

Una arquitectura multi-agente para apoyar el uso de comunidades de práctica en las organizaciones

A Multi-Agent Architecture to Support Communities of Practice in Organizations

Juan Pablo Soto Barrera¹, Aurora Vizcaíno Barceló², Javier Portillo Rodríguez² y Mario Piattini Velthuis²

¹Departamento de Matemáticas Universidad de Sonora Hermosillo, México
jpsoto@gauss.mat.uson.mx

²Grupo de Investigación ALARCOS Universidad de Castilla – La Mancha Ciudad Real, España
{aurora.vizcaino, javier.portillo, mario.piattini}@uclm.es

Artículo recibido en Marzo 09, 2009; aceptado en Octubre 22, 2009

Resumen.

Este artículo presenta una arquitectura multi-agente diseñada para dar apoyo a las comunidades de práctica en organizaciones preocupadas por el intercambio de conocimiento. El principal objetivo de esta propuesta es facilitar el intercambio de conocimiento en empresas donde sus empleados están organizados en comunidades. Además de esto, este trabajo trata de apoyar a los miembros de la comunidad al momento de decidir en qué o quién confiar. Para esto, se propone un modelo de confianza basado en cuatro factores que las personas de manera consciente o inconsciente suelen considerar al momento de decidir si confían o no en un objeto o fuente de conocimiento.

Palabras Clave: Agentes inteligentes, Gestión del Conocimiento en Organizaciones, Confianza, Arquitecturas Multi-agente.

Abstract.

This paper presents a two layer multi-agent architecture designed to support communities of practice in organizations which are concerned about knowledge sharing. The main goal of this proposal is, therefore to facilitate knowledge exchange in organizations whose employees are organized into communities.

Keywords: Intelligent Agents, Knowledge Management in Companies, Trust, Multi-agent architecture

1 Introducción

En la actualidad, los Sistemas de Gestión de Conocimiento (SGC) han captado la atención de aquellas organizaciones preocupadas por su ventaja competitiva [Sher & Lee, 2004]. Este tipo de organizaciones constantemente intentan implementar mecanismos que apoyen y asistan a sus empleados durante el desarrollo de sus

actividades. Sin embargo, la mayoría de estos sistemas están enfocados en aspectos tecnológicos [Hahn & Subramani, 2000] dejando a un lado los aspectos sociales los cuales constituyen una parte importante en el desarrollo y uso de estos sistemas [He et al, 2009]. Un instrumento importante en la gestión del conocimiento son las comunidades [Gebert et al, 2003; Wasko & Faraj, 2000]. Actualmente no existe una definición consensuada de comunidad pero con frecuencia se define como un grupo de personas con un vínculo común y que regularmente se encuentran en un mismo lugar [Hillery, 1955]. El desarrollo de Internet y de herramientas *groupware* han propiciado el nacimiento de un nuevo tipo de comunidad “las comunidades virtuales” donde los miembros pueden o no conocerse personalmente y utilizan redes de comunicación para intercambiar ideas [Chiu et al, 2006; Geib et al, 2004; Koh & Kim, 2004]. Este tipo de comunidad se puede dividir según sus objetivos en orientadas a la sociedad, al negocio o a la organización. Nuestra investigación se centra en el último tipo que consiste básicamente en comunidades de empleados que comparten información con el objetivo de ayudarse en las tareas profesionales. Un caso especial de este tipo de comunidad son las Comunidades de Práctica (CoPs) definidas por Wenger (2000) como grupos de personas que comparten una inquietud, un conjunto de problemas o una pasión sobre un tema y aumentan sus conocimientos y experiencia gracias a su comunicación e intercambio de información. Desde un punto de vista de

conocimiento una CoP es un centro donde se comparten creencias y formas de hacer las cosas. Debido a esto, cada vez son más las organizaciones que están adoptando la idea de hacer uso de comunidades como parte de sus estrategias para gestionar el conocimiento [Davenport, 2001; du Plessis, 2008]. Para que los miembros de una comunidad puedan interactuar es muy importante que exista la confianza entre ellos mismos. Sin embargo, en la actualidad la mayoría de las comunidades son virtuales donde la comunicación cara a cara (“face-to-face”) desaparece y con ella algunas de sus ventajas como la confianza que surge al conocer personalmente a los miembros de la comunidad [Ridings et al, 2002; Hinds & McGrath, 2006]. Teniendo esto en cuenta, en este trabajo diseñamos una arquitectura multi-agente que apoye la gestión del conocimiento en CoPs con el fin de ayudar a sus integrantes a decidir qué fuentes de conocimiento son más confiables o recomendables de acuerdo a sus necesidades. Por eso, antes de describir la arquitectura, en la Sección 2 se describe el uso de agentes software como miembros de una CoP virtual. Por otra parte, en la Sección 3 se describe la arquitectura multi-agente propuesta. En la Sección 4 se describe el prototipo desarrollado para probar la funcionalidad de la arquitectura. En la Sección 5 se presenta la evaluación realizada al prototipo. En la Sección 6 se presenta el trabajo relacionado con nuestra propuesta. Finalmente, en la Sección 7 la conclusión y trabajo futuro son presentados.

2 Agentes Software en Comunidades de Práctica

Los agentes software han sido propuestos como soluciones al problema de sobrecarga de información [Atkinson-Abutridy & Ferreira-Cabrera, 1998; Maes, 1994]. El comportamiento autónomo de los agentes constituye un factor crítico para el objetivo de esta investigación ya que permite reducir la sobrecarga de trabajo a los empleados cuando utilizan un SGC. Otra ventaja importante del uso de esta tecnología es que los agentes tienen la capacidad de aprender de su propia experiencia. Es por esto, que conforme pasa el tiempo, los SMA son más eficientes ya que los agentes pueden aprender de sus propios errores y éxitos [Maes, 1994].

En este trabajo se optó por este paradigma ya que los agentes constituyen una metáfora natural para los sistemas que tratan de emular cómo las personas piensan o trabajan [Wooldridge & Ciancarini, 2001]. Así, se ha producido un incremento en el interés en usar agentes inteligentes para emular aspectos sociales tales como motivación, liderazgo o confianza [Fuentes et al, 2004]. Teniendo en cuenta el enfoque de nuestro trabajo, los agentes han sido organizados en comunidades donde sus miembros comparten intereses en temas de conocimiento similares. Las comunidades suelen dividirse de acuerdo a los objetivos que persiguen, por ejemplo, social, comercial y profesional. En este trabajo las CoPs están formadas por empleados que se comunican e intercambian información con el fin de mejorar sus procesos y de reutilizar lecciones aprendidas.

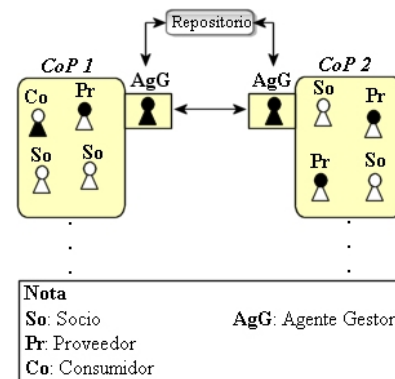


Fig. 1. CoPs formada por agentes

En la Figura 1, se ilustra la distribución de nuestros agentes en CoPs. En una organización pueden existir n cantidad de comunidades conformadas por dos tipos de agentes: Agente Usuario y Agente Gestor.

El Agente Usuario es el agente encargado de representar a la persona que consulta o introduce objetos de conocimiento en un repositorio o base de conocimientos. En este caso, un Objeto de Conocimiento (OC) puede ser, un documento de texto, un archivo de audio, video, etc. Por lo tanto, el principal objetivo del Agente Usuario es proveer información relevante a la persona que representa en el momento indicado, es decir, el agente debe conocer las necesidades de la persona y el tipo de información requerida para cumplir con sus tareas [Sánchez et al, 1998]. Teniendo en cuenta esta

idea, el Agente Usuario asume tres tipos de comportamientos o roles similar a las actividades de una persona que trabaja con SGC. Es decir, el Agente Usuario jugará un rol u otro dependiendo de las acciones ejecutadas por la persona que representa, las cuales pueden ser:

- Que la persona contribuya con un nuevo OC a la comunidad en la cual está registrado. En este caso el Agente Usuario juega el rol de Proveedor (Pr).
- La persona utilice un OC previamente almacenado en la comunidad. El Agente Usuario debe ser considerado como Consumidor (Co).
- La persona ayuda a otros usuarios a lograr sus objetivos, por ejemplo, proporcionando una evaluación a un OC. En este caso el rol del agente Usuario es el de Socio (So).

Como se ilustra en la Figura 1, en la comunidad 1, existen dos Agentes Usuarios jugando el rol de Socio, un Agente Usuario jugando el rol de Consumidor y otro jugando el rol de Proveedor.

Por otra parte, el Agente Gestor es el encargado de gestionar y controlar la comunidad. Las tareas del Agente Gestor son las siguientes:

- Registrar a los agentes en su comunidad. Esto le permite controlar el número de agentes registrados en la comunidad y el tiempo que llevan registrados en ella.
- Registrar la frecuencia de contribución de cada agente. Este valor es actualizado cada vez que el agente realiza una contribución a la comunidad.
- Registrar la frecuencia con que los agentes dan retroalimentación al conocimiento proporcionado por otros agentes. Por ejemplo, cuando un agente "A" utiliza un OC de otro agente "B", el agente A debe evaluar este OC. La monitorización del intercambio de conocimiento entre los agentes permite detectar si los agentes contribuyen a la creación de flujos de conocimiento en la comunidad ya que es importante que los agentes contribuyan tanto con nueva información como con la evaluación o relevancia de la información consultada.

Además del uso de la tecnología de agentes nuestro trabajo considera los conceptos de confianza y reputación, los cuales se representan a través de un modelo de confianza el cual permite determinar la fiabilidad de las fuentes de

información al momento de compartir conocimiento. El modelo de confianza que se propone en el apartado 3.2 de este trabajo considera aspectos sociales como la intuición y posición los cuales influyen directamente en la comunicación e interacción entre personas. Así, un conjunto de agentes inteligentes, utilizando valores de confianza, seleccionarán el conocimiento más útil para la persona que representan y le ahorrarán tiempo de búsqueda y selección de la información. De esta manera se pretende también fomentar la reutilización de la información almacenada en los SGC, los cuales a menudo no son consultados porque las personas no saben qué información hay almacenada y a veces desconocen cómo buscar o temen ser inundados de información que no sea valiosa para ellos [Vizcaíno et al, 2007].

3 Arquitectura multi-agente

El objetivo principal de este trabajo es el diseño de una arquitectura multi-agente que facilite el intercambio de conocimiento en empresas donde sus empleados están organizados en CoPs. La arquitectura propuesta se compone de un modelo cognitivo el cual considera dos niveles (ver Figura 2): reactivo y deliberativo-social.

La mayoría de los autores consideran los niveles reactivo y deliberativo como los niveles típicos que un Sistema Multi-agente (SMA) debe tener [Ushida et al, 1998]. Sin embargo, algunos trabajos como [Ferguson, 1992; Imbert, 2005] han considerado además el nivel social, dicho nivel se añade en aquellos sistemas que simulan comportamientos sociales o aquellos que representan arquitecturas más genéricas creadas para representar uno o varios comportamientos.

Debido a que la arquitectura se ha diseñado bajo el enfoque de CoPs consideramos fundamental contemplar aspectos sociales que permitan representar los comportamientos y reacciones de las personas ante los demás miembros de la comunidad. En un principio se consideró utilizar una capa para cada nivel (reactivo, deliberativo y social) [Soto et al, 2007]. Sin embargo, tras la implementación de un prototipo nos dimos cuenta que el nivel deliberativo y social estaban tan relacionados entre sí que una acción deliberativa por lo general implicaba una o varias acciones sociales lo que nos llevó a agruparlas en un solo nivel que hemos denominado "nivel deliberativo-social".

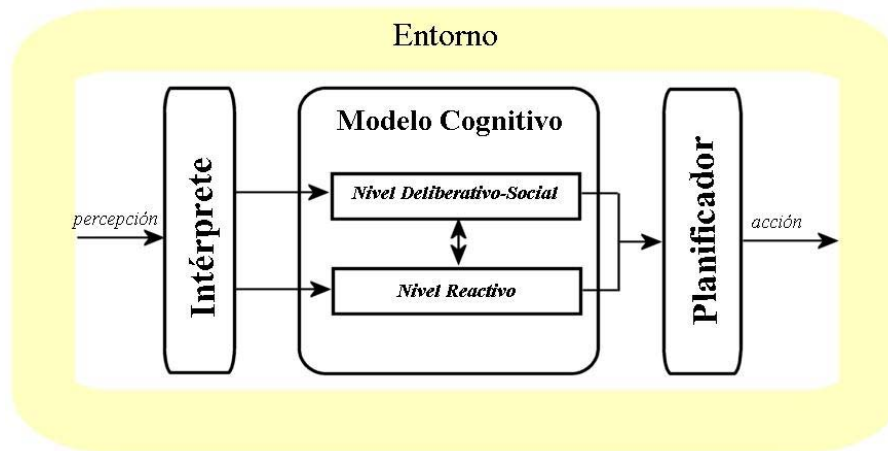


Fig. 2. Arquitectura general

La necesidad de este tipo de acciones sociales surge de situaciones en las que, aunque se conozca lo que se quiere hacer, hasta el momento mismo de la ejecución de las mismas no se puede determinar cómo se hará. Por ejemplo, en nuestro trabajo, una acción social puede ser: intercambiar información con el agente con mayor nivel de confianza en la comunidad. Para explicar la descomposición de esta acción social en acciones abstractas, se describe el escenario de la Figura 3:

Para poder ejecutar la acción social la cual consiste en consultar al agente con mayor confianza en la comunidad se debe ejecutar la siguiente serie de acciones abstractas:

- 1) En primer lugar, obtener la lista de agentes que pertenezcan a la comunidad.
- 2) El agente encargado de gestionar la comunidad responde con una lista que contiene los agentes registrados en la comunidad.
- 3) y 4) Después el agente debe determinar cuál es el agente con mayor nivel de confianza mediante su modelo de confianza.
- 5) y 6) Por último, interactuar con el agente con el nivel de confianza más alto en la comunidad.

Por tanto, es en el nivel deliberativo-social donde el agente determina la manera de interactuar con los demás agentes que integran la comunidad. Además, es en este nivel donde se comparte e intercambia la información entre los miembros de la comunidad y entre comunidades. En cambio en el nivel reactivo el agente determina la manera de actuar ante situaciones que requieran una respuesta inmediata y sin ningún tipo de razonamiento, es decir, en este nivel el agente utiliza información personal definida en su modelo interno sin considerar las opiniones y comportamientos de los demás agentes que integran el sistema.

Debido a que en nuestro trabajo intentamos emular el comportamiento de las personas en CoPs, es oportuno recordar que las personas que intervendrán en una comunidad suelen actuar de manera que sus acciones beneficien a la comunidad. Por tanto, el agente debe deliberar o decidir su forma de actuar en base a los objetivos que persiga la comunidad. Para esto, es necesario contar con un nivel que provea al agente un enfoque comunitario y global de las relaciones entre los agentes.

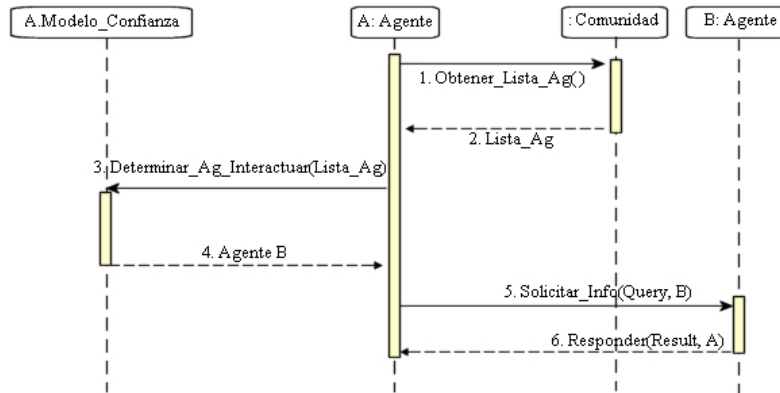


Fig. 3. Escenario que determina el agente con más confianza en la comunidad

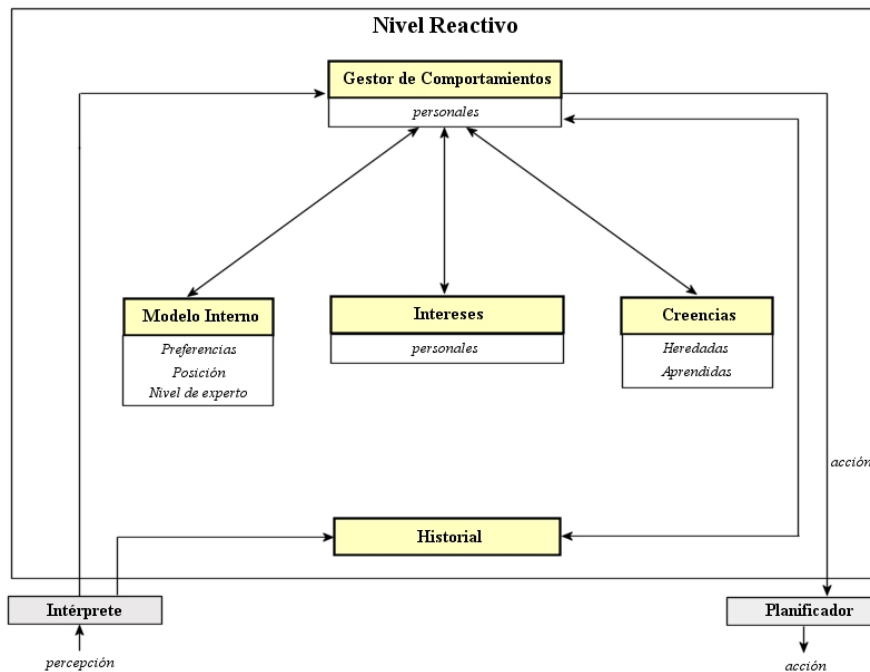


Fig. 4. Nivel reactivo

Teniendo en cuenta el enfoque de nuestro trabajo (CoPs) consideramos necesario contar además del nivel reactivo con un nivel que permita al agente interactuar por decisión propia en procesos de negociación o cooperación con otros agentes, este nivel ha sido llamado deliberativo-social. Debido al comportamiento que todo miembro de una comunidad debe tener, se considera importante que el agente pueda deliberar y tomar decisiones que beneficien a los demás agentes y cumpla con

los objetivos de la comunidad, los cuales se definen a continuación:

- Proveer la confianza necesaria para que sus miembros se sientan motivados en participar y compartir de forma abierta conocimiento útil.
- Identificar a los miembros expertos de la comunidad.

- Apoyar las relaciones interpersonales y de compañerismo promoviendo la discusión e intercambio de información.
- Proveer un banco de información y conocimiento que contenga las lecciones aprendidas y objetos de conocimiento generado en la comunidad.

Además, es en éste nivel donde el agente analiza la situación actual frente a los objetivos e intereses, desde la perspectiva de interacción con otros agentes. A diferencia de otras arquitecturas en nuestra propuesta el nivel deliberativo y social son considerados en un mismo nivel ya que se busca involucrar al agente en los procesos de razonamiento que se llevan en una comunidad. Para esto, el agente no sólo considera sus propios intereses sino que además debe tener en cuenta los propios intereses de la comunidad. La estrecha relación y dependencia del nivel deliberativo y social nos ha llevado a considerarlos en un mismo nivel.

En los siguientes sub-apartados se describe con detalle cada uno de los niveles que conforman la arquitectura propuesta.

3.1 Nivel Reactivo

El nivel reactivo es el nivel que le da la capacidad al agente para percibir los cambios generados en su entorno y responder a ellos en el momento en que éstos ocurren. Los componentes de este nivel son (ver Figura 4):

Modelo interno. Este módulo se compone de un conjunto de atributos que caracterizan a la persona que representa el agente en una CoP. Estos atributos son:

- Posición. Este factor puede definirse como el cargo o nivel jerárquico que ocupa una persona en la organización. Es importante hacer énfasis que una organización moderna constituye ante todo una composición de seres humanos; en donde es preciso que exista una estructura y una jerarquía necesaria para que se logren los fines que la organización se propone. Consideramos que es importante que este factor sea tenido en cuenta en las CoPs ya que afecta durante el intercambio de información entre las fuentes de conocimiento. Por ejemplo, en una CoP el conocimiento que proporciona el director de la empresa puede tener más valor o peso

que el aportado por una persona que pertenece a una escala más baja [De Long & Fahey, 2000; Wasserman & Galaskiewicz, 1994]. Sin embargo, esta no es una verdad universal ya que depende del tipo de situación. Por ejemplo, en un entorno de aprendizaje colaborativo es más probable que el aprendizaje se genere entre personas de estatus similares que entre un jefe y su empleado o incluso entre un profesor y sus alumnos [Dillenbourg, 1999]. Así la posición, es otro factor que puede influir en la confianza que se tiene en una fuente de conocimiento (por fuente de conocimiento consideramos tanto una persona o como un agente). El grado en que la posición puede influir en el nivel de confianza varía según el dominio donde se encuentre la comunidad y de las normas de la misma [De Long & Fahey, 2000]. En una organización este factor puede establecerse de distintas maneras, por ejemplo, utilizando un diagrama organizacional o clasificando la posición de los empleados de acuerdo al nivel que tenga la persona en la comunidad. Es importante tener en cuenta el número de niveles (NL) considerados por la comunidad, ya que de este número dependerá el valor que se le asigna a la posición de un agente en la comunidad. Una vez definidos los niveles que componen la comunidad (ver Tabla 1), se puede determinar el valor de la Posición (P) mediante la siguiente fórmula:

$$P = UPL / NL \quad (1)$$

donde, UPL indica el nivel de la posición del usuario y NL el número de niveles considerados por la comunidad. Por lo tanto, si en una comunidad, por ejemplo, existen 4 niveles, entonces $NL = 4$, y si el nuevo integrante tiene un $UPL = 2$, entonces el valor de la posición (P) será 0.50. Por ejemplo, los diferentes valores de P para una persona en una organización con 4 niveles son los descritos en la tabla 1.

- Preferencias. Definir las preferencias del usuario que representará el agente es útil para poder clasificar a las personas respecto a su percepción, interpretación y procesamiento de la información [Miller & Zhichao, 2004]. Para ello, consideramos

adecuado que el agente conozca, por ejemplo, cómo la persona prefiere que se le muestre la información, es decir, en forma textual, gráfica, etc., con el fin de ofrecer un mejor servicio y tener una relación más personalizada con los usuarios.

Tabla 1. Niveles considerados para representar la posición del agente en la comunidad

Factor	Nivel	Descripción	Valor
Posición	1	Este nivel lo conforman aquellos miembros registrados en la comunidad y que no aportan un beneficio, es decir, aquellas personas que sólo se registran en la comunidad para decir que pertenece a ella y no llevan a cabo ningún tipo de actividad dentro de la misma.	0.25
	2	Nivel compuesto por personas interesadas en los temas tratados en la comunidad.	0.50
	3	Es el nivel conformado por miembros activos que colaboran y están constantemente aportando sus experiencias.	0.75
	4	Es el nivel conformado por el director, moderador y personas expertas de la comunidad.	1

Tabla 2. Notación utilizada para representar el nivel de experto

Factor	Nivel	Descripción	Valor
Nivel Experto	Ignorante	“no saber algo determinado” o “tener un conocimiento imperfecto sobre...” [ausencia del conocimiento]	0.20
	Principiante	Persona que a efectos laborales, se halla en el primer grado de una profesión manual, antes de pasar a oficial.	0.40
	Competente	Persona capacitada para hacer algo por su competencia	0.60
	Experto	Persona que demuestra ser hábil, práctico y experimentado en el desarrollo de sus tareas laborales	0.80
	Maestro	Persona que es práctica en una materia y la maneja con desenvoltura.	1.0

Por otro parte, conocer las preferencias del usuario permitirá al agente atraer la atención de la persona a la que representa en la comunidad ya que al ofrecerle información relacionada con sus gustos e intereses en el formato y contenido adecuado se puede lograr captar su atención lo cual permite un mejor aprovechamiento y reutilización de la información almacenada en un SGC. Considerar la forma de presentar la información en una comunidad de práctica es sin duda un aspecto clave para transmitir confianza y calidad en el intercambio de información entre sus miembros. Para esto, existen varias propuestas relacionadas con tests psicológicos que permiten determinar

las preferencias del usuario. Por ejemplo, en este trabajo se optó por utilizar el test de Felder-Silverman (1988) para descubrir el modo en que el usuario le gusta percibir la información (sensorial / intuitivo), el tipo de canal de entrada que prefiere (visual / verbal), la forma de procesar la información (activo / reflexivo) y de acuerdo al modo de entender (secuencial / global).

- Nivel de experto. Este término puede definirse como la habilidad o conocimiento de una persona sobre un tema específico. Este es un factor importante debido a que las personas por lo general confían más en empleados expertos que en noveles. Es importante distinguir entre posición y nivel

de experiencia, puesto que una persona que sea jefe de determinado departamento no significa que su nivel de experiencia sea alto en todos los temas que se tratan en la comunidad. Por esto se ha considerado conveniente hacer esta distinción. El nivel de experto de un empleado en una organización o en una comunidad puede calcularse a partir de su currículum vitae o considerando el tiempo que una persona ha estado trabajando en un determinado tema o actividad y ha demostrado tener cierta pericia. Varios son los valores que puede tomar este atributo, es decir, una persona de acuerdo a su nivel de experto puede ser catalogado conforme a la clasificación descrita en la Tabla 2.

Una vez definidos los niveles expertos de la comunidad, cada usuario que se registre en ella debe indicar el nivel de experto que se supone que él tiene para cada área de conocimiento considerada por la comunidad. Una vez que el valor del nivel de experto ha sido introducido por el nuevo integrante de la comunidad, el Agente Usuario (j) que representa a este nuevo integrante calculará el valor correspondiente al nivel con cada una de las áreas. Para esto, el Agente Usuario utiliza la siguiente fórmula:

$$NE_i = N/NT + VA_j \quad (2)$$

donde, NE_i es el nivel de experto asignado por el agente a su usuario en la área i , N es el nivel de experto elegido por el usuario, y NT es el número de niveles en la comunidad. El término VA_j es la variable que sirve para ajustar el nivel de experto de acuerdo con la experiencia e interacción del usuario con los demás miembros de la comunidad. Inicialmente el valor de VA_j es 0, y cada vez que otro miembro de la comunidad interactúa con un objeto de conocimiento que ha sido proporcionado por el Agente Usuario i , éste miembro debe evaluar la información consultada. Esto se hace con el fin de recopilar información que justifique la calidad de las contribuciones y capacidad del agente en ciertos temas. Cuando la valoración es positiva, entonces el nivel de experiencia del Agente Usuario se ajusta calculando VA_j como:

$$VA_j = (VN_n - VN_{n-1})/UP \text{ donde } (n \neq 1) \quad (3)$$

Si la valoración es negativa, entonces:

$$VA_j = - (VN_n - VN_{n-1})/UP \text{ donde } (n \neq 1) \quad (4)$$

donde VN_n representa un valor particular para el nivel de experiencia que se tiene. UP es un umbral de promoción que se utiliza para determinar el número de valoraciones positivas necesarias para subir de nivel de experiencia. Para ilustrar mejor esta operación centrémonos en el siguiente ejemplo:

Supóngase que en una comunidad existen cinco niveles de experto con los valores descritos en la Tabla 2. En este ejemplo, la diferencia entre los niveles es 0.20, el cual se calcula como:

$$VN_n - VN_{n-1} = 0.20$$

En este caso, se asume que se necesitan al menos 5 valoraciones para cambiar de nivel, es decir, si $UP = 5$, entonces VA_j será 0.04. Por lo tanto, 0.04 es el valor que se debe añadir cuando la valoración recibida es positiva. Con cinco valoraciones positivas ($5 * 0.04 = 0.20$) se alcanza un nuevo nivel de experto. En cambio, con cinco valoraciones negativas el valor del nivel de experto descende una posición.

Creencias. Este módulo permite representar el conocimiento o conjunto de opiniones que el agente tiene sobre su entorno. Las creencias del agente son de dos tipos: heredadas de la organización y lecciones aprendidas. Las primeras vienen dadas por la organización. Por ejemplo, organigrama organizacional, filosofía de la empresa o comunidad. Mientras que las lecciones aprendidas son aquellas adquiridas conforme el agente interactúa con el entorno lo cual le permite por ejemplo establecer parámetros para saber en qué fuentes de conocimiento confiar. Así las creencias pueden dividirse en:

- Creencias sobre Objetos de Conocimiento (OC). Representan las creencias que el agente tiene sobre los OCs de la comunidad. Cada OC contiene información que puede resultar de interés para las personas que intentan acceder a él. Por lo

tanto, los atributos del OC deben facilitar su búsqueda y localización en el repositorio o base de conocimientos para su posterior reutilización y distribución entre los miembros de la comunidad. La representación de este tipo de creencias viene dada por la siguiente terna:

$$\text{Cree}_i = \{\text{Id_OC}, \text{estado}, \text{valoración}\}$$

donde *Id_OC* corresponde al identificador del objeto de conocimiento, *estado* corresponde al nivel de satisfacción que el agente tiene sobre el objeto y *valoración* representa el valor con que el agente valora a dicho objeto.

- Creencias sobre Temas de Conocimiento (TC). Representan las creencias del agente sobre los temas de conocimiento tratados por en la comunidad. La representación de este tipo de creencias viene dada por la siguiente terna:

$$\text{Cree}_i = \{\text{Id_TC}, \text{interés}, \text{valoración}\}$$

donde *Id_TC* corresponde al identificador del tema de conocimiento, *interés* es el atributo que se valora, y *valoración* es el nivel de interés que el agente tiene en el tema.

Intereses. Un tipo particular de creencias son los intereses. A pesar de esto, en este nivel, los intereses son considerados en un módulo aparte y no dentro de las mismas creencias ya que son un factor relevante para gestionar el comportamiento del agente. Por lo tanto, este módulo nos permite conocer los intereses del agente, es decir, es el módulo encargado de almacenar los intereses del agente hacia los artefactos de la comunidad los cuales pueden ser: Objetos de Conocimiento (OC) y Temas de Conocimiento (TC).

- Intereses sobre OC. Este tipo de interés corresponde al interés mostrado por el agente sobre determinados objetos de conocimiento de una comunidad. Precisamente, el hecho de gestionar intereses relativos a los OC de la comunidad va a posibilitar:
 - Identificar los OC más consultados por los miembros de la comunidad.

- Identificar su localización y frecuencia con que se consultan y por quién son consultados.
- Intereses sobre TC. Este tipo de interés es el mostrado por el agente en temas específicos de la comunidad. Los temas tratados en una comunidad pueden ser muy variados pero siempre relacionados con un área en particular. Por ejemplo, en una comunidad de práctica de Ingeniería de Software los temas de conocimiento pueden ser: lenguajes de modelado y diseño, metodologías de desarrollo de software, métricas de desarrollo, etc.

Por otro lado, el módulo de los intereses sirve para potenciar el intercambio de conocimiento, es decir, si se conocen sus intereses en los OC y TC de la comunidad entonces conoceremos el tipo de información que se puede obtener del agente. Por lo tanto, los agentes deben ser capaces de buscar conocimiento y comprobar si existe nueva información que se ajuste a sus intereses. Esta conducta, por ejemplo, fomenta la adquisición de conocimiento de forma automática y evita que la persona a la que el agente representa tenga que buscarlo, evitando así una de las críticas de la gestión del conocimiento como es la sobrecarga de trabajo a los empleados.

Gestor de comportamientos. Este componente es fundamental para el desarrollo del nivel reactivo ya que es el módulo encargado de decidir el tipo de comportamiento del agente hacia el entorno. Este comportamiento se define en base a las creencias e intereses personales del agente, así como la información del modelo interno. Además, es el módulo responsable de encontrar una respuesta inmediata de acuerdo a las creencias del agente. Esto es, a partir de las creencias e intereses actuales del agente, éste es capaz de producir respuestas adecuadas en el instante que son requeridas.

Historial. Es el módulo del nivel reactivo capaz de almacenar las interacciones del agente con su entorno. Este componente actúa como una bitácora en donde se registran cada una de las interacciones del agente con los objetos o fuentes de conocimiento.

3.2 Nivel deliberativo-social

Es en este nivel donde el agente determina la manera de interactuar con los demás agentes que

integran la comunidad (ver Figura 5). Además, es en este nivel donde se comparte e intercambia la información entre los miembros de la comunidad y entre comunidades. En cambio en el nivel reactivo

el agente determina la manera de actuar ante situaciones que requieran una respuesta inmediata y sin ningún tipo de razonamiento.

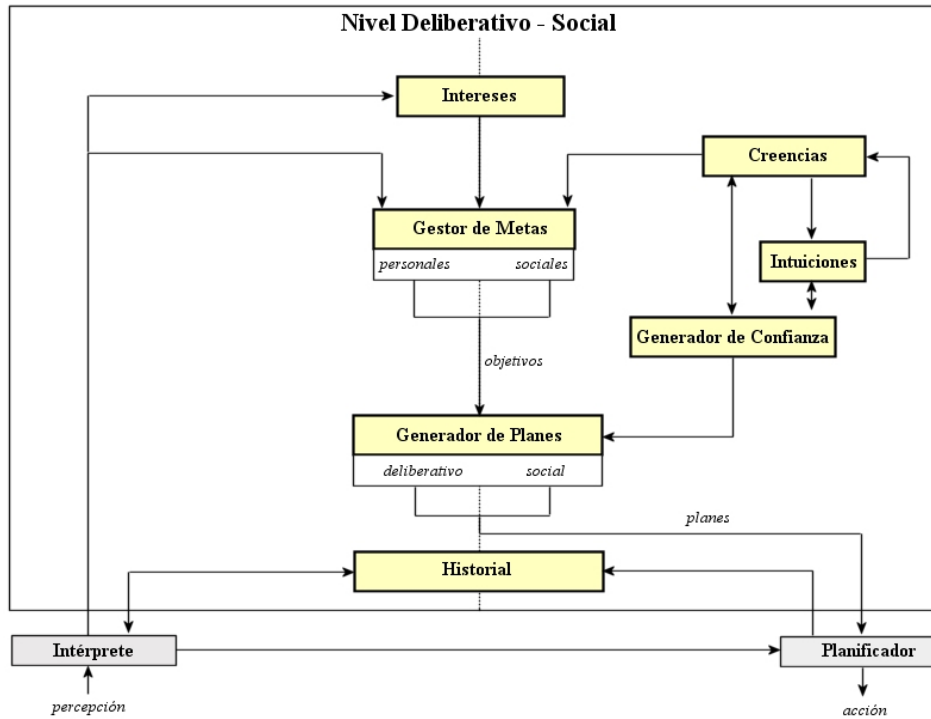


Fig. 5. Nivel deliberativo-social

Por otra parte, en el nivel deliberativo-social el agente tiene un tipo de comportamiento orientado en sus propios objetivos, esto es, le da la capacidad al agente para deliberar continuamente sobre sus objetivos actuales seleccionado un subconjunto de éstos para ser alcanzados. Para ello, el agente analiza sus propias creencias para decidir qué objetivos se podrían cumplir o qué objetivos deben activarse o desactivarse. Esta gestión de objetivos desencadena una serie de planes que a su vez pueden activar sucesivamente creencias, objetivos y planes. Es decir, primero se decide qué objetivos tienen mayor prioridad para intentar alcanzarlos. Los componentes del nivel deliberativo-social de esta arquitectura son:

Gestor de metas. Este componente define el conjunto de estados deseables en el entorno en el que se desenvuelve un agente. Un agente puede decidir alcanzar o no cada una de tales metas de acuerdo a las ventajas que ésta presente. Las

metas pueden ser tanto personales como sociales y dependerán de la situación por la que atraviese el agente. Estas metas son generadas en base a los intereses y creencias que el agente tenga de sí mismo y de la comunidad. Al tratarse de un entorno en donde debe predominar el intercambio de información y colaboración de todos los miembros de una comunidad es indispensable considerar no sólo los beneficios propios de un agente sino los de la comunidad en general. Este objetivo global puede ser dividido en sub-objetivos, los cuales son asignados como metas a los agentes del sistema, de acuerdo a alguna política organizacional.

Generador de planes. Es el módulo encargado de evaluar cómo cumplir con los objetivos planteados, y qué planes son los más indicados para alcanzarlos. El principal objetivo de este módulo consiste en generar los planes de acción

que le permitirán al agente colaborar con los miembros de la comunidad.

Creencias. En este módulo se representan el conocimiento o conjunto de opiniones que el agente tiene sobre los demás agentes. Puede comenzar como una opinión o conocimiento inicial y que ésta cambie con las acciones y desempeño de cada uno de los agentes, incluido él mismo cuando el sistema está en funcionamiento

Intereses. Este módulo representa los intereses de la comunidad, como son la identificación y localización de expertos de la comunidad. Además, es el componente encargado de mantener actualizado el conocimiento de la comunidad, de proporcionar un entorno amigable que provea a los usuarios de la confianza necesaria para compartir conocimiento, etc.

Intuiciones. Este modulo representa las creencias que no han sido comprobadas por el agente, es decir, información que puede deducirse sin antes haber tenido relación o intercambio de información con la fuente de conocimiento. Este módulo es de suma importancia para el agente ya que forma parte de uno de los factores utilizados para determinar la confianza de una fuente.

Generador de confianza. Es el módulo encargado de generar un valor de confianza para las fuentes de conocimiento con las que interactúa el agente en la comunidad. Para esto, el modulo generador de confianza considera el modelo de confianza explicado con detalle en [Soto et al, 2008] el cual considera la información obtenida del modelo interno e intuiciones del agente. Este modelo nos permite calcular el nivel de confianza mediante la siguiente fórmula:

$$C_{sj} = w_p * P_j + w_{NE} * NE_j + w_I * I_{sj} + EP_{sj} \quad (5)$$

donde C_{sj} representa la confianza que un miembro de la comunidad (s) tiene sobre otro miembro (j). P_j es el valor asignado a la posición que ocupa la persona en una organización. NE_j representa el nivel de experto el cual se calcula de acuerdo al grado de experiencia de una persona en un dominio en particular (ver fórmula 2). I_{sj} representa la intuición que el miembro de la comunidad s tiene sobre j (esto se calcula comparando los perfiles de s y j). En este caso el perfil del agente corresponde a la información personal y preferencias de la persona que representa el agente.

Por último, EP_{sj} representa la experiencia previa que el miembro de la comunidad s tiene sobre otro

miembro j . w_e , w_p y w_i son los pesos con los que se ajusta el valor de la confianza con el fin de dar mayor o menor importancia a cada uno de los factores de acuerdo a los distintos escenarios en los que se pueden ver involucrados los miembros de la comunidad. EP_{sj} constituye el componente más decisivo de la fórmula para determinar la confianza de una fuente de conocimiento. De hecho, el uso de los demás factores (nivel de experto, posición e intuición) dependen de la propia experiencia previa del agente con la fuente de conocimiento, ya que en base a este factor el agente decidirá si utiliza o no el resto de factores que componen la fórmula de acuerdo a la cantidad de experiencia previa que tenga con la fuente. La EP se calcula utilizando la Fórmula 6, la cual se explica a continuación.

$$EP_{sj} = EP_{sj}(n-1) + IA_{sj}(n) \quad (6)$$

Supongamos que un agente (s) consulta un OC que ha sido almacenado por otro agente (j) de la comunidad, en este caso, el agente (s) debe evaluar que tan útil le ha resultado el OC del agente (j)

De esta manera podemos obtener un valor que representa la interacción entre ambos agentes (uno consultando y el otro facilitando información). Este valor se representa en la fórmula como IA_{sj} (Valor de la Interacción Actual que el agente s tiene sobre el agente j). Consecuentemente, el valor de la $EP_{sj}(n-1)$ se obtiene calculando el promedio de los valores asignados a cada una de las contribuciones de j que el agente s ha evaluado en la interacción $n-1$.

En la fórmula 6, EP_{sj} representa la valoración general del agente s hacia el agente j .

Historial. Este módulo se encarga de llevar un registro cronológico de sucesos, en donde especifiquen las interacciones del agente con las demás fuentes de conocimiento de la comunidad.

Una vez descritos los módulos que conforman el nivel reactivo y deliberativo-social del modelo cognitivo de la arquitectura, en la siguiente sección se describe el prototipo desarrollado para probar la funcionalidad de la arquitectura propuesta.

4 Prototipo

El prototipo desarrollado provee la opción de almacenar/consultar documentos de una CoP y actualizar los valores de confianza sobre los

agentes que proveen esos documentos. Por otra parte, el prototipo permite la creación de comunidades donde el creador pueda establecer los temas a tratar. Además, los usuarios que lo deseen podrán buscar comunidades que se hayan creado y registrarse en aquellas cuyos temas tratados sean de interés para dicho usuario. Respecto a los agentes software a utilizar, por cada usuario existe un Agente Usuario que gestiona la información del mismo y que sea capaz

de realizar recomendaciones de OC al mismo. Por ejemplo, si un usuario está interesado en un tema de conocimiento de la comunidad y desea buscar documentos relacionados con ese tema, entonces su Agente Usuario deberá contactar a otros Agentes Usuarios de la comunidad que han propuesto documentos relacionados con el tema en cuestión.

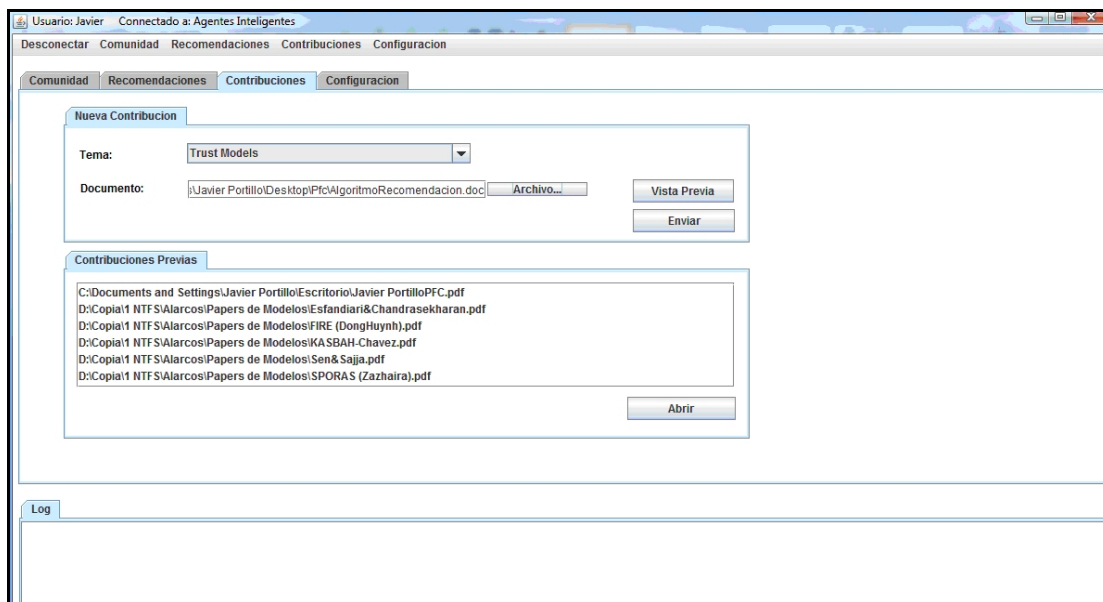


Fig. 6. Proponer un nuevo documento

Una vez que un usuario se conecta a una comunidad dispondrá de una ventana para acceder a los servicios que su agente ofrece para esa comunidad como, por ejemplo, el envío de OC a la comunidad o las recomendaciones de OC. Estos dos tipos de situaciones se describen a continuación.

Proponiendo un nuevo OC (Figura 6). Para proponer un OC, una persona debe acceder al apartado "contribuciones" de la herramienta y configurar las siguientes opciones :

- Tema. En cada comunidad hay diferentes áreas o temas de conocimiento. Los usuarios deben seleccionar el área correspondiente al OC que proponen.
- Documento. Corresponde al OC propuesto.

1.-Uso de los OC de la comunidad (Figura 7). Los miembros de la comunidad pueden solicitar una recomendación sobre los OCs relacionados con un tema en particular. Para esto, el Agente Usuario recibe la petición y contacta al Agente Gestor que es el encargado del mantenimiento de la comunidad. El Agente Gestor responde al Agente Usuario con una lista de OCs de la comunidad relacionados con la consulta. Después el Agente Usuario ordena esta lista considerando las siguientes situaciones:

- Aquellos OCs que han sido evaluados previamente. Un OC consultado por los miembros de la comunidad tendrá una serie de valoraciones (junto con la identidad de cada uno de los usuarios

que lo evaluaron). Ante esta situación el Agente Usuario debe determinar el valor de confianza de cada documento, considerando la totalidad de las valoraciones del OC y el nivel de confianza del agente hacia la persona que hizo la evaluación, para ello, utilizará las fórmulas 5 y 6 explicadas en la sección anterior. Además de esto, el Agente Usuario también debe considerar la confianza que tiene en la persona que proporcionó el OC a la comunidad. Por lo tanto, el agente ordenará los OCs según la opinión del Agente Usuario acerca del proveedor del OC

- y la información de los demás miembros que evaluaron dicho OC.
- OCs sin evaluar. Un OC puede carecer de evaluación. En este caso el Agente Usuario consulta a otros agentes (los cuales son considerados sus 'amigos', en otras palabras, los agentes más fiables para este agente) acerca del nivel de confianza que ellos tienen en la persona que proporcionó el OC a la comunidad. La opinión personal del Agente Usuario en relación al proveedor es, por supuesto, también utilizada para calcular el valor de confianza.

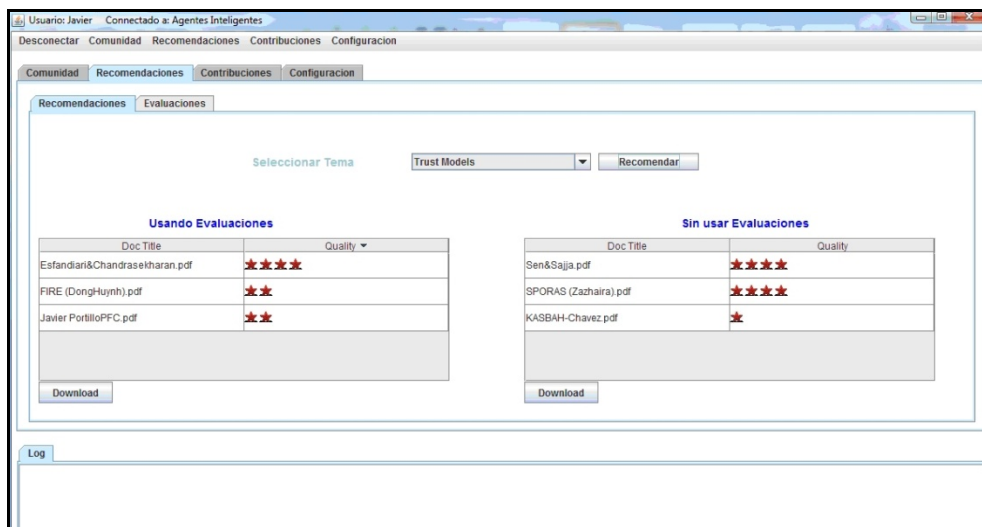


Fig. 7. Listado de documentos recomendados

Una vez que los OCs han sido ordenados por el Agente Usuario éste le mostrará los más relevantes al usuario como muestra la Figura 7 donde los OCs se dividen en dos tablas, aquellos que tienen evaluación y los que no. El usuario normalmente seleccionará primero aquellos OCs contenidos en la tabla de la izquierda ya que al tener retroalimentación sobre ellos es más probable que la recomendación sea más adecuada. En caso de que no existan OCs en esta tabla, el usuario podría consultar los OCs contenidos en la otra tabla (OCs sin evaluación).

El nivel de confianza se representa por el número de estrellas. Por ejemplo, cinco estrellas

indican el mayor nivel de confianza que un agente puede tener y una estrella indica el nivel de confianza más bajo.

Una vez que los OCs han sido seleccionados, el usuario debe evaluar los OCs consultados con el fin de proporcionar feedback a la comunidad respecto a ellos. Esta acción es muy importante ya que estas valoraciones son utilizadas por los agentes para recomendar OCs.

Detalles de la implementación

La plataforma de desarrollo utilizada es JADE (Java Agent Development Framework) debido a que cumple con el estándar FIPA y constituye una

de las plataformas más utilizadas en el desarrollo de agentes. Además JADE ha sido utilizada con gran éxito en el desarrollo de sistemas relacionados con la gestión del conocimiento.

5 Evaluación

Actualmente se están realizando distintas evaluaciones del prototipo, dichas evaluaciones se pueden enfocar tanto desde el punto de vista de la tecnología (evaluar la eficiencia de la arquitectura multi-agente) como desde un punto social (ver el impacto del sistema en la comunidad). El primer tipo de validación, comienza por comprobar cuál sería el número de agentes más adecuado para el sistema y para una comunidad. Para esto, hemos preparado tres grupos de mensajes: el primer grupo, contiene un total de 300 mensajes; el segundo grupo, está formado por 800 mensajes y por último el tercer grupo que contiene 1000 mensajes. En estos grupos, todos los mensajes son peticiones de información de los Agentes Usuarios miembros de la Comunidad. Considerando esto, los resultados obtenidos del experimento fueron los siguientes:

- Cuando se enviaron todos los mensajes del primer grupo (al mismo tiempo), éstos fueron procesados correctamente.
- Cuando se enviaron los mensajes del segundo grupo, 13 de los 800 no se procesaron correctamente, es decir, 13 de los mensajes no obtuvieron respuesta.

- Cuando se envió el tercer grupo, 80 de los 1000 mensajes no se procesaron correctamente.

Estos resultados indican que el sistema empieza a ser menos eficiente cuando hay más de 800 usuarios haciendo peticiones al mismo tiempo. Sin embargo, como nuestro enfoque es para pequeñas y medianas empresas (PyMES) es de esperarse que el número de usuarios sea menor que 800, por lo tanto el rendimiento del sistema debería ser el óptimo.

Además de usar los experimentos para detectar si el intercambio de mensajes es adecuado, estamos evaluando la fórmula de confianza que proponemos. Para esto, hemos implementado una aplicación que permite configurar un grupo de agentes que simulen un grupo de personas. En esta simulación hacemos uso de las fórmulas de confianza propuestas para determinar el nivel de confianza de un agente, y como este nivel cambia conforme el tiempo.

Para esto, hemos configurado la simulación de tal forma que exista un agente fiable (que llamaremos *Agente Truster*). Este agente es fundamental para nuestro experimento ya que intentaremos probar que los demás agentes de la comunidad detecten que es un agente en el que pueden confiar.

La idea de esta simulación es descubrir si es mejor utilizar todos los factores propuestos en la fórmula de confianza o sólo utilizar el factor correspondiente a la experiencia previa, tal y como se considera en otros trabajos.

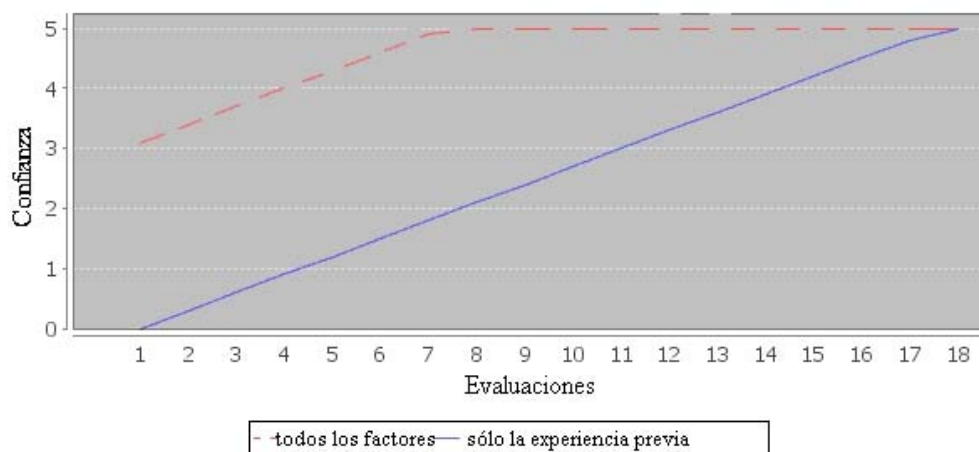


Fig. 8. Resultado de la simulación

Antes de iniciar la simulación hemos asignado al *Agente Truster* el valor de confianza 5. Este valor representa el nivel de confianza más alto que un agente puede tener.

La Figura 8 ilustra el nivel de confianza del *Agente Truster* respecto a las evaluaciones de los demás agentes de la comunidad. Cuando sólo se utiliza el factor correspondiente a la experiencia previa, el valor inicial de la confianza para el *Agente Truster* es cero, esto implica que los agentes no saben qué tan confiable es este agente. Después de 18 evaluaciones este valor alcanza el nivel de confianza 5, que corresponde al nivel asignado en un principio al *Agente Truster*.

Por otro lado, si utilizamos todos los factores de la fórmula de confianza, la cantidad de evaluaciones necesaria para alcanzar el nivel de confianza 5 es de 7.

Por lo tanto, podemos concluir que si se consideran todos los factores propuestos es más fácil y rápido identificar fuentes de conocimiento fiables que utilizando sólo la experiencia previa ya que los agentes obtienen un valor de confianza inicial y no necesitan de mucha experiencia previa para obtener un valor de confianza real.

Una vez que se termine con la validación tecnológica se comenzará la evaluación social donde se pretenderá contestar a una importante cuestión: ¿Cómo el uso del sistema recomendador puede afectar el rendimiento de una comunidad?. Como se detalla en [Geib et al, 2004] medir el rendimiento de comunidades es un reto importante puesto que las comunidades sólo tienen un impacto indirecto en los resultados de la empresa. Para realizar este tipo de validación se tendrán en cuenta ideas del marco de trabajo propuesto por [McDermott, 2002] para evaluar comunidades. En dicho marco se mide el rendimiento en función de los siguientes parámetros: ganancia de conocimiento personal, incremento de relaciones y acceso a la información. Se espera que utilizando el concepto de confianza para detectar las fuentes y objetos de conocimiento más confiables de una organización y gracias al uso de agentes que recomiendan el conocimiento más adecuado para sus usuarios, la eficiencia de la comunidad mejore. Sin embargo, este hecho todavía debe ser contrastado y será uno de nuestros trabajos futuros.

6 Trabajo relacionado

Nuestro trabajo puede ser comparado con otras propuestas que utilizan agentes y modelos de confianza durante el intercambio de conocimiento. Zacharia et al (1999) proponen Sporas, un mecanismo de reputación mejorado de los sistemas de reputación online para modelar dinámicamente la confianza de los usuarios en el comercio electrónico. En esta propuesta, sólo se considera la evaluación de la interacción más reciente. El problema con este modelo es que el sistema requiere de una gran cantidad de interacciones y cuando un usuario tiene un valor de reputación alto es difícil que este valor pueda cambiar. Otro enfoque de este trabajo es el de Histos, que es un modelo más personalizado al de Sporas. Histos también se enfoca en la información relativa a las evaluaciones que un agente hace de otro al interactuar, considerando sólo las experiencias más recientes. Al contrario que en Sporas, el valor de reputación es una propiedad subjetiva asignada por cada agente. Una desventaja de este modelo es utilizar el valor de reputación asignado a un tercero como medida de su confiabilidad. Si una gente es buen colaborador no significa que sea buen proveedor de información actuando como tercero. Sabater y Sierra (2001) presentan un modelo de reputación llamado Regret en el cual los valores de reputación dependen del tiempo, es decir, las evaluaciones más recientes son más importantes que las anteriores. Carbó et al (2003) presentan Afras, un modelo basado en Sporas cuya característica central es el uso de variables difusas para representar la reputación, mientras que la toma de decisiones está inspirada en un enfoque cognitivo. Carter et al (2002) presentan un modelo que considera la reputación y confianza como propiedades emergentes de a interacción directa entre agentes, basadas en múltiples interacciones entre las dos partes. En este modelo la confianza se presenta como la creencia de un agente respecto al rendimiento de la otra parte al momento de realizar cierta tarea, de acuerdo a su propio conocimiento. Wang y Vassileva (2003) proponen un mecanismo de reputación y confianza en entornos P2P (*peer-to-peer*) que permite a los agentes identificar personas capaces de resolver sus necesidades a través de la experiencia directa y la compartición de sus experiencias con otras

personas con preferencias similares. Por otro lado, Sierra y Debenham (2005) proponen un modelo de decisión para agentes inteligentes involucrados en cuestiones de múltiples negociaciones utilizado para seleccionar socios con los cuales se puede colaborar o seleccionar estrategias para discutir con el socio elegido. El modelo se fundamenta en la teoría de la información, el cual considera aspectos de confianza para idear mecanismos que permitan manejar diálogos. Este modelo permite el diseño de agentes que puedan tomar 'decisiones' en base a la información observada. Otro trabajo interesante es el de Barber y Kim (2001), en su modelo el agente es capaz de evaluar la información de entrada para generar una base de conocimiento consistente, y evitar información fraudulenta por parte de las fuentes o agentes de poca confianza o engañosos. El objetivo de este trabajo es similar al nuestro. Sin embargo, los medios para alcanzarlo son diferentes. Los autores definen confianza como "el nivel de seguridad que se tiene sobre la capacidad de una fuente para proveer información fiable". En este modelo cada vez que una fuente envía conocimiento, esta fuente debe indicar el factor de certeza que tiene sobre el conocimiento. En nuestro caso, el enfoque es diferente ya que la fuente que recibe la información es quien se encarga de evaluar el conocimiento. La principal diferencia entre estos modelos y nuestro enfoque es que los modelos descritos en este apartado requieren de un mínimo de interacciones para poder determinar un valor de reputación aceptable. Por lo tanto, no son útiles para calcular valores de confianza para usuarios recién llegados a una comunidad. En la literatura existe una variedad de propuestas para calcular la reputación y confianza que utilizan el paradigma de agentes inteligentes. A continuación se describe

una tabla de los modelos siguiendo la clasificación de Sabater (2003).

7 Conclusiones y trabajo futuro

Las CoPs son un medio para compartir conocimiento. Sin embargo, si el conocimiento no es reutilizado, el conocimiento pierde su valor y se vuelve obsoleto. Por otro lado, los miembros de una comunidad suelen ignorar el conocimiento generado por la propia comunidad. Con el fin de asegurar el reuso del conocimiento en una CoP, en este trabajo proponemos una arquitectura multi-agente en la cual los agentes representan a miembros de una CoP que utilizan un modelo de confianza para evaluar la fiabilidad de las fuentes de conocimiento con las que interactúan.

Una ventaja de este trabajo es que la arquitectura ha sido diseñada para dar apoyo a las CoPs, y considera factores tales como el nivel de experto, posición, intuición y experiencia previa, que suelen influir en las personas al momento de confiar en fuentes de conocimiento. Otro aspecto a destacar es el modelo de confianza utilizado en la arquitectura, que a diferencia de otros modelos, permite calcular la confianza de una fuente cuando no existe experiencia previa con la fuente ya que considera otros factores como la intuición, posición y nivel de experto. Esta es una ventaja importante sobre otros trabajos que requieren de mucha experiencia previa para poder dar un valor de confianza.

Tabla 3. Modelos de reputación y confianza basados en el paradigma de agentes

Modelo	Dominio	Tipo modelo	Fuentes de información	Tipo visibilidad	Suposición del comportamiento	Tipo de información intercambiada
[Zacharia et al, 1999] [Wang y Vassileva, 2003]	Comercio electrónico P2P	R C, R	IT, ID IT, ID	G Pr	N1 N1	Entre 0 y 1 0 y 1
[Sabater & Sierra, 2001] [Carbó et al, 2003]	Comercio electrónico Comercio electrónico	C,R R	IT, ID, IS, P IT, ID	Pr Pr	N3 N3	Entre -1 y 1 Valores difusos
[Caballero et al, 2006]	P2P	C, R	IT, ID, P	Pr	N1	[0, 1]
[Sierra & Debenham, 2005]	Genérico	C	ID, OD	Pr	N2	NE
[Carter et al, 2002]	Genérico	R	IT	G	N1	Entre 0 y 1
<p><i>Notación:</i> C: Confianza R: Reputación IT: Información de terceros ID: Interacción directa G: Global Pr: Personalizada P: Prejuicio OD: Observación directa N1: Los agentes no mienten ni ocultan información N2: Los agentes no mienten pero pueden ocultar información N3: Los agentes pueden mentir NE: No explicado</p>						

En nuestro caso el modelo es capaz de calcular un valor de confianza preliminar para la fuente de conocimiento sin utilizar el factor correspondiente a la experiencia previa. Otra ventaja de la arquitectura es que considera aspectos sociales los cuales son implementados en el nivel deliberativo-social del modelo cognitivo del agente. De hecho, la característica principal de nuestra arquitectura se centra en la capa deliberativa-social, que surgió tras el diseño de una versión anterior de la arquitectura con tres niveles (reactivo, deliberativo y social). Sin embargo, nos dimos cuenta que en nuestro dominio de aplicación, todos los objetivos deliberativos tienen de manera implícita un comportamiento social. Como resultado de esto, los niveles deliberativo y social son tratados conjuntamente en un solo nivel, siendo esta una de las principales diferencias de nuestra arquitectura con otras. Otra contribución importante es que en la arquitectura se modela el

concepto de intuición, que debido a que es un factor subjetivo, por lo general, no es considerado por los SMA.

Como trabajo futuro, estamos trabajando en añadir nuevas funcionalidades al prototipo, tales como la identificación de experto y detección de usuarios que intentan engañar el sistema con información irrelevante para los miembros de la comunidad.

Agradecimientos

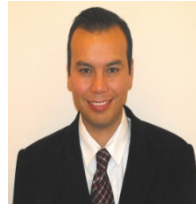
Este trabajo ha sido parcialmente financiado por el proyecto FABRUM (PPT-430000-2008-063), MELISA (PAC08-0142-3315), ENGLOBAS (PII2109-0147-82-35), Junta de Comunidades de Castilla-La Mancha, Ministerio de Educación y Ciencia (Dirección General de Investigación)/Fondos Europeos de Desarrollo Regional (FEDER) en España y al Consejo

Nacional de Ciencia y Tecnología (CONACYT) en México.

Referencias

1. **Atkinson-Abutridy, J. & Ferreira-Cabrera, A. (1998).** Un enfoque integrado para disminuir la sobrecarga en la búsqueda de información utilizando un agente adaptativo guiado por interacciones dialógicas en lenguaje natural. *Computación y Sistemas*, 2(1), 14-23.
2. **Barber, K. & Kim, J. (2001).** Belief revision process based on trust: simulation experiments. *Autonomous Agents '01 Workshop on Deception, Fraud and Trust in Agent Societies*. Montreal, Canada, 1-12.
3. **Caballero, A., Botia, J. & Skarmeta, A. (2006).** A new model for trust and reputation management with an ontology based approach for similarity between tasks, *Multiagent System Technologies. Lecture Notes in Computer Science*, 4196, 172-183.
4. **Carbó, J., Molina, M. & Dávila, J. (2003).** Trust management through fuzzy reputation. *International Journal of Cooperative Information Systems*, 12(1), 135-155.
5. **Carter, J., Bitting, E., & Ghorbani, A. (2002).** Reputation formalization for an information-sharing multi-agent system. *Computational Intelligence*, 18(4), 515-534.
6. **Chiu, C.-M., Hsu, M.-H. & Wang, E. (2006).** Understanding knowledge sharing in virtual communities: an integration of social capital and social cognitive theories. *Decision Support Systems*, 42(3), 1872-1888.
7. **Davenport, E. (2001).** Knowledge management issues for online organisations: 'communities of practice' as an exploratory framework. *Journal of Documentation*, 57(1), 61-75.
8. **De Long, D. & Fahey, L. (2000).** Diagnosing cultural barriers to knowledge management. *Academy of Management Executive*, 14(4), 113-127.
9. **Dillenbourg, P. (1999).** What do you mean by 'collaborative learning'?. In P. Dillenbourg (Ed.), *Collaborative-Learning: Cognitive and Computational Approaches* (1-19). United Kingdom: Emerald Group Publishing Limit.
10. **du Plessis, M. (2008).** The strategic drivers and objectives of communities of practice as vehicles for knowledge management in small and medium enterprises. *International Journal of Information Management*, 28(1), 61-67.
11. **Felder, R. & Silverman, L. (1988).** Learning and Teaching Styles in Engineering Education. *Engineering Education*, 78(7), 674-681.
12. **Ferguson, I. (1992).** *TouringMachines: an architecture for dynamic, rational, mobile agents*. Phd Thesis, University of Cambridge, Cambridge, United Kingdom.
13. **Fuentes, R., Gómez-Sanz, J. & Pavón, J. (2004).** A sociological framework for multi-agent systems validation and verification. *Conceptual Modeling, Lecture Notes in Computer Science*, 3289, 458-469.
14. **Gebert, H., Geib, M., Kolbe, L. & Brenner, W. (2003).** Knowledge-enabled customer relationship management: integrating customer relationship management and knowledge management concepts. *Journal of Knowledge Management*, 7(5), 107-123.
15. **Geib, M., Braun, C., Kolbe, L. & Brenner, W. (2004).** Measuring the utilization of collaboration technology for knowledge development and exchange in virtual communities. *37th Annual Hawaii International Conference on System Sciences (HICSS'04)*, Big Island, Hawaii, Vol. 1, 1-10.
16. **Hahn, J. & Subramani, M. (2000).** A framework of knowledge management systems: issues and challenges for theory and practice. *Twenty First International Conference on Information Systems*, Atlanta, USA, 302-312.
17. **He, W., Qiao, Q. & Wei, K.-K. (2009).** Social relationship and its role in knowledge management systems usage. *Information & Management*, 46 (3), 175-180.
18. **Hillery, G. (1955).** Definitions of community: areas of agreement. *Rural Sociology*, 20(2), 111-123.
19. **Hinds, P. & McGrath, C. (2006).** Structures that work: social structure, work structure and coordination ease in geographically distributed teams. *20th Anniversary Conference on Computer Supported Cooperative Work*. Banff, Alberta, Canada, 345-352.
20. **Imbert, R. (2005).** *Una arquitectura cognitiva multinivel para agentes con comportamiento influido por características individuales y emociones propias y de otros agentes*. Tesis de doctorado, Universidad Politécnica de Madrid, Madrid, España.
21. **Koh, J. & Kim, Y.-G. (2004).** Knowledge sharing in virtual communities: an e-business perspective. *Expert Systems with Applications*, 26(2), 155-166.
22. **Maes, P. (1994).** Agents that reduce work and information overload. *Communications of the ACM*, 37(7), 31-40.
23. **McDermott, R. (2002).** Measuring the impact of communities. *Knowledge Management Review*, 5(2), 26-29.
24. **Miller, J. & Zhichao, Y. (2004).** A cognitive-based mechanism for constructing software inspection teams. *IEEE Transactions on Software Engineering*, 30(11), 811-825.
25. **Ridings, C., Gefen, D. & Arinze B. (2002).** Some antecedents and effects of trust in virtual communities. *The Journal of Strategic Information Systems*, 11(3-4), 271-295.
26. **Sabater, J. & Sierra, C. (2001).** Regret: a reputation model for gregarious societies. *Fourth Workshop on Deception, Fraud and Trust in Agent Societies*. Montreal Canada, 61-69.
27. **Sabater, J. (2003).** *Trust and reputation for agent societies*. PhD Thesis, Universidad Autónoma de Barcelona, Barcelona Spain.
28. **Sánchez, J., López, C. & Schnase, J. (1998).** Chrysalis: agentes de usuario en la construcción de bibliotecas digitales botánicas. *Computación y Sistemas*, 2(2-3), 95-103.
29. **Sher, P. & Lee, V. (2004).** Information Technology as a facilitator for enhancing dynamic capabilities through knowledge management. *Information & Management*, 41(8), 933-945.

30. **Sierra, C. & Debenham, J. (2005).** An information-based model for trust. *4th International Joint Conference on Autonomous Agents & Multi Agent Systems (AAMAS'05)*, Utrecht, Netherlands, 497-504.
31. **Soto, J.P., Vizcaíno, A., Portillo-Rodríguez, J. & Piattini, M. (2007).** A three level multi-agent architecture to foster knowledge exchange. *19th International Conference on Software Engineering and Knowledge Engineering (SEKE)*, Boston, USA, 565-569.
32. **Soto, J.P., Vizcaíno, A., Portillo-Rodríguez, J., Rodríguez-Elias, O-M. & Piattini, M. (2008).** A prototype to recommend trustworthy knowledge in communities of practice. *Third International Conference on Software and Data Technologies (ICSOFT 2008)*, Vol. PL/DPS/KE, Porto Portugal, 321-326.
33. **Ushida, H., Hirayama, Y. & Nakajima, H. (1998).** Emotion model for life like agent and its evaluation. *AAAI'98/IAAI'98 Fifteenth National Conference on Artificial Intelligence and Tenth Innovative Applications of Artificial Intelligence Conference*. Madison, Wisconsin, USA, 62-69.
34. **Vizcaíno, A., Soto, J.P., Portillo, J. & Piattini, M. (2007).** A multi-agent model to develop knowledge management systems. *40th Annual Hawaii International Conference on Systems Sciences (HICSS)*. Big Island, Hawaii, 203b.
35. **Wang, Y. & Vassileva, J. (2003).** Trust and reputation model in peer-to-peer networks. *3rd International Conference on Peer-to-Peer Computing*, Linköping, Sweden, 150-157.
36. **Wasserman, S. & Galaskiewicz, J. (1994).** *Advances in social networks analysis: Research in the Social and Behavioral Sciences*. Sage Publications. Thousand Oaks, California: Sage Publications Inc.
37. **Wasko, M. & Faraj, S. (2000).** "It is what one does": why people participate and help others in electronic communities of practice. *The Journal of Strategic Information Systems*, 9(2-3), 155-173.
38. **Wenger, E. (2000).** Communities of practice and social learning systems. *Organization*, 7(2), 225-246.
39. **Wooldridge, M. & Ciancarini, P. (2001).** Agent-oriented software engineering: The state of the art. *First international workshop, AOSE 2000 on Agent-Oriented Software Engineering*, Secaucus, NJ, USA, 1-28.
40. **Zacharia, G., Moukas, A. & Maes, P. (1999).** Collaborative reputation mechanisms in electronic marketplaces. *32nd Annual Hawaii International Conference on System Science (HICSS'99)*. Maui, Hawaii. Vol. 8, 8026.



Juan Pablo Soto Barrera

Doctor Ingeniero en Informática y Máster en Tecnologías Informáticas Avanzadas por la Universidad de Castilla – La Mancha (UCLM). Licenciado en Ciencias Computacionales por la Universidad Autónoma de Baja California. Es miembro del grupo de investigación Alarcos en la Escuela Superior de Informática de Ciudad Real (España). Ha trabajado como consultor para varias empresas privadas. Actualmente se desempeña como Profesor- Investigador en el departamento de Matemáticas de la Universidad de Sonora. Sus líneas de investigación son: Ingeniería del Software, Gestión del Conocimiento y Agentes Inteligentes.



Aurora Vizcaíno Barceló

Doctora por la UCLM y Licenciada en Informática por la Universidad de Granada. Es miembro del grupo de investigación Alarcos en la Escuela Superior de Informática (ESI) de Ciudad Real (España) donde trabaja como profesora titular de universidad. Sus líneas de investigación son: Desarrollo Global del Software, Gestión del Conocimientos y Agentes Inteligentes.



Javier Portillo Rodríguez

Alumno de Doctorado en la UCLM. Máster en Tecnologías Informáticas Avanzadas e Ingeniero en Informática por la UCLM. Es miembro del grupo de investigación Alarcos en la Escuela Superior de Informática de Ciudad Real (España). Sus líneas de investigación son: Desarrollo Global del Software y Agentes Inteligentes.



Mario Piattini Velthuis

Doctor y Licenciado en Informática por la Universidad Politécnica de Madrid. Licenciado en Psicología por la Universidad Nacional de Educación a Distancia. Máster en Auditoría Informática (CENEI). Especialista en la Aplicación de Tecnologías de la Información en la Gestión Empresarial (CEPADE-UPM). CISA (Certified Information System Auditor), CISM (Certified Information System Manager) y CGEIT (Certified in the Governance of Enterprise IT) por la ISACA. Diplomado en Calidad por la Asociación Española para la Calidad. Ha trabajado como consultor para numerosos organismos y empresas. Catedrático de Universidad de Lenguajes y Sistemas Informáticos en la ESI de la UCLM, donde dirige el grupo de investigación Alarcos, especializado en Calidad de Sistemas de Información.