Europea (programa transversal

KA3-ICT: Lifelong Learning Programme, Call for Proposals 2008, EAC-30/07, Referencia: 143423-2008-LLP-ES-KA3-KA3MP), y al programa Torres Quevedo del Ministerio de Ciencia e Innovación de España (PTQ-10-03942) por el apoyo financiero mostrado durante estos años.

Igualmente, nos gustaría agradecer a la Fundación IAVANTE la creación del modelo de síntomas y enfermedades del paciente y su ayuda para seleccionar los atributos emocionales, las personalidades, y las reglas difusas del controlador del estado emocional del Paciente Simulado Virtual. Por último, queremos agradecer a la Fundación CITIC su ayuda en la programación del módulo de reconocimiento de lenguaje natural.

### Referencias

- [Carpenter 12] C. Carpenter, L. Osterberg, G. Sutcliffe. “SAMHT — Suicidal Avatars for Mental Health Training”. Proceedings of the Twenty-Fifth International Florida Artificial Intelligence Research Society Conference. Association for the Advancement of Artificial Intelligence (www.aaai.org). 2012.
- [Dickerson 05] R. Dickerson, K. Johnsen, A. Raij, B. Lok, J. Hernandez, A. Steens. “Evaluating a script-based approach for simulating patient-doctor interaction”. In Proceedings of International Conference on Human-Computer Interface Advances for Modeling and Simulating, pages 79–84. 2005.
- [Eisman 09] E. M. Eisman, V. López, J. L. Castro. “Controlling the emotional state of an embodied conversational agent with a dynamic probabilistic fuzzy rules based system”. Expert Systems with Applications, Volume 36, Issue 6, Pages 9698–9708. August 2009.
- [Ekman 84] P. Ekman, K. R. Scherer. Approaches to Emotion. 1984.
- [Gervas 05] P. Gervas, B. Diaz-Agudo, F. Peinado, R. Hervas. “Story plot generation based on cbr”. Knowledge-Based Systems, 18, 235–242. 2005.
- [Graesser 05] A. C. Graesser, P. Chipman, B. C. Haynes, A. Olney. “Autotutor: An intelligent tutoring system with mixed-initiative dialogue”.

- IEEE Transactions on Education, 48, 612–618. 2005.
- [IAVANTE 13] Fundación IAVANTE. <http://www.iavante.es/>. 2013.
- [Kenny 07] P. Kenny, T. D. Parsons, J. Gratch, A. Leuski, A. A. Rizzo. “Virtual patients for clinical therapist skills training”. Lecture Notes in Artificial Intelligence, 4722:197–210. 2007.
- [Kenny 11] P. G. Kenny, T. D. Parsons. “Embodied Conversational Virtual Patients”. Conversational Agents and Natural Language Interaction: Techniques and Effective Practices. IGI Global, 2011. pp 254-281. 2011.
- [Kopp 05] S. Kopp, L. Gesellensetter, N. C. Kramer, L. Wachsmuth. “A conversational agent as museum guide – design and evaluation of a real-world application”. In Panayiotopoulos, T., Gratch, J., Aylett, R., Ballin, D., Olivier, P., & Rist T., (Eds.), Intelligent Virtual Agents, Proceedings. Lecture Notes in Artificial Intelligence. 5<sup>th</sup> International working conference on intelligent virtual agents September 12–14, Kos, Greece (Vol. 3661, pp. 329–343). 2005.
- [López 08] V. López, E. M. Eisman, J. L. Castro. “A tool for training primary health care medical students: The virtual simulated patient”. In 20th IEEE international conference on tools with artificial intelligence proceedings (Vol. 2, pp. 194–201). 2008.
- [López 12] V. López, E. M. Eisman, J. L. Castro, J. M. Zurita. “A case based reasoning model for multilingual language generation in dialogues”. Expert Systems with Applications Volume 39, Issue 8, Pages 7330–7337. 15 June 2012.
- [Mairesse 10] F. Mairesse, M. Walker. “Towards personality-based user adaptation: psychologically informed stylistic language generation”. User Modelling and User-Adapted Interaction, 20, 227–278. 2010.
- [Mann 05] W. Mann, S. A. Thompson. “Rhetorical structure theory: Toward a functional theory of text organization”. Text, 8(3), 8, 243–28. 2005.
- [MVSP 13] Multilingual Virtual Simulated Project. <http://www.mvsp.eu/>. 2013.
- [Rehm 05] M. Rehm, M. Wissner. “Gamble – A multiuser game with an embodied conversational agent”. In F. Kishino, Y. Kitamura, H. Kato, & N. Nagata (Eds.), Entertainment computing – ICEC 2005, Lecture Notes in Computer Science. 4th international conference on entertainment computing (ICEC 2005) September 16–21, 2005 Sanda, Japan (Vol. 3711, pp. 180–191). 2005.
- [Wallace 05] R. S. Wallace and A. A. I. Foundation. “Artificial Intelligence Markup Language (AIML)”. <http://www.alicebot.org/aiml.html>. 2005.
- [Weizenbaum 66] J. Weizenbaum. “Eliza – A computer program for study of natural language communication between man and machine”. Communications of the ACM, 9, 36–45. 1966.
- [Woo 06] C. W. Woo, M. W. Evens, R. Freedman, M. Glass, L. S. Shim, Y. Zhang, Y. Zhou, and J. Michael. “An intelligent tutoring system that generates a natural language dialogue using dynamic multi-level planning”. Artificial Intelligence in Medicine, 38(1):25–46. 2006.
- [Yanaru 97] T. Yanaru, N. Shirahama, K. Yoshida, and M. Nagamatsu. “An emotion processing system based on fuzzy inference and subjective observations”. Information Sciences, 101(3 - 4):217–247. 1997.