Interfaz en español para recuperación de información en una base de datos geográfica

Mauricio Osorio Galindo, Moisés Quintero Orea, Rogelio Dávila Pérez

(josorio@mail.udlap.mx, moiquin@yahoo.com.mx, rdav@cs.utep.edu)

Universidad de las Américas-Puebla, Departamento de Ingeniería en Sistemas Computacionales,

Sta. Catarina Mártir, CP 72820, Cholula, Puebla, México

Resumen

En este trabajo se estudió el Procesamiento del Lenguaje Natural utilizando el Sistema XSB como sistema de programación lógica y bases de datos deductivas, NATLIN como reconocedor de oraciones interrogativas en español y Java como interfaz final con el usuario. Se diseñó una arquitectura para una interfaz en español que permite recuperar información geográfica de una base de datos. Se implementó el Sistema JProlog que utiliza InterProlog para obtener desde Java datos procesados en XSB. Se implementó BDNATLIN que es una interfaz en español para recuperar información de una base de datos declarativa. El Volcán Popocatépetl: el caso de estudio.

1 Introducción

Debido a los acontecimientos de los últimos años en la región del Volcán Popocatépetl, ha surgido la necesidad en organizaciones gubernamentales e instituciones de obtener información relacionada a poblaciones, rutas de evacuación, ríos y refugios, pertenecientes a tal región.

Al mismo tiempo, algunas personas con la misma necesidad no tienen acceso a computadoras con capacidad para procesar interfaces sofisticadas de recuperación de información, mientras que otras no utilizan computadoras cotidianamente y consecuentemente no son hábiles en el uso de dispositivos periféricos como el ratón.

Para resolver dicha problemática, es vital que la tecnología sea puesta al alcance de un mayor número de usuarios y por ello se estudió cómo diseñar una interfaz en la que el usuario es capaz de hacer una consulta mediante una oración interrogativa, analizarla siguiendo las técnicas del Procesamiento de Lenguaje Natural (PLN) y obtener información requerida de una base de datos geográfica.

Sin embargo, la aplicabilidad de esta investigación puede trascender hasta la incorporación de interfaces en español a dispositivos electrónicos móviles que son medios potenciales para brindar un servicio en lenguaje natural a sus usuarios, quienes tendrían que efectuar sus preguntas oralmente o en forma de texto de acuerdo a la información deseada.

2 Trabajos relacionados

Las interfaces en lenguaje natural para bases de datos tuvieron un fuerte impulso desde 1972, destacando LUNAR que fue un prototipo construido por William Woods y su equipo de la NASA para recuperar información de rocas lunares de las expediciones Apollo [1] y en 1988 JANUS un sistema desarrollado por BBNLabs e ISI para obtener información referente a la Fuerza Naval de Estados Unidos [2].

A mediados de la década de los 80's el desarrollo de ILNBD comenzó a disminuir debido a que los resultados obtenidos si bien favorables no alcanzaron las expectativas originales. No obstante, algunos sistemas comerciales de ILNBD han sido INTELLECT, BBN´s PARLANCE, IBM´s LANGUAGEACCESS, Q&A, NATURAL LANGUAGE, LOQUI y ENGLISH WIZARD [3].

Por otro lado, debido al crecimiento de la información en forma de texto entre 1995 y el 2000 en páginas web, cartas de correo electrónico y documentos electrónicos especializados se ha incrementado la investigación en el área de procesamiento de voz e interpretación de textos en estos años [4]. Sin embargo, la relevancia de esta investigación radicó en que se estudió el lenguaje español y la información almacenada en la base fue de índole geográfico.

3 Herramientas utilizadas

El procesamiento del lenguaje natural se ha desarrollado en gran medida en el lenguaje Prolog [5], puesto que existe una alianza entre el lenguaje natural y la programación lógica [6]. Por eso XSB, construido

en SUNY Stony Brook siendo sus inventores los investigadores David S. Warren, I. V. Ramakrishnan y Terrance Swift [7], fue seleccionado como sistema de programación lógica y base de datos deductiva ya que además de ser código abierto y extender todas las funcionalidades de Prolog [8], avalan su potencial importantes trabajos como el sistema de modelado de interacción de agentes en programación lógica de Pereira y Quaresma [9] y el proyecto MENTAL en el que se estudian agentes KS y programación lógica dinámica para actualizaciones [10].

Asimismo, XSB es un sistema portable que puede trabajar en computadoras con sistema operativo basado en Unix o Windows e incorpora varias características que regularmente no se encuentran en los sistemas de programación lógica, como interfaces con los sistemas de software C y Oracle. Además la compañía XSB ha construido código propietario basado en su código abierto para importantes clientes como: Reuters, National Science Foundation, Hill's, PartMiner, Defense Logistics Agency y también Argus [7].

Como lenguaje para implementar la interfaz final con el usuario, el lenguaje Java ha mostrado ser una herramienta importante pues además de contar con una biblioteca de clases gráfica y facilitar la conexión a bases de datos, hay evidencias de la completa utilidad de Java para sistemas de memoria distribuidos y para computación paralela a partir de técnicas de implementación eficientes [11].

En la primera parte de nuestro proyecto, se estableció una metodología que permite desde Java recuperar procesos efectuados en XSB en forma controlada y sencilla, con la cual es posible agregar funcionalidades facilitadas por las bibliotecas de clases del lenguaje Java a diferentes módulos implementados en Prolog. Para esto, se implementó el sistema JProlog [12] que permite obtener desde Java términos que pueden ser constantes, variables o listas productos de un procesamiento en XSB, asimismo se pueden enviar comandos en tiempo de ejecución para incrementar la memoria de trabajo con nuevas reglas y hechos.

El sistema JProlog utiliza InterProlog, que es una biblioteca de clases y conjunto de predicados de licencia GNU desarrollados por Miguel Calejo en Portugal, que permiten una comunicación entre Java y XSB utilizando redirección de consola estándar y sockets TCP/IP [13]. Asimismo, tiene la facilidad de invocar métodos Java desde XSB y metas XSB desde Java, sin embargo es difícil tanto la recuperación de listas, como el control del flujo de datos entre XSB y Java, problemas que se resuelven al utilizar el sistema JProlog.

En lo que concierne al entendimiento de una pregunta en español por parte de la computadora destaca el Sistema NATLIN [14] que es una interfaz en español, codificada en Prolog, que permite recuperar información de una base de datos declarativa relacionada a geografía universal. El usuario sólo tiene que introducir una pregunta en forma de texto, después se analiza la oración gramaticalmente conforme al vocabulario descrito en el módulo "lexicon", en caso de ser aceptada, y luego de pasar por un proceso de conversión lambda y simplificación, se transforma a una forma lógica de la cual obtiene una forma clausal que a su vez origina la forma optimizada, y a partir de ésta intenta utilizar los hechos que constituyen la memoria de trabajo del sistema para contestar la pregunta.

4 Arquitectura del sistema

La segunda parte de nuestro proyecto consistió en utilizar las herramientas seleccionadas para desarrollar la interfaz en español en Java que permite efectuar consultas en forma de texto para recuperar información geográfica relacionada al Volcán Popocatépetl organizada conforme a la especificación del Consorcio OpenGIS para mantener y almacenar información geográfica [15] y con esto dejar posibilidades abiertas para enriquecer sistemas implementados en XSB. Para este efecto se diseñó la siguiente arquitectura:

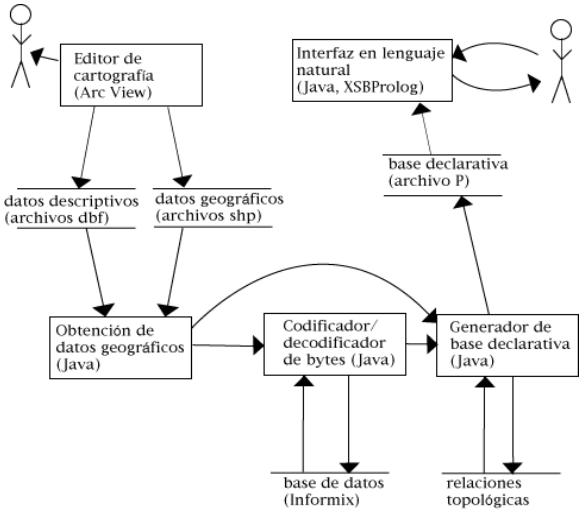

Fig. 1. *Arquitectura del sistema*

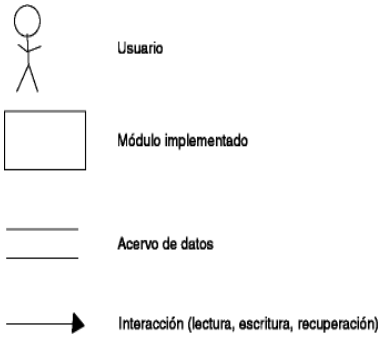

Fig. 2. *Notación del diagrama de la figura 1*

Como se aprecia en la figura 1 el usuario de la izquierda puede generar cartografía a partir de una herramienta de edición de mapas como Arc View, de ahí se generan archivos en formato "dbf" y "shp" que contienen información descriptiva y geométrica, respectivamente, que representan la información geográfica de la cartografía.

Mediante el paquete PaqueteLectores del sistema SIGAU [16] se pueden obtener en memoria desde Java los datos geográficos almacenados en los archivos "dbf" y "shp". Además a través del paquete PaqueteDeBaseDeDatos [16] se puede codificar y decodificar la información geométrica en un atributo de tipo byte para almacenarla y recuperarla en la base de datos del Popocatépetl en Informix.

El generador de la base declarativa a partir de la información geográfica almacenada en memoria puede utilizar algún modelo como el de 9 intersecciones de Beddoe para generar aseveraciones en una base de datos declarativa. La implementación de este módulo implica un manejo de calidad de archivos de texto.

Para los fines de nuestra investigación este procedimiento no se hizo en forma automatizada, pues preferimos concentrarnos en el módulo de la interfaz en lenguaje natural, en el cual en un principio hubo una mayor incertidumbre con relación a los resultados que se podían obtener.

No obstante, debido a la modularidad y generalidad del sistema NATLIN, se implementó en XSB (con cambios mínimos) y se modificaron el módulo del vocabulario de dicho sistema, así como su base declarativa de hechos, por el lenguaje que se utiliza en el dominio del Popocatépetl.

Además, debido al éxito de la herramienta JProlog se implementó BDNATLIN [12] una incorporación de una interfaz gráfica a NATLIN, la cual permite recuperar información geográfica del Volcán Popocatépetl desde una aplicación en Java mediante oraciones interrogativas.

5 Casos de prueba

En la figura 3 se puede apreciar la interfaz gráfica en Java de BDNATLIN en la cual el usuario introduce su pregunta. En un proceso oculto para el usuario final se conecta la interfaz a XSB mediante JProlog, con la conexión abierta se carga en memoria el sistema NATLIN y se le envía la petición del usuario para que después de consultar la base de datos declarativa, recupere los resultados desde Java y luego de pasar por un proceso final de análisis de unidades lexicográficas se despliega el resultado en el área de texto.

Además, se pudieron efectuar las preguntas descritas en la tabla 1. En la columna de la izquierda de la tabla se indica la categoría (según nuestra propia clasificación) correspondiente a cada pregunta ejemplo.

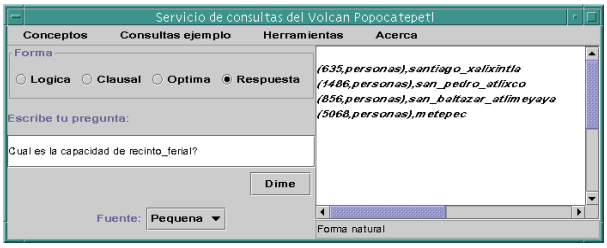

***Fig. 3**. Caso de prueba en Solaris*

Preguntas	Ejemplos
Generales	Que comunidades son de peligro_mayor?
Atributos de entidades conceptuales	Cual es la población de santiago_xalixintla?
Valores específicos de atributos	Que comunidades tienen una ruta que es num_2?
Relaciones topológicas	Que comunidad colinda con una comunidad cuyo municipio es calpan?
Con excede	Que comunidad tiene una poblacion_a_evacuar que excede a la poblacion de metepec?
Con pertenece	Que refugios pertenecen a san_mateo_ozolco?
En forma negativa	Que comunidades no colindan con textepec?
Compuestas	Que comunidades que no colindan con tenextepec tienen una ruta que es num_2?

***Tabla 1**. Algunas preguntas efectuadas en BDNATLIN*

6 Conclusiones y trabajo futuro

Con el enlace de calidad y controlado entre Java y XSB, manifestado en la implementación de BDNATLIN, se tiene una herramienta con la cual se pueden enriquecer las funcionalidades de los sistemas propios de Inteligencia Artificial implementados en lenguajes lógicos declarativos, ya que mediante JProlog se pueden incorporar a sistemas implementados con el paradigma de programación lógica aspectos de programación en red, imágenes, video, sonido e interfaces gráficas con botones y formas.

En esta investigación adoptamos la postura de generar una base declarativa a partir de una base de datos relacional geográfica porque nos concentramos en el módulo de lenguaje natural para mostrar la utilidad de JProlog tanto para enriquecer las funcionalidades de sistemas en XSB, como para poner al alcance los sistemas de programación lógica a los programadores

que son expertos en Java, cuyo paradigma es orientado a objetos. Pero otro enfoque consiste en transformar una pregunta en español a un lenguaje estructurado de consultas para evitar tener una base de datos declarativa, para ello una de las alternativas más apropiadas es la utilización de una gramática de árbol semántico.

Desde luego, podemos afirmar que se ha dado un paso importante para que el usufructo de la tecnología beneficie a un mayor número de personas ya que se ha facilitado la incorporación de interfaces de este tipo a interfaces multimodales. Se ha incrementado la posibilidad de continuar trabajando en sistemas de reconocimiento de voz en español que permitan recuperar información de un acervo concentrándose en un dominio particular del lenguaje para resolver tareas específicas más que en entender una lengua completamente y mantener una discusión. Desde esta perspectiva se podría investigar si es factible desarrollar un sistema en que los usuarios pudiesen efectuar consultas en español vía telefónica a un servidor generador de respuestas, conectado a su vez a la base de datos con información del Popocatépetl.

Asimismo, un proyecto que se puede desprender de esta investigación es el estudio de cómo generar una base de datos declarativa a partir de documentos XML, ya que debido a sus propiedades estructurales podría ser precisamente el vínculo de comunicación entre bases de datos declarativas, relacionales y documentos de texto.

También es posible continuar trabajando en esta línea de investigación en relación a cómo efectuar actualizaciones de una base de datos resolviendo los problemas de consistencia que pudiesen manifestarse, cómo efectuar preguntas acerca de la información que contiene una base de datos, y cómo diseñar una arquitectura para resolver preguntas temporales.

Agradecimientos

Esta investigación fue parcialmente financiada por el Consejo Nacional de Ciencia y Tecnología bajo el proyecto W35804 “Access to High Quality Digital Services and Information for Large Communities of Users”.

Referencias

[1] Russell, Stuart y Norvig, Peter. *Artificial Intelligence. A modern approach.* Prentice-*Hall*, USA, 1995. Págs. 691-723.

[2] Hinrichs, Erhard W. “Tense, Quantifiers and Contexts”. *Computational Linguistics*. Vol. 14, No.2, Junio, 1988.

[3] Androutsopoulos, Richie, G. D., Thanisch, P. Natural language “Interfaces to Databases-An Introduction”. Research Paper no. 709, Department of Artificial Intelligence, University of Edinburgh, 1994.

[4] Jurafsky, Daniel y Martin, James H. *Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition.* Prentice – Hall, USA 2000.

[5] Gazdar Gerald and Mellish Chris. *Natrual Language Processing In Prolog. An Introduction to Computational Linguistics.* Addison-Wesley, 1989.

[6] Dahl, V. “Tutorial: Natural Language Understanding and Logic Programming”. *ICLP97*, Leuven,1997. http://www.cs.sfu.ca/fas-info/cs/people/Faculty/Dahl/personal/ILPS97/ICLP97.tutorial/html.html

[7] http://www.xsb.com/coretechnology.asp

[8] Sagonas Konstantinos, Swift Terrance, Warren David S., Freire Juliana, Rao Prasad. “The XSB System Versión 2.2 Volume 1: Programmers Manual”. Abril, 2000.

[9] Sadri, Fariba y Toni, Francesca. “Computational Logic an Multi-Agent Systems: a RoadMap”. Department of Computing, Imperial College. Diciembre, 1999. Pág. 18. Reino Unido. http://www-lp.doc.ic.ac.uk/~{fs,ft}

[10] Eiter, Thomas, et. al. “Using Methods of Declarative Logic Programming for Intelligent Information Agents”. INFSYS Research Report. Octubre 2000. Viena, Austria.

[11] Kielmann, Thilo, et. al. “Enabling Java for High-Performance Computing”. *Communications of the ACM*. Octubre 2001. Vol. 44, No. 10.

[12] Quintero, Moisés. “Interfaz en español para recuperación de información en una base de datos geográfica”. *Tesis Profesional. Universidad de las Américas – Puebla*. Ingeniería en Sistemas Computacionales. Otoño 2001.

[13] Calejo, Miguel. “Introduction to InterProlog”. Noviembre 16, 1998. http://www.declarativa.com

[14] Dávila, Pérez R, “Una Interfaz en Español para Bases de Datos expresadas en Lógica, a través de Lógica”. *Memorias de la IV Reunión Nacional de la SMIA.* Puebla, México (Marzo 1987).

[15] Razo, Antonio y Sol, David. “Standard 2D and 3D geo-spatial data formats for a Volcano Geographic Information System”. *SMCC Enc’01. Memoria 3er Encuentro Internacional de Ciencias de la Computación. Tomo II.* Aguascalientes, México, 2001. Págs 737-744.

[16] Gómez, Humberto Ariel. “Sistema de información geográfica para el análisis de catástrofes ubanas”. *Tesis Profesional. Universidad de las Américas – Puebla.* Ingeniería en Sistemas Computacionales. Primavera 2001.