

Conversión de expresiones temporales en español a SQL *

Conversion of temporal expressions in Spanish to SQL

Ismael Esquivel, Josué León, Ma. de los Ángeles Marrujo

Universidad Veracruzana

Veracruz, Veracruz, México

iesquivel@uv.mx, jlc_2006@hotmail.com, angeles_marrujo@hotmail.com

Resumen: El filtrado de resultados de una consulta a una base de datos es una tarea recurrente que permite a los usuarios finales, una mayor precisión en sus búsquedas, teniendo al filtrado temporal como uno de sus casos más representativos. Este trabajo presenta un sistema para el reconocimiento de expresiones temporales (ET) en español, presentes en una consulta realizada en lenguaje natural a una base de datos relacional y su conversión a condiciones de filtrado del comando SELECT del SQL; formando parte de una interfaz de lenguaje natural a base de datos (ILNBD) aun en construcción.

Palabras clave: Expresiones temporales, ILNBD, SQL

Abstract: Filtering the results of a database query is a recurrent task that allows end users a greater accuracy in their searches, and the filtering on a temporary basis is a representative case. This paper presents a system for the recognition of temporal expressions (TE) in Spanish found in a natural language query made to a relational database and its conversion to filtering conditions of the SQL SELECT command. It is a module of a natural language interface to database (NLIDB) still under construction.

Keywords: Temporal expressions, NLIDB, SQL.

1 Introducción

Las interfaces de lenguaje natural a bases de datos (ILNBDs) permiten a los usuarios finales formular sus necesidades de información en lenguaje natural y cubrirlas, sin la intervención de personal del área de Informática ó T.I.

El filtrado de los resultados de una consulta a una base de datos, a partir de sus atributos temporales, es una necesidad recurrente en los usuarios finales. Cuando éstos expresan sus necesidades de información en lenguaje natural, usan expresiones temporales para obtener resultados más específicos (Ej. *¿Cuales proyectos terminaron en el segundo bimestre de 1986?*). De acuerdo con (Ahn, D., Fissaha, S. y Rijke, 2005), "Las expresiones temporales (también denominadas timexes) son fragmentos del

lenguaje natural que aluden directamente a instantes en el tiempo o a intervalos."

Conforme a (Androustopoulos y Ritchie, 2000) para una ILNBD, su manejo impone serios retos por las características que distinguen a las consultas que contienen expresiones temporales, ya que requieren la combinación de tres áreas vinculadas: teorías de aspecto y tiempo, lógica temporal y sistemas de bases de datos temporales.

Aunque dichas expresiones pueden indicar la fecha de forma absoluta (Ej. *12 de enero de 1999*), la mayor complejidad proviene de relaciones anafóricas o expresiones deícticas, como por ejemplo: *hace dos días, la semana pasada, este domingo*, etc.

Este documento describe una propuesta, que forma parte de una ILNBD en español denominada SNL2SQL, aún en desarrollo. Sirve para la conversión de diversos tipos de expresiones temporales a condiciones de filtrado del comando SELECT del SQL, bajo el estándar ANSI-92, el cual es soportado por la mayoría de los motores de bases de datos relacionales.

* Este trabajo ha sido subvencionado por la Secretaria de Educación Pública a través del Programa de Mejoramiento del Profesorado (PROMEP). Oficio No. 103.5/09/4482.

2 Trabajos relacionados

Androutopoulos (2002) en su obra, maneja preguntas temporales en inglés para una base de datos de aeropuertos. Consiste de un analizador que convierte preguntas que contienen términos temporales a expresiones en lenguaje TOP. Luego define una metodología para convertir expresiones TOP a instrucciones SQL para bases de datos temporales, en cuyo caso usaron un gestor de base de datos denominado TIMEDB. El lenguaje TOP maneja constantes y variables, de manera que aplicando símbolos de predicados a argumentos se pueden generar fórmulas atómicas. Éstas se traducen a TSQL2, el cual contiene extensiones del SQL para el manejo temporal, para lo cual se requiere previamente del mapeo de constantes y predicados a expresiones TSQL2.

Existen alternativas de solución para el idioma español, aunque orientadas hacia sistemas Q&A. Entre ellas destacan la de Martínez-Barco, Saquete y Muñoz (2002), quienes presentan un modelo basado en una gramática para el reconocimiento y traducción, que permite mapear datos de entrada al formato estándar (*dd/mm/yyyy*). Para ello, los textos son etiquetados con información léxica y morfológica para que sirvan de entrada a un analizador temporal, el cual utiliza una técnica ascendente basada en gramática con dos diferentes clases de reglas, tanto para manejar expresiones no anafóricas, como para aquellas anafóricas.

En esa misma línea, Vicente-Díez, De Pablo-Sánchez y Martínez, (2007) presentan un sistema que procesa cada entrada siguiendo cuatro etapas secuenciales. En la primera, un preprocesador formatea el texto de entrada y le añade información de posición, etiquetado gramatical, morfosintáctico y semántico. Para la segunda, detecta los *timexes* existentes en cada oración del texto de entrada. En la tercera, se normaliza lo detectado, en el formato estándar (*yyyy-mm-dd*), dependiendo del tipo de expresión: absoluta, anafórica, intervalos y festividades. En la última etapa, se expresan los resultados en un formato de salida XML especial.

En su trabajo, Galicia-Haro y Gelbukh (2009), realizan el reconocimiento de expresiones temporales relacionadas con la edad. Presentan un método para enriquecer clases de frases temporales cuando se tienen pocos ejemplos. Por cada clase se generó una muestra más representativa usando Internet. Definieron el contexto para dichas frases para poderlas identificar automáticamente, obteniendo un contexto específico para nueve de veintidós clases. La clasificación se realizó con el algoritmo Árbol de decisiones basado en atributos directos.

3 Metodología

3.1 Tipos de preguntas temporales

Para lograr mejores resultados, se hace necesaria la clasificación de las consultas temporales que los usuarios pueden hacer a una base de datos. Para ello, se trabajó con la propuesta de Saquete y otros (2009), también para sistemas Q&A, en la cual existen cuatro categorías considerando el número de eventos y la existencia o no de expresiones temporales.

Por otro lado, expresiones tales como: *el año pasado, en 1990*, etc., denotan un filtrado de respuestas, estableciendo límites temporales. En la tabla 1, se muestra cada categoría con un ejemplo a partir de la base de datos usada como prueba.

Cat.	Característica	Ejemplo
1	Un evento sin expresión temporal	<i>¿Cuándo inició el proyecto OPERATION?</i>
2	Un evento con expresión temporal	<i>¿Quiénes ingresaron a la empresa después del 2000?</i>
3	Varios eventos sin expresión temporal	<i>¿Quiénes ingresaron a la empresa en el año que arrancó el proyecto OPERATION?</i>
4	Varios eventos con expresión(es) temporal(es)	<i>¿Cuántos proyectos iniciaron en la crisis del 94 y 95?</i>

Tabla 1. Categorías de preguntas temporales

Para el presente trabajo, se trataron solamente peticiones de los tipos 2 y 4, sin incluir el tratamiento de horas y para sistemas de bases de datos atemporales.

3.2 Arquitectura del sistema

El sistema propuesto analiza el texto de una consulta expresada por escrito en lengua castellana y la convierte a una condición de la cláusula WHERE del comando SELECT, para el filtrado de respuestas. De modo que una consulta como:

¿Quiénes ingresaron entre 1950 y 1960?

Se convierte al texto:

“ and EMPL.HIREDATE between ‘01/01/1950’ and ‘31/12/1960’ ”

Dado que en una pregunta temporal, puede aparecer más de una expresión, es necesario un procesamiento iterativo hasta la conversión total. Por lo mismo, el proceso conformado de tres etapas secuenciales, como se muestra en la figura 1, se repite por cada expresión encontrada.

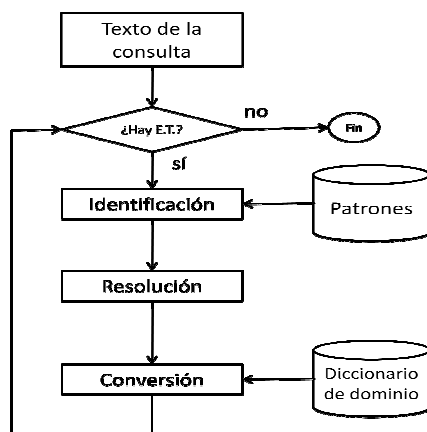


Fig. 1. Etapas del proceso

3.2.1 Identificación

En la etapa de identificación, se detecta la expresión temporal (E. T.) en el texto de la consulta, a partir de los términos temporales

que se muestran en la tabla 2, organizados en categorías. Dichos términos actúan como disparadores para ser procesados en combinación con artículos, demostrativos, adjetivos, adverbios temporales, verbos, preposiciones, conjunciones y contracciones contenidos en la tabla 3. Los patrones así conformados, se muestran parcialmente en la tabla 4, en la cual el signo “+” implica concatenación, “[]” indican elementos opcionales y “()” sirven para compendiar opciones, las cuales son separadas con “|”.

3.2.2 Resolución

En la etapa de resolución, se plantea que para toda expresión temporal incluida en la consulta se obtenga un rango de fechas, por lo cual se generan dos valores en el formato dd/mm/yyyy, uno para la fecha inicial (FechaIni) y otro para la final (FechaFin). Estos datos se usan en conjunción con el operador BETWEEN para conformar el filtrado temporal.

En esta etapa, se procede a la obtención de las fechas, conforme al tipo de expresión, que como se ha planteado, puede ser absoluta o anafórica. De modo que, si se trata del tipo absoluto, ya sea completa (“15 de marzo de 2010”) o incompleta (“26 de Febrero”), al no requerirse de una fecha de referencia, la generación del rango es casi directa.

En caso de tratarse del tipo anafórico, dada la complejidad que representan, es necesario considerar como referencia la fecha actual, para obtener datos correctos. En el caso de que no fuera posible la resolución, la consulta se marca como “No Resuelta”.

Para expresiones sobre festividades, dado que la totalidad son puntos en el tiempo, los valores se obtienen directamente, tomando en cuenta el texto adicional que indica el año de ocurrencia (Ej. “la navidad pasada”), aun cuando para casos como el Carnaval, se necesita de ajustes.

CATEGORIA	TERMINOS	EJEMPLO
AdverbioTiempo	(hoy ahora anteayer antier ayer mañana anoche anteanoche pasado mañana antes de ayer antes de anoche al medio día por la noche hoy en día hoy día)	<i>ayer</i>
DD	(1 a 31)	<i>15</i>
MM	(1 a 12)	<i>10</i>
Mes	(enero febrero ... diciembre ene feb ... dic ene. feb. ... dic.)	<i>diciembre</i>
Basico	(DD-MM-[YY]YY DD/MM/[YY]YY DD/MM/[YY]YY)	<i>02/12/2005</i>
BasicoInvertido	(YYYYMMDD YYYY-MM-DD YYYY/MM/DD YYYY/Mes/DD)	<i>20051210</i>
DiaDeMesDeAño	DD de Mes de YYYY	<i>5 de marzo de 2005</i>
DiaDeSemana	(lunes martes miércoles jueves viernes sábado domingo)	<i>domingo</i>
DiaDeSemana y DiaDeMesDeAño	DiaDeSemana DD de Mes de [YY]YY	<i>lunes 12 de abril de 2010</i>
DiaFestivo	(año nuevo día de reyes día de la madre navidad fin de año...)	<i>navidad</i>
Estación	(primavera verano otoño invierno)	<i>invierno</i>
UnidadDeTiempo	(día semana quincena mes bimestre cuatrimestre trimestre semestre año bienio trienio lustro quinquenio sexenio siglo)	<i>mes</i>

Tabla 2. Términos temporales

Elemento	Ejemplos
Preposiciones	A, de, desde, dentro, en, entre, durante, para, por
Conjunciones	Y
Artículos	El, la, los, las
Verbos	Hacer, iniciar, comenzar, terminar, acabar, ...
Adverbios temporales	Después, antes, luego, enseguida, ...
Adjetivos	Próximo, pasado, siguiente, ...
Contracciones	Al, del
Pronombres demostrativos	Este, esta
Cantidad numérica	1, 2, 3, 4,...
Cantidad textual	Un, una, dos, tres, cuatro,...
Cantidad ordinal	Primera, segunda, tercera, ...
Conectores	De, del

Tabla 3. Elementos adicionales

3.2.3 Conversión

En la etapa de conversión, se genera la condición que se agrega a la clausula WHERE para el filtrado temporal. Para ello, a partir de los elementos restantes de la consulta, se determinan la tabla y columna correspondientes, para que asociados a los valores, obtenidos en la etapa anterior, se genere un texto como el siguiente:

' and '|Tabla.Columna|' between '|fechaIni|' and '|fechaFin

La tabla y columna se obtienen a partir de un diccionario de dominio construido previamente, de la base de datos de trabajo y sólo para los atributos cuyo tipo de dato es fecha o fecha y hora. El diccionario contiene por cada atributo elegible: Nombre de la tabla, Nombre de la columna y los términos que califican al atributo fecha junto con sus sinónimos y las diferentes formas verbales asociadas. Algunas entradas del diccionario se aprecian en la tabla 5.

[AdverbioOrden] + Cantidad(<i>Numerico</i> <i>Textual</i>) + (UnidadTiempo DiaDeSemana) [+ AdverbioOrden]	Próximos 7 días Sigüientes 2 Meses Cinco meses sigüientes
AdverbioTemporal + Contracción + Día (<i>Numerico</i> <i>Textual</i> <i>Ordinal</i>) [+ Conector + Mes] [+ Conector + AñoNumerico]	Antes del 21 Luego del veintiuno de Enero Después del 21 de Enero de 2009
[AdverbioTemporal + conector] Demostrativo + (DiaDeSemana Mes AñoNumerico UnidadTiempo Estacion DiaFestivo)	Esta Semana Este Marzo Este Otoño Antes de este año Antes de este verano
Contracción (“Del”) + Día (<i>Numerico</i> <i>Textual</i> <i>Ordinal</i>) [+ Conector + Mes] [+ Conector + AñoNumerico] + Contracción (“al”) + Día (<i>Numerico</i> <i>Textual</i> <i>Ordinal</i>) [+ Conector + Mes] [+ Conector + AñoNumerico]	Del 5 al 21 Del 21 de Enero al 10 de Marzo Del 21 de Enero de 2009 al 10 de Marzo de 2010
Preposición [+ artículo (“el”)] + (DiaDeSemana Mes [+ conector + AñoNumerico] AñoNumerico DiaFestivo + DD(<i>Numerico</i> <i>Textual</i>)) [+ (Preposición Conjunción [+ artículo]) + (DiaDeSemana Mes [+ conector + AñoNumerico] AñoNumerico DD (<i>Numerico</i> <i>Textual</i>) [“la fecha” “acá”]]	Entre Lunes y Miércoles De Enero hasta Marzo Entre 2009 y 2010 De agosto a diciembre De 2007 a la fecha
DD + Conector + Mes Mes + DD Mes + conector + [YY]YY	15 de abril Mayo 30 Marzo de 2003 Enero del 90
“En lo que va” + conector [+ articulo] + UnidadTiempo	En lo que va del año En lo que va de la semana
AdverbioTemporal + Conector [+ artículo] + [Cantidad(<i>Ordinal</i>)] + (UnidadTiempo) [+ Conector + Mes] [+ Conector + AñoNumerico] [+ Conector + artículo + Cantidad(<i>Textual</i>)]	Después de la segunda semana de Marzo Luego del tercer bimestre de 2005 Antes de la década de los noventa
Verbo (“Iniciar, comenzar, empezar”) + Preposición [+ artículo] + (Mes AñoNumerico Estacion DiaFestivo) [+ conjunción + Verbo (“Terminar, acabar, finalizar”) + Preposición [+ artículo] + (Mes AñoNumerico Estacion DiaFestivo)]	Comenzando en febrero Iniciando en 2005 Arrancando en mayo y terminando en agosto
[AdverbioOrden +] (DiaDeSemana Mes AñoNumerico UnidadTiempo Estacion DiaFestivo) [+ AdverbioOrden]	Próximo Domingo Pasado Marzo Enero próximo Año pasado

Tabla 4. Muestra de algunos patrones

Tabla	Columna	Calificador	Sinónimos	Formas Verbales
Empl	Birthdate	Nacimiento Nació	Alumbramiento, origen, principio	Nació, nacieron,
Empl	Hiredate	Ingreso Ingresó	Entrada, contratación, alta, admisión	Contratados, ingresaron, iniciaron, contrataron,

Tabla 5. Muestra del diccionario de dominio

El diccionario de dominio se propuso construirlo a partir de un diccionario de sinónimos y del metadata de la base de datos, apoyado en el trabajo de Pazos et. al (2005), como se aprecia en la figura 2. Del metadata se extraen por cada tabla, la columna elegible y su descripción. A partir del procesamiento de ésta, se obtiene el término calificador de cada fecha. Para identificarlo plenamente, se consideran sólo descripciones como las siguientes: “Fecha de venta”, “Fecha de inicio del semestre”, “Cuando inició el proyecto”, “Cuando nació el empleado”, etc.

En estos casos, el término aparece como sustantivo o verbo en pasado, por lo cual se procede a buscarlo en el diccionario de sinónimos para completar la tabla 5. Para este trabajo, el diccionario de sinónimos se ha construido manualmente usando el diccionario en línea que proveen Rodríguez y Carretero (2008), aunque la idea es generarlo de forma automática en el futuro.

De la pregunta formulada se discrimina la expresión temporal previamente procesada. Los términos restantes (sustantivos, adjetivos ó verbos) son buscados en las entradas del diccionario para identificar la tabla y columna.

De no encontrarse se marca como “Sin Respuesta”, indicando que la consulta contiene términos no considerados en el diccionario de sinónimos. En algunos casos, se encuentran elementos calificadores que tienen sinónimos y formas verbales similares, en cuyo caso, a la consulta se le califica como “Ambigua”.

3.3 Consideraciones

Para esta versión inicial del sistema, se establecieron solamente ciertos verbos para indicar inicio y fin de los eventos, como son: empezar, iniciar, comenzar, terminar, acabar, concluir, finalizar.

Además, las descripciones de los atributos de tipo fecha, que fueron susceptibles de procesarse, responden a ciertos patrones solamente:

- “Fecha “ + (“de” | “del”) + <sustantivo> (Ej. Fecha de ingreso)
- “Fecha “ + (“de” | “del”) + <sustantivo> del <objeto> (Ej. Fecha de inicio del proyecto)
- “Cuando ” + <verbo en pasado> + “el” + <objeto> (Ej. Cuando inició el proyecto)
- “Cuando “ + <verbo en pasado> + “el” + <objeto> (Ej. Cuando nació el empleado)

Por otro lado, cuando la expresión temporal se unía a los artículos determinados (el, la) y demostrativos (este, esta) y hacían falta datos para un procesamiento correcto, fue necesario tomar los siguientes criterios para lograr la resolución:

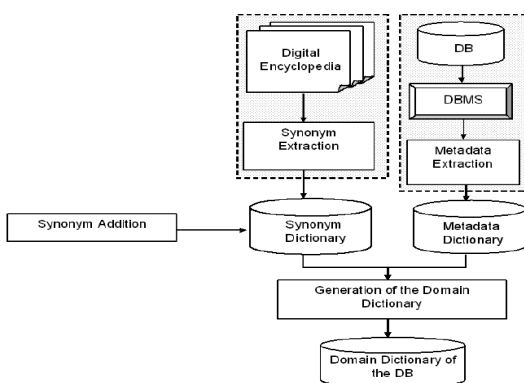


Fig. 2. Generación del diccionario de dominio.

Fuente: A Domain Independent Natural Language Interface to Databases Capable of Processing Complex Queries.

“Este <mes>”. Si el mes de la fecha actual es anterior a julio, se considera al mes del mismo año o del pasado, de lo contrario, el del próximo año.

“Este <diadelasemana>”. Si el día de la fecha actual es anterior a jueves, se toma al día de la misma o pasada semana, en caso contrario, de la siguiente.

“Este <diafestivo>”. Según el mes que pertenezca se aplica un criterio similar a los meses.

“Este(a) <estación>”. Si la estación actual es anterior al otoño, se asigna la estación del año actual o pasado, en forma opuesta, la del próximo año.

Adicionalmente, no se consideró la resolución de expresiones anafóricas que incluyeran artículos indeterminados (Ej. *Hace unos días, de unos meses para acá*, etc.)

Cuando en la pregunta formulada se hacía referencia a un evento histórico correspondiente a una fecha o periodo (Ej. *Dos días después del terremoto del 85*), no pudo ser tratada como expresión anafórica.

3.4 Implementación

Se utilizó como base de datos de prueba, la mostrada en la figura 3, en la cual se tienen cuatro atributos de tipo fecha (*Birthdate, Hiredate, PrStdate, PrEnddate*). El corpus de peticiones en lenguaje natural, fue conformado a partir de la presentación de la Base de Datos y explicación del proyecto a tres diferentes públicos del área de T.I.: alumnos de licenciatura de una IES pública, alumnos de maestría de universidad privada y maestros de ambas instituciones. Se obtuvieron 33, 22 y 13 peticiones respectivamente, luego de descartar las similares.

El lenguaje de programación utilizado para el desarrollo de los diferentes módulos es Rexx (REstructured eXtended eXecutor), un lenguaje de programación desarrollado en IBM por Michael Cowlshaw del que existen numerosas implementaciones disponibles con código abierto (Mertz, 2004). Se eligió porque cuenta con un gran conjunto de funciones, especialmente de tratamiento de textos y de fechas.

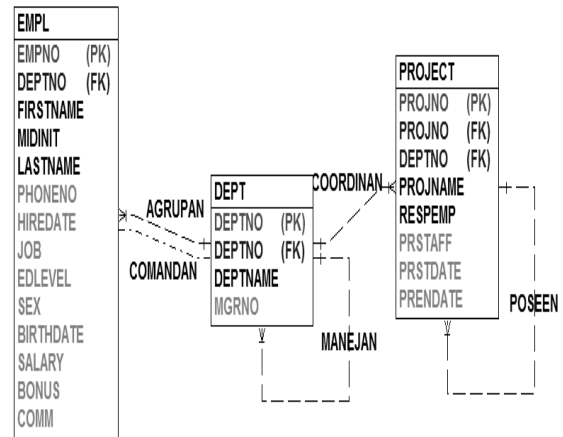


Fig. 3. Diagrama lógico relacional de la BD de prueba.

4 Resultados

El proceso de conversión, tomaba las preguntas cargadas en un archivo y generaba una salida como la mostrada en la figura 4.

```

C:\Users\ADTUSI\Desktop\traductores\Traductor29>rexx version15

MUESTRA LOS EMPLEADOS CONTRATADOS DURANTE 1990
WHERE Empl.Hiredate BETWEEN '01/01/90' and '31/12/90'

QUE PROYECTOS FINALIZARON EL 15 DE MARZO DE 2009
WHERE Project.Prenddate BETWEEN '15/03/09' and '15/03/09'
  
```

Fig. 4. Muestra de la salida del proceso de conversión

A partir del corpus generado, que constó de 68 peticiones, se obtuvieron los resultados de la tabla 6, en la cual las preguntas “No Resueltas”, contenían expresiones temporales que no fueron procesadas correctamente. Las denominadas “Sin Respuesta” y “Ambiguas” fueron resueltas, pero no se obtuvo la clausula WHERE de forma correcta.

En general, los resultados son bastante prometedores para tratarse de una primera aproximación. El número de expresiones temporales no procesadas correctamente fue muy bajo, detectando en esos casos el uso de expresiones incorrectas (Ej. “*el periodo de 1985 – 1988*”, “*Durante la década de 1990 – 2000*”, etc.), por parte de los

usuarios, aun cuando sí se encontraron expresiones que no se habían considerado.

Aunque representativa la proporción de preguntas sin respuesta, se encontraron dos causas principales: La primera, que los usuarios hicieron peticiones orientadas hacia una base de datos de tipo temporal (Ej. *¿Cuántos programadores trabajaron en el Proyecto User Education hace dos meses?*, *¿Cuáles trabajadores estuvieron en el departamento Operations, en diciembre pasado?*, etc.), las cuales no fueron consideradas para este trabajo. La segunda, que hicieron peticiones fuera del contexto de la base de datos (Ej. *“Dame las ventas del viernes pasado”*, *“Dame las ganancias obtenidas después de la segunda semana de marzo”*, etc.). Si este tipo de preguntas se descartan, aumenta notablemente la efectividad. Sin embargo, se encontraron tres casos no considerados (*“Cumplen años en”*, *“cumpleaños cae en”*, *“Nacieron en el mes de”*, etc.), que requieren de procesamiento adicional. En cuanto a las ambiguas, se plantea trabajar en el futuro para reducir las al máximo.

Preguntas Tipo de usuario	Formuladas	No Resueltas	Sin Respuesta	Ambiguas	Convertidas (%Efectividad)
Estudiantes de Licenciatura	33	4	10	1	54.5%
Estudiantes de Maestría	22	2	7	1	54.5%
Maestros	13	2	3	0	61.5%
Total	68	8	20	2	55.9%

Tabla 6. Resultados obtenidos

5 Conclusiones y trabajo futuro

Se han detallado las características del sistema para la conversión de expresiones absolutas y deícticas encontradas en consultas formuladas a una base de datos de prueba. Se ha encontrado que varias de las peticiones de los usuarios, implican el manejo de una base de datos temporal y por tanto, el mejoramiento del presente trabajo. Sin embargo, la efectividad lograda hasta el

momento, sin considerar ese tipo consulta, anima al desarrollo de una versión mejorada. Además, se reconoce que el tratamiento de expresiones deícticas, implica tomar criterios que pudieron ser distintos a los presupuestos por los usuarios finales, por lo cual podrán existir diferencias sensibles y, aun cuando no se cuenta con trabajos previos contra los cuales comparar, los resultados son muy alentadores, lo cual conduce a plantear las siguientes actividades futuras:

- El manejo de patrones adicionales en las descripciones de los atributos tipo fecha y la reducción de casos de ambigüedad.
- La prueba con otras bases de datos que contengan un mayor número de atributos de tipo fecha y la conversión de expresiones basadas en horas.
- La generación automática del diccionario de sinónimos y la verificación automatizada de la conversión, conforme avance el desarrollo de la ILNBD.
- Finalmente, el desarrollo de una versión mejorada que soporte bases de datos temporales, para mejorar la efectividad, es una línea de trabajo prioritaria.

Bibliografía

- Ahn, D., Fissaha, S. y de Rijke, M. 2005. Extracting Temporal Information from Open Domain Text: A Comparative Exploration. *J. Digital Information Management*, 3(1):14-20.
- Androutsopoulos, I. y Ritchie, G.D. 2000. "Database Interfaces" In R. Dale, H. Moisl, and H. Somers (Eds.), *Handbook of Natural Language Processing*, chapter 9, pp. 209-240, Marcel Dekker Inc.
- Androutsopoulos, I. 2002. *Exploring Time, Tense and Aspect in Natural Language Database Interface*. Benjamins John, *Natural Language Database Interfaces*. (Natural Language Processing Series, Vol. 6), ISBN 9027249903
- Galicia-Haro, S. y Gelbukh, A. 2009. "Supervised Recognition of Age-Related Spanish Temporal Phrases". *MICAI 2009: Lecture Notes in Computer Science N 5845*, Springer, 2009, ISBN 978-3-642-05257-6, pp. 145-156.
- Martínez-Barco P., Saquete E. y Muñoz R. 2002. "A Grammar-Based System to Solve Temporal Expressions in Spanish Texts". *PorTAL 2002*: 53-62
- Mertz David. 2004. *Rexx for everyone: Scripting with Free Software Rexx implementations*.
<http://www.ibm.com/developerworks/library/l-rexx.html>
- Pazos R., Pérez J., González J. J., Gelbukh A., Sidorov G. y Rodríguez M. 2005. "A Domain Independent Natural Language Interface to Databases Capable of Processing Complex Queries". *MICAI 2005: Advances in Artificial Intelligence*. *MICAI 2005*: 833-842
- Rodríguez S. y Carretero J. 2008. *COES: Herramientas para Procesamiento de Lenguaje Natural en Español*.
<http://www.datsi.fi.upm.es/~coes/interactivo/sinonimos.cgi>
- Saquete E., Vicedo J. L., Martínez-Barco P., Muñoz R. y Llorens H. 2009. "Enhancing QA Systems with Complex Temporal Question Processing Capabilities". *Journal of Artificial Intelligence Research Volume 35*, pages 775-811.
- Vicente-Díez, M. T., De Pablo -Sánchez, C. y Martínez, P. 2007. "Evaluación de un sistema de reconocimiento y normalización de expresiones temporales en español". *Procesamiento del lenguaje natural. REV - PLN - Nº 39* (septiembre 2007); pp. 113-120. ISSN:1135-5948