

Modelo de Servicio de Razonamiento con Preferencias

Juan Carlos Teze¹, Sebastian Gottifredi²,
Alejandro Javier García³ y Guillermo Ricardo Simari⁴

Laboratorio de Investigación y Desarrollo en Inteligencia Artificial

^{2,3,4} Departamento de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur,
Av. Alem 1253, (B8000CPB) Bahía Blanca, Argentina

Laboratorio de Investigación y Desarrollo en Inteligencia Artificial Concordia

¹Facultad de Ciencias de la Administración, Universidad Nacional de Entre Ríos,

^{1,2,3} Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

e-mail: {jct,sg,ajg,grs}@cs.uns.edu.ar

Resumen

En la última década formalizar modelos para que agentes deliberativos puedan razonar en un sistema multi-agente se ha convertido en un importante área en AI. Muchos enfoques se han propuesto, uno de estos es el de Servicio de Razonamiento [15]; esta línea de investigación básicamente se focalizará en dicho servicio. Actualmente este servicio cuenta con un conjunto de operadores que modifican temporalmente la información almacenada en el servidor y un mecanismo de inferencia que es el que resuelve una consulta dada. No obstante, una instanciación de este servicio es el Servicio de Razonamiento basado en argumentación rebatible. En particular, en este servicio el mecanismo de inferencia está configurado para que utilice siempre los mismos criterios de preferencia al momento de resolver consultas cliente. El objetivo general de esta línea de investigación es intentar modelar un Servicio de Razonamiento con preferencias donde las mismas puedan manipularse de forma dinámica. Como resultado se buscará definir algunos operadores para lograr tales objetivos.

Palabras clave: Sistema Multi-agente, Agente Inteligente Deliberativo, Servicio de Razo-

namiento, Preferencias.

Contexto

Esta línea de investigación se llevará a cabo dentro del ámbito de colaboración entre el Laboratorio de Investigación y Desarrollo (LIDIA) del Dep. de Ciencias e Ingeniería de la Computación, Universidad Nacional del Sur; y el Laboratorio de Investigación y Desarrollo en Inteligencia Artificial Concordia (LIDIA Concordia) de la Facultad de Ciencias de la Administración, Universidad Nacional de Entre Ríos. Está asociada a los siguientes proyectos de investigación:

- Desarrollo de Sistemas de Argumentación Masiva sobre Base de Datos Federadas. PID-UNER 7041.
- Representación de conocimiento y Razonamiento Argumentativo: Herramientas Inteligentes para la Web y las Bases de Datos Federadas. PGI 24/N030, financiados por la Universidad Nacional del Sur.

Además, este trabajo se realiza en el marco del desarrollo de una tesis doctoral para optar por el título de Doctor en Ciencias de la Computación del Departamento de Ciencias de la Computación de la Universidad Nacional del Sur.

Introducción

Una característica esencial de los Sistemas Multi-Agente (SMAs) es la interacción entre agentes. Los agentes pertenecientes a un SMA interactúan con el objetivo de llevar a cabo tareas, ya sea individuales o colectivas. Los agentes deliberativos generalmente razonan utilizando dos tipos de conocimiento: uno público que es el que comparten con otros agentes y otro privado que surge en parte de su percepción del mundo.

En [15] se propone un modelo cliente-servidor que permite a los agentes compartir conocimiento con otros agentes y representar el conocimiento privado de cada uno. En este modelo se define el concepto de Servicio de Razonamiento. Este servicio permite representar información o conocimiento público y responder consultas de clientes utilizando dicho conocimiento público junto con información privada provista por el cliente. Básicamente un Servicio de Razonamiento está constituido por:

1. el conocimiento público que será utilizado como base para la resolución de las consultas de los clientes;
2. un mecanismo de inferencia que utilizará ese conocimiento para computar las respuestas a consultas de los clientes; y
3. un conjunto de operadores de tratamiento de contexto mediante los cuales se podrá, por ejemplo, agregar en forma temporal información para crear el contexto en el cual se procesará la consulta del cliente.

Aquellos contextos en donde la información que se maneja es incompleta o potencialmente contradictoria brindan un escenario ideal para sistemas argumentativos [10, 1, 3, 4] capaces de representar el conocimiento y definir el razonamiento de agentes inteligentes. El mecanismo de inferencia sobre el cual están basado, permite decidir entre conclusiones contradictorias y adaptarse fácilmente a entornos cambiantes.

En argumentación dos argumentos pueden estar en conflicto. Para determinar que argumento prevalece en una situación de conflicto se utilizan criterios de preferencia. El argumento que prevalece se dice que es un derrotador del otro argumento en conflicto. En la literatura existen numerosos criterios de preferencia [8, 2, 12] para comparar argumentos. En particular, en los Servicios de Razonamiento basado en argumentación rebatible, los criterios de preferencia se encontrarían embebido en el Mecanismo de Inferencia.

El modelo actual de Servicio de Razonamiento, puede ser utilizado en aplicaciones particulares en dominios de aplicación concretos con diferentes lenguajes de representación de conocimiento. Concretamente en [9], se propone un servidor basado en programación en lógica rebatible. Es importante mencionar que estos servicios están estrechamente relacionados con sistemas recomendadores [6, 7, 14, 11, 16]; esto se debe a que los sistemas recomendadores usualmente operan creando un modelo de las preferencias de los usuarios con el objetivo de poder anticiparse a las necesidades y deseos del usuario. Un Servicio de Razonamiento puede ser implementado como un servidor de recomendación.

En base a lo expuesto arriba, un paso importante para avanzar es el de formalizar un modelo de Servicio de Razonamiento basado en argumentación rebatible [5, 13] en donde

se puedan manejar diversas preferencias por parte del servidor y el tratamiento que puedan recibir sea dinámico. En la actualidad el Servicio de Razonamiento no cuenta con mecanismos para manejar preferencias. El uso de preferencias y operadores concretos aplicados a estas es un avance fundamental en estos tipos de servicios.

A continuación, se explica cómo esta línea de investigación integra la noción de preferencia a un Servicio de Razonamiento. Se repasarán los resultados obtenidos y el trabajo en progreso.

Líneas de investigación y desarrollo

La línea principal de investigación de este artículo toma como base el concepto de Servicio de Razonamiento y busca avanzar sobre un tipo de servicio en particular; Servicio de Razonamiento basado en argumentación rebatible. Un primer avance va ser intentar extender un Servicio de Razonamiento para que pueda soportar varios criterios de preferencia sobre la información a computar. Además, se espera poder construir consultas específicas dirigidas a estos servicios en concreto.

Por lo tanto, nuestra línea de investigación está motivada por tres ejes: el primero se basa en adaptar los servicios mencionados en la introducción para que los criterios de preferencias sean un modulo independiente del Mecanismo de Inferencia; el segundo en definir consultas de clientes dirigidas a estos servicios en particular; y el tercero en definir nuevos operadores que permitan la manipulación de dichas preferencias.

Servicio de razonamiento con preferencias

Un Servicio de Razonamiento responde una consulta recibida de un clien-

te. Para responder consultas, estos tienen almacenado conocimiento público el cual es utilizado junto al conocimiento privado del agente para crear un contexto adecuado a la consulta. Un Servicio de Razonamiento basado en argumentación rebatible podría contar de cuatro componentes: el conocimiento público, un mecanismo de inferencia, un conjunto de operadores de tratamiento de contexto y un conjunto de criterios de preferencia.

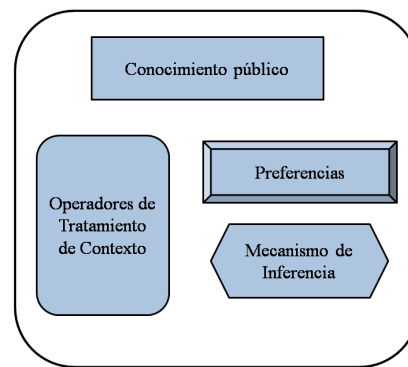


Figura 1: Servicio de razonamiento basado en argumentación rebatible.

Los Servicios de Razonamiento responden a un modelo cliente-servidor por consiguiente buscan resolver determinadas consultas de clientes. En nuestra línea de investigación las consultas siguen el formato de las consultas contextuales definidas en [15, 16], las cuales se detallarán en la siguiente sección.

Consultas extendidas

La Consulta Contextual, permite a los clientes incluir en sus consultas conocimiento privado y especificar qué tratamiento debe darse a dicho conocimiento, al momento de integrarse con el conocimiento público almacenado en el Servicio de Razonamiento. Esta información es lo que se denomina contexto de la consulta y,

en general, representa conocimiento propio de los clientes.

Tener el conjunto de criterios de preferencias separadas del mecanismo de inferencia brinda muchas ventajas en un modelo de razonamiento cliente-servidor. A continuación mencionaremos algunas de las líneas de investigación a seguir en trabajos futuros.

En primer lugar como están definidas actualmente las consultas de los clientes, no permiten que la consulta contenga los criterios de preferencias que el cliente desea que el servidor utilice para computar su consulta. De esta manera, intuitivamente lo que se busca es configurar el servidor para que razona utilizando las preferencias del cliente.

Otra ventaja de poder incluir sus preferencias en las consultas es poder indicarle al servidor, de todas sus preferencias cuales utilizar al momento resolver la consulta. En definitiva, lo que se le intenta brindar al cliente es mayor capacidad de interacción con el servidor. Para llevar a cabo esta interacción es imprescindible definir operadores específicos para manipular preferencias.

Operadores para manipular preferencias

Los operadores propuestos en [15], son utilizados para modificar temporalmente el conocimiento almacenado e indicar que tratamiento darle al conocimiento privado del cliente. Al agregar las preferencias como un componente más al Servicio de Razonamiento lo que proponemos es incluir operadores que puedan manipular dichas preferencias. A modo de ejemplo algunos de los operadores que proponemos se detallan a continuación:

- \otimes Operador para reemplazar preferencias del servidor.
- \succ Operador para dar prioridad a ciertas preferencias.

- \oplus Operador para agregar preferencias al servidor.

Podemos definir infinidad de operadores, no obstante, estos dependen de la configuración en particular que el cliente le da al servidor al momento de la consulta.

Resultados y Objetivos

Esta línea de investigación tiene como objetivo general formalizar un modelo de Servicio de Razonamiento basado en preferencias, de manera tal que dichas preferencias puedan manipularse de forma dinámica a través de operadores específicos. Para lograr los objetivos propuestos se tomarán como base los resultados obtenidos en los artículos [15, 9, 16].

Formación de Recursos Humanos

Los temas de esta línea de investigación están estrechamente relacionados con el desarrollo de la tesis doctoral del 1º autor del presente artículo.

Referencias

- [1] Alsinet, T., Chesñevar, C.I., Godo, L., Simari, G.R.: A logic programming framework for possibilistic argumentation: Formalization and logical properties. *Fuzzy Sets and Systems* 159(10), 1208–1228 (2008)
- [2] Antoniou, G., Maher, M.J., Billington, D.: Defeasible logic versus logic programming without negation as failure. *J. Log. Program.* 42(1), 47–57 (2000)
- [3] Capobianco, M., Chesñevar, C.I., Simari, G.R.: Argumentation and the dyna-

- mics of warranted beliefs in changing environments. *Autonomous Agents and Multi-Agent Systems* 11(2), 127–151 (2005)
- [4] Capobianco, M., Simari, G.R.: A proposal for making argumentation computationally capable of handling large repositories of uncertain data. In: *SUM*. pp. 95–110 (2009)
- [5] Chesñevar, C.I., Maguitman, A.G., Loui, R.P.: Logical models of argument. *ACM Computing Surveys* 32(4), 337–383 (2000)
- [6] Chesñevar, C.I., Maguitman, A.G., Simari, G.R.: Recommender system technologies based on argumentation 1. In: *Emerging Artificial Intelligence Applications in Computer Engineering*, pp. 50–73 (2007)
- [7] Chesñevar, C.I., Maguitman, A.G., Simari, G.R.: Argument-based user support systems using defeasible logic programming. In: *AIAI*. pp. 61–69 (2006)
- [8] Garcia, A., Simari, G.: Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming (TPLP)* 4, 95–138 (2004)
- [9] García, A.J., Rotstein, N.D., Tucacat, M., Simari, G.R.: An argumentative reasoning service for deliberative agents. In: *KSEM*. pp. 128–139 (2007)
- [10] García, A.J., Simari, G.R.: Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming* 4(1-2), 95–138 (2004)
- [11] Konstan, J.A.: Introduction to recommender systems: Algorithms and evaluation. *ACM Trans. Inf. Syst.* 22(1), 1–4 (2004)
- [12] Loui, R.P.: Defeat among arguments: a system of defeasible inference. *Computational Intelligence* 3, 100–106 (1987)
- [13] Prakken, H., Vreeswijk, G.: Logics for defeasible argumentation. In: Gabbay, D., Guenther, F. (eds.) *Handbook of Philosophical Logic*, vol. 4, pp. 218–319. Kluwer Academic Pub. (2002)
- [14] Resnick, P., Varian, H.R.: Recommender systems - introduction to the special section. *Commun. ACM* 40(3), 56–58 (1997)
- [15] Tucacat, M.: Grupos de Servicios de Razonamiento para el Procesamiento de Consultas Contextuales en Paralelo. Ph.D. thesis, Universidad Nacional del Sur, Bahía Blanca, Argentina (2000)
- [16] Tucacat, M., García, A.J., Simari, G.R.: Using defeasible logic programming with contextual queries for developing recommender servers. In: *AAAI Fall Symposium: The Uses of Computational Argument* (2009)