

CHARLIE: Un robot conversacional como interfaz de una plataforma de tele-educación

Fernando A. Mikic Fonte

Departamento de Ingeniería Telemática
E.T.S.I. Telecomunicación, Universidad de Vigo
Vigo, España
mikic@det.uvigo.es

Juan C. Burguillo Rial

Departamento de Ingeniería Telemática
E.T.S.I. Telecomunicación, Universidad de Vigo
Vigo, España
jrial@det.uvigo.es

Martín Llamas Nistal

Departamento de Ingeniería Telemática
E.T.S.I. Telecomunicación, Universidad de Vigo
Vigo, España
martin@det.uvigo.es

David Fernández Hermida

Departamento de Ingeniería Telemática
E.T.S.I. Telecomunicación, Universidad de Vigo
Vigo, España
davidfh@gist.det.uvigo.es

INES (INtelligent Educational System) es un prototipo operativo de una plataforma de tele-educación que combina funcionalidades concernientes a un LMS (Learning Management System), un LCMS (Learning Content Management System), y un ITS (Intelligent Tutoring System). Para llevar a cabo todas estas funcionalidades, nuestro sistema en su conjunto engloba toda una serie de herramientas y tecnologías, como pueden ser: herramientas para la gestión semántica de usuarios y contenidos, un robot conversacional inteligente (comúnmente conocido como *chatterbot*), un agente inteligente basado en tecnología BDI (Believes, Desires, Intentions) que actúa como el cerebro del sistema, un motor de inferencia basado en JESS (motor de reglas para la plataforma Java) y una ontología (para modelar a los usuarios, los contenidos educativos, y sus relaciones). En el presente artículo nos centraremos en el *chatterbot*, CHARLIE (CHatteR Learning Interface Entity), desarrollado y utilizado en la plataforma, el cual es un robot basado en tecnología AIML (Artificial Intelligence Markup Language).

CHARLIE; INES; AIML; chatterbots; inteligencia artificial, agentes; BDI; tele-educación; LMS; LCMS; ITS

I. INTRODUCCIÓN

INES (INtelligent Educational System) es un prototipo operativo de una plataforma de tele-educación que nuestro grupo de investigación está desarrollando. Dicha plataforma combina las que creemos que deben ser tres funcionalidades esenciales relacionadas con las tareas que una plataforma de este estilo tiene que desempeñar, y que no son otras que aquellas llevadas a cabo por un LMS [1] (Learning Management System), un LCMS [2] (Learning Content Management System), y un ITS [3] (Intelligent Tutoring System).

Un LMS es una aplicación software instalada en un servidor, la cual se utiliza para gestionar, distribuir, y supervisar todas las tareas educativas de una organización o institución. Sus principales funciones son: gestionar usuarios, recursos, y actividades y materiales educativos, controlar el

acceso, supervisar el proceso y el progreso educativo, realizar evaluaciones, etc. Un LMS a menudo no incluye capacidades de autoría, es decir, el poder crear sus propios contenidos educativos, lo cual normalmente recae sobre un LCMS.

Un LCMS se utiliza para crear y gestionar los contenidos de una parte de un programa educativo (por ejemplo un curso), el cual puede ser usado, gestionado, y personalizado de muy diferentes formas (por ejemplo en diferentes cursos).

Por último, un ITS es un sistema de apoyo educativo (una especie de tutor virtual), que se encarga de prestar ayuda a los estudiantes en sus tareas de aprendizaje, y de suministrarles contenidos específicos, personalizados, y adaptados a sus propias aptitudes.

Para llevar a cabo todas estas funcionalidades, nuestro sistema en su conjunto engloba toda una serie de herramientas y tecnologías, como pueden ser: herramientas para la gestión semántica de usuarios (administradores, profesores, alumnos...) y contenidos, un robot conversacional inteligente (comúnmente conocido como *chatterbot*) capaz de comunicarse con los estudiantes en lenguaje natural, un agente inteligente basado en tecnología BDI (Believes, Desires, Intentions) que actúa como el cerebro del sistema, un motor de inferencia basado en JESS (motor de reglas para la plataforma Java) y una ontología (para modelar a los usuarios, los contenidos educativos, y sus relaciones) que contribuyen a la parte semántica del sistema, etc.

En el presente artículo nos centraremos en el *chatterbot*, CHARLIE (CHatteR Learning Interface Entity), desarrollado y utilizado en la plataforma, el cual es un robot basado en tecnología AIML (Artificial Intelligence Markup Language) [4]; y en particular, en su funcionamiento y su contribución a INES.

El resto del artículo está organizado de la siguiente manera: La siguiente sección estará dedicada a presentar nuestra plataforma de tele-educación INES, y en la sección III nos

centraremos específicamente en el *chatterbot* CHARLIE, para terminar el artículo con unas conclusiones y líneas futuras.

II. INES (INTELLIGENT EDUCATIONAL SYSTEM)

INES es un prototipo operativo de una plataforma de tele-educación, la cual combina tres capacidades esenciales relacionadas con las actividades de aprendizaje en línea. Estas capacidades son aquellas pertenecientes a un LMS, a un LCMS, y a un ITS.

Por tanto, INES es capaz de llevar a cabo todo un conjunto de tareas específicas de estos tres tipos de sistemas, las cuales resumiremos a continuación:

- Administración y gestión de cursos y usuarios.
- Creación, gestión, y distribución de contenidos.
- Tutorización y ayuda al alumno.

Las partes principales de las que se compone INES se pueden agrupar en una serie de bloques (ver Figura 1):

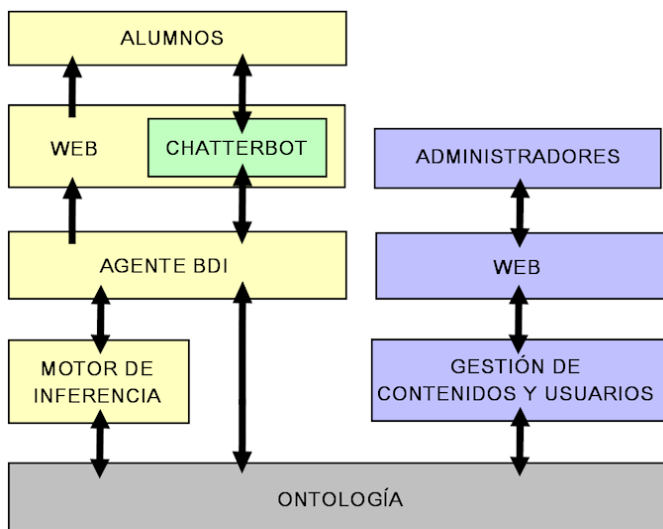


Figura 1. Diagrama de bloques de INES

- Ontología: Actualmente existe una ontología, la cual se subdivide en otras tres subontologías principales, para definir semánticamente los contenidos de los cursos (objetos de aprendizaje), los usuarios de la plataforma, y las relaciones entre ellos. La primera de ellas está basada en LOM [5] y la segunda en IMS-LIP [6].
- Módulo de gestión de contenidos y usuarios: Este módulo permite a los administradores gestionar tanto a los usuarios del sistema como los contenidos de los cursos.
- Motor de inferencia: El cual procesa las peticiones del agente BDI y decide lo que se permite hacer y lo que no.
- Agente BDI: El auténtico cerebro del sistema. Está basado en tecnología BDI [7], y es el responsable de

tomar decisiones personalizadas sobre el aprendizaje de cada estudiante de manera inteligente.

- Chatterbot: Responsable de la comunicación con los estudiantes (y sobre el que nos centraremos en la próxima sección).

III. CHARLIE (CHATTER LEARNING INTERFACE ENTITY)

CHARLIE (CHAtter Learning Interface Entity) es un *chatterbot* que realiza tareas de interfaz entre INES y los estudiantes, es decir, es la parte de nuestro sistema que se comunica directamente con los estudiantes en lenguaje natural.

A. Características Generales

Nuestro *chatterbot* ha sido desarrollado usando la tecnología de A.L.I.C.E. [8], con lo cual su cerebro está compuesto por un conjunto de ficheros AIML, que son simples módulos estímulo-respuesta. Está basado en Program D [9] y utiliza tecnología AJAX (Asynchronous JavaScript And XML) [10], la cual permite construir aplicaciones interactivas o RIA (Rich Internet Applications). Esta técnica permite a nuestro robot mantener una comunicación asíncrona con el servidor en segundo plano, con lo cual es posible realizar cambios en la página web sin necesidad de recargarla. Esto supone una mejora sustancial de la interactividad, la velocidad, y la facilidad de uso.

Dentro de nuestra plataforma de tele-educación (INES), el *chatterbot* (CHARLIE), lleva a cabo las funciones de interfaz de usuario, es decir, le ofrece al estudiante la posibilidad de comunicarse con el sistema en lenguaje natural. En lo que se refiere a la arquitectura de CHARLIE en el nivel de abstracción más alto, nuestro *chatterbot* consta de una BUI (Bot User Interface), un intérprete, y una base de datos AIML. La Figura 2 muestra dicha arquitectura con los principales elementos que toman parte en el sistema.

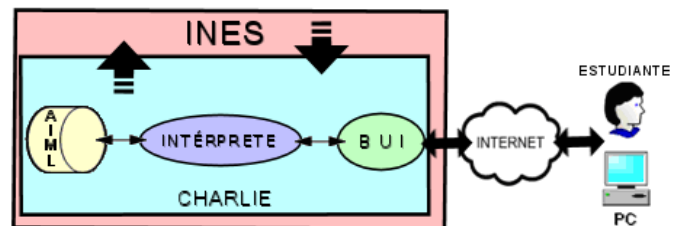


Figura 2. Arquitectura de CHARLIE como parte de INES

Los estudiantes son capaces de conectarse a la plataforma de tele-educación a través de Internet e interactuarán con el *chatterbot* a través de la BUI (que no es más que una caja de texto donde los estudiantes pueden introducir lo que quieran decir). El *chatterbot* recoge los datos de entrada de esta BUI y decide qué hacer a continuación mediante una búsqueda en su base de conocimiento, pudiendo surgir dos casos:

- El *chatterbot* encuentra en su base de conocimiento contenido adecuado para responder al alumno, y le responde.

- El *chatbot* encuentra en su base de conocimiento instrucciones para enviar la entrada del usuario al sistema y esperar una respuesta del mismo. En el momento en que reciba esta respuesta, la procesará y le responderá al usuario.

B. Funcionalidades de CHARLIE

Para explicar en detalle las funcionalidades de CHARLIE consideraremos dos puntos de vista: el punto de vista del estudiante y el de un administrador.

1) *Punto de vista del estudiante*: El *chatbot* le ofrece al estudiante la oportunidad de interactuar con la plataforma de tele-educación en lenguaje natural. Tal como mencionamos anteriormente, esta interacción entre INES y los estudiantes se lleva a cabo a través de una ventana emergente con un área de texto donde se muestra la conversación mantenida, y una caja de texto donde se introduce la entrada del estudiante.

De esta manera, los estudiantes son capaces de mantener una conversación con el *chatbot*, la cual puede ser una conversación en general (gracias a un conjunto de ficheros AIML predefinidos que contienen información general) o una conversación específica relacionada con el contenido de un curso.

Si el *chatbot* no detecta ninguna entrada relacionada con el contenido de un curso, le responderá al estudiante con una expresión contenida en su base de conocimiento general. En cambio, si el *chatbot* detecta una entrada del alumno en la cual ha utilizado alguna de las palabras clave relacionadas con el contenido de un recurso, el *chatbot* recuperará la asociación que previamente haya definido el administrador y tomará las medidas oportunas.

Además, mientras un estudiante mantiene una conversación con el *chatbot*, puede solicitar ciertas actividades relacionadas con test:

- Solicitar la realización de un test: El *chatbot* escoge uno de los test que el administrador haya creado para que sea realizado por el alumno.
- Solicitar la realización de un test personalizado: El alumno debe escoger el número de preguntas que serán incluidas en el test y el *chatbot* creará un test que cumpla ese requisito.
- Solicitar preguntas sueltas (sin que tengan que formar parte de un test): El *chatbot* comenzará a realizarle preguntas al alumno, y seguirá así hasta que este le diga que pare.

2) *Punto de vista de un administrador*: El administrador es capaz de gestionar el *chatbot* a través de un módulo de gestión específico. Este módulo ofrece una interfaz amigable para seleccionar los ficheros AIML que serán cargados. Además, es posible personalizar el *chatbot* dándole un nombre, aficiones, ciudad, cumpleaños, etc., y una imagen de fondo para la ventana de conversación. Todo ello hará que el

chatbot parezca más humano a la hora de interactuar con los usuarios.

Otra tarea propia del administrador es cargar el contenido específico para un curso en la base de conocimiento del *chatbot*. Este contenido es aquel que (tal como comentamos anteriormente) hace que se envíe la entrada del usuario al sistema para ser procesada, dejando al *chatbot* en espera de una respuesta del sistema para entregar al usuario (el resto de la base de conocimiento del *chatbot* se utiliza para poder llevar una conversación general con el usuario). Para conseguir esto, CHARLIE cuenta con una interfaz donde el administrador puede seleccionar recursos de una estructura de módulos e insertar una serie de palabras clave para identificarlos. De esta manera, si el alumno utiliza alguna de estas palabras clave en su entrada el *chatbot* será capaz de llevar a cabo las acciones apropiadas.

El administrador puede introducir él mismo las palabras clave, pero en un principio un analizador sintáctico le recomendará algunas de manera automática. La primera vez que un conocimiento específico se asocia al *chatbot*, se lleva a cabo un análisis sintáctico que da como resultado un conjunto de palabras clave, es decir, el analizador sintáctico se utiliza para obtener una serie de palabras clave recomendadas relacionadas con el contenido del curso considerado. Para ello se utiliza el analizador sintáctico de código abierto FreeLing 1.5 [11].

Las razones que nos han llevado a utilizar este analizador son las siguientes: de código abierto, disponible en línea, versátil y convincente, fácil de instalar y de hacer funcionar, modular (lo cual permitirá posibles cambios del mismo en futuras versiones del *chatbot*), y soporta diferentes idiomas (lo cual permitirá posibles cambios del mismo en futuras versiones del *chatbot*).

Otra característica importante de CHARLIE es su capacidad para presentarles preguntas a los estudiantes, las cuales estarán relacionadas con recursos de la plataforma. Estas preguntas resultarán idóneas para ayudar a los alumnos a darse cuenta de qué partes de un curso son las más importantes, y para entrenarlos para los exámenes tipo test que deberán realizar a lo largo del curso.

Toda vez que una pregunta (y su respuesta asociada) se asigna a un recurso, será añadida a la base de datos AIML del *chatbot* (para ello, se han desarrollado un conjunto de funciones).

Una vez que una serie de preguntas ya han sido almacenadas, es posible definir test (ver Figura 3), los cuales consisten en un conjunto de preguntas seleccionadas de aquellas ya creadas. Las diferentes partes de la Figura 3 se detallan brevemente a continuación:

- Añadir un nuevo test: Para añadir un nuevo test al curso seleccionado.
- Test en el curso: Es una lista de los test que pertenecen al curso seleccionado. Desde aquí es posible cambiar el orden de cada test, acceder a él, y eliminarlo.
- Preguntas en el test "Nombre del test": Muestra las preguntas que conforman el test llamado "Nombre del

test". Desde aquí es posible cambiar el orden de aparición de cada pregunta dentro del test y eliminarlas.

- Banco de preguntas: Permite acceder a una serie de acciones a realizar sobre las preguntas: creación, edición, ordenación, eliminación, etc.

Añadir un Test nuevo

Nombre del Test:

Mostrar la nota al alumno: Sí No

Tests en este curso

Ordenar	#	Nombre del Test	Borrar Test
↓	1	Test 1	X
↑	2	Test 2	X

Preguntas en el test Test 2

Ordenar	#	Nombre del Test	Tipo
↓	1	Clases Java	☰ >>
↑	2	Funciones Java	☰ >>
↑	3	Herencias en Java	☰ >>
↑	4	Hilos en objetos Java	☰ >>

Banco de preguntas

Crear una pregunta nueva:

Acción	Nombre de la pregunta	Ordenar por tipo, nombre	Tipo
☒ X	Aplicaciones de los Applet	☰	☰
☒ X	Excepciones en CAML	☰	☰
☒ X	Operador Class	☰	☰
☒ X	Clases Java	☰	☰
☒ X	Listas in CAML	☰	☰
☒ X	Métodos	☰	☰
☒ X	Restricciones de los Applet	☰	☰

Seleccionar todos / Omitir todos Con seleccionadas:

Figura 3. Gestión de test

Por tanto, el administrador tiene la capacidad de crear y gestionar tanto preguntas como test asociados a un curso, así como especificar el orden en el cual las preguntas aparecerán dentro de un test determinado o el orden en el cual los test relacionados con un curso serán propuestos al usuario.

IV. CONCLUSIONES Y LÍNEAS FUTURAS

La contribución más importante del presente artículo es la presentación de un prototipo funcional de plataforma de tele-

educación (llamada INES), la cual incluye capacidades propias de sistemas LMS, LCMS, e ITS; y más específicamente, la presentación de su interfaz de usuario. Esta interfaz es un tipo de robot conversacional (comúnmente conocidos como *chatterbots*), llamado CHARLIE, el cual es un agente basado en AIML capaz de comunicarse con los usuarios en lenguaje natural.

CHARLIE es capaz de mantener una conversación general con los alumnos, mostrándoles contenidos relacionados con un curso, y haciéndoles preguntas sobre el material educativo presentado.

Las principales líneas futuras relacionadas con CHARLIE consisten básicamente en realizar una actualización de su base de conocimiento para dar soporte a contenido semántico. Esto conllevará consigo una evolución del lenguaje AIML utilizado y el desarrollo de un nuevo intérprete (posiblemente llamado Program G). Además, intentaremos conseguir que CHARLIE sea capaz de comunicarse en diferentes idiomas.

AGRADECIMIENTOS

Este trabajo ha sido financiado por el Ministerio de Educación a través del proyecto "Servicios Adaptativos para E-learning basados en estándares" (TIN2007-68125-C02-02), y por la Consellería de Innovación e Industria de la Xunta de Galicia (Programa de Promoción General de la Investigación del Plan Gallego de IDIT) a través del proyecto "E-BICS: E-learning – Bases de Integración e Coordinación sobre eEstándares" (PGIDIT06PXIB32 2270PR). También queremos agradecer a la acción de coordinación del CYTED código 508AC0341 "Software Libre en Teleformación" y al Ministerio de Ciencia e Innovación de España y al Plan Nacional Español I+D+I 2008-2011 el apoyo a este artículo dentro del proyecto RedOBER - Proyecto TSI2007-31091-E Objetos Educativos Reutilizables (para el EEES en las especialidades de las Tecnologías de la Información y las Comunicaciones).

REFERENCIAS

- [1] A. Grace and T. Butler, *Beyond Knowledge Management: Introducing Learning Management Systems*, Idea Group Publishing, 2005.
- [2] W. K. Horton, *Designing Web-Based Training: How to Teach Anyone Anything Anywhere Anytime*, John Wiley and Sons, 2000.
- [3] T. Murray, "Authoring Intelligent Tutoring Systems: An Analysis of the State of the Art", *International Journal of Artificial Intelligence in Education*, 10, 98-129, 1999.
- [4] A. M. M. Neves, I. Diniz, and F.A. Barros, "Natural Language Communication via AIML Plus Chatterbots", *V Symposium on Human Factors in Computers Systems (IHC 2002)*, Fortaleza – CE, 387, 2002.
- [5] LOM (Learning Object Metadata). (2002). Draft Standard for Learning Object Metadata. Available at: http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf.
- [6] M. Norton, J. Treviranus, *IMS Learner Information Package Information Model Specification*, IMS Technical Report, 2001.
- [7] M. E. Bratman, *Intention, Plans, and Practical Reason*, CSLI Publications, 1999.
- [8] A.L.I.C.E. Artificial Intelligence Foundation. [Online]. Available: <http://www.alicebot.org/.AIML>
- [9] Program D. (2002). *Getting Started with Program D*. Available at: <http://www.alicebot.org/resources/programd/readme.html>.
- [10] R. Asleson and N. T. Schutta, *Foundations of AJAX*, Apress, 2005.
- [11] FreeLing. [Online]. Available: <http://garraf.epsevg.upc.es/freeling/>.