



Universitat d'Alacant
Universidad de Alicante

Interfaz en Lenguaje Natural en sistemas de
toma de decisiones para tratamiento de
preguntas contextuales con anáfora y elipsis

Yenory Rojas Hernández



Tesis

Doctorales

www.eltallerdigital.com

UNIVERSIDAD de ALICANTE



Universitat d'Alacant
Universidad de Alicante

Departamento de Lenguajes y Sistemas
Informáticos
Universidad de Alicante

Interfaz en Lenguaje Natural en sistemas de
toma de decisiones para tratamiento de
preguntas contextuales con anáfora y elipsis

Yenory Rojas Hernández

Tesis presentada para aspirar al grado de
DOCTORA POR LA UNIVERSIDAD DE
ALICANTE

Universidad de Alicante

Dirigida por:

Dr. Jesús Peral Cortés y Dr. Antonio
Ferrández Rodríguez

Alicante, junio 2016

Agradecimientos

Alabad a Jehová, porque él es bueno;
Porque para siempre es su misericordia.

Salmo 118: 1

El proceso de tesis es un camino empinado y nada fácil, este se logra con el soporte de Dios y con las personas que pone al lado para apoyar en el proceso, en mi historia debo agradecer a:

Don Álvaro Castro, por hacer llegar a mí, el formulario de la beca, y ayudarme con toda la gestión de la entrega de documentos.

Al Banco Santander, por la beca de estudio durante mi estancia en Alicante, al inicio del Doctorado.

A la Universidad Hispanoamericana por concederme el permiso laboral para poder ir a la defensa.

En el ámbito académico, no tengo manera de agradecer a mis tutores Jesús y Antonio (el dúo dinámico). A Jesús que le ha tocado perseguirme y jalarme, Antonio que no desistió con su labor de tutor y no me abandonó. Esto de trabajar en otro país y seguir con la tesis, es una labor ardua para todo el equipo.

A mis papás, que luchan a mi lado para que crezca y logre las metas propuestas, papi ahora en el cielo, pero sé que está. Y mami que me ha acompañado en mis desveladas, me prepara cafecitos de madrugada para que siga dándole. Gracias por creer en mí y amarme tanto.

iv Agradecimientos

A mis hermanos Isaac, Verny, David que permanecen como apoyo y ayuda. Los amo con todo mi corazón.

A mi cuñada, Lilly que también le tocó trabajar, revisando filológicamente la memoria.

Miguel Pérez, otro participante de este engranaje, me acompañó a congresos, y estuvo dándome soporte de mil maneras en el proceso, gracias.

Amigos, sí: José, David, Nora, Raquel, que han vivido conmigo este camino, algunas veces me han tenido que prestar el hombro para llorar, otras veces han pasado jornadas conmigo haciendo que las cosas funcionaran. Cada apoyo, guía, tiempo y oraciones han calado en mí, ya lo saben pero les repito: los quiero mucho.

El recién incorporado al equipo de apoyo, y no menos importante, Mau, sus palabras de aliento, su compañía y sus oraciones, también han tenido efecto positivo en mí.

Quizá no soy muy elocuente, pero al mencionarlos aquí, es mi manera de que se evidencie que han impactado positivamente en mi vida y sobre todo en este logro, le agradezco a Dios por darme personas tan especiales en mi vida. Sí, soy una mujer bendecida.

Dios gracias por permitirme llegar hasta aquí,
gracias por tu gran amor.

San José, CR

Yenory Rojas Hernández

Índice General

| | |
|--|-----------|
| 1. Introducción | 1 |
| 1.1. Recuperación de información (Information Retrieval, IR) | 2 |
| 1.2. Búsqueda de respuestas (Question Answering, QA) | 4 |
| 1.3. Bases de datos relacionales (BD) | 5 |
| 1.4. Almacenes de Datos (Data Warehouse, DW) | 6 |
| 1.5. Interfaces de lenguaje natural (Natural Language Interfaces, NLI) | 6 |
| 1.6. Organización de la tesis..... | 9 |
| 2. Antecedentes | 11 |
| 2.1. Interfaces que utilizan Lenguaje Natural | 12 |
| 2.2. Preguntas contextuales..... | 41 |
| 2.3. Resolución de la anáfora..... | 50 |
| 2.4. Resolución de la elipsis..... | 54 |
| 3. Propuesta..... | 57 |
| 3.1. Arquitectura propuesta | 57 |
| 3.1.1 Conexión e interacción con la fuente de datos | 59 |
| 3.1.2 Recepción de la consulta | 69 |
| 3.1.3 Tratamiento de la consulta | 70 |
| 3.1.4 Recuperación y presentación de resultados | 71 |
| 3.2. Detalle del interfaz gráfico propuesto..... | 72 |
| 3.2.1 Fase de configuración | 72 |

vi Índice General

| | | |
|-----------|---|------------|
| 3.2.2 | Solución integrada para el tratamiento de la anáfora, elipsis y preguntas contextuales..... | 73 |
| 3.2.3 | Detalle de algoritmos propuestos | 75 |
| | Algoritmo 1: Algoritmo integrado para manejar: la anáfora, la elipsis y preguntas contextuales..... | 75 |
| | Algoritmo 2: Algoritmo para la resolución de la anáfora | 83 |
| | Algoritmo 3: Algoritmo para la resolución de la elipsis. | 93 |
| 3.3. | Caso de estudio | 98 |
| 3.3.1 | Descripción del caso de estudio..... | 99 |
| 3.3.2 | Generación de la ontología..... | 104 |
| 3.3.3 | Serie de preguntas contextuales | 109 |
| 3.3.4 | Cuadro de mando presentado al usuario..... | 121 |
| 4. | Conclusiones y trabajos futuros | 125 |
| 4.1. | Trabajos futuros..... | 128 |
| 5. | Referencias | 129 |
| 6. | Anexos | 139 |
| 6.1. | Anexo 1 - Consultas propuestas por expertos..... | 139 |
| 6.2. | Anexo 2- Trazas de ejecución de las consultas..... | 146 |
| 6.2.1 | Salida del análisis morfológico de las series de la a.1 a la r.3..... | 146 |
| 6.2.2 | Salida del análisis sintáctico de las series de la a.1 a la j.2 | 151 |
| 6.2.3 | Salida del sistema de resolución de la anáfora ... | 189 |

Índice de tablas

| | |
|---|-----|
| Tabla 2-1: Bases de datos que componen la base de conocimiento bilógico de BioBIKE..... | 33 |
| Tabla 2-2: Cuadro comparativo de los sistemas de ILN analizados | 40 |
| Tabla 3-1: Descripción de las fuentes de datos que se tiene acceso a través de la interfaz propuesta | 62 |
| Tabla 3-2: Parámetros de entrada del algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales..... | 79 |
| Tabla 3-3: Variable local del algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales..... | 80 |
| Tabla 3-4: Variable de salida del algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales..... | 80 |
| Tabla 3-5: Estructuras utilizadas por el algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales..... | 81 |
| Tabla 3-6: Variable local del algoritmo para la resolución de la anáfora..... | 89 |
| Tabla 3-7: Estructuras utilizadas por el algoritmo de resolución de la anáfora..... | 91 |
| Tabla 3-8: Resumen de las relaciones para creación de la ontología..... | 108 |
| Tabla 3-9: Medidas utilizadas para evaluar el proceso de reformulación de la consulta en el sistema propuesto | 121 |

Índice de figuras

| | |
|--|----|
| Figura 2-1: Arquitectura del Sistema PRECISE, Popescu et al. (2003) | 15 |
| Figura 2-2: Arquitectura del Sistema propuesto por Stratica et al. (2005) | 17 |
| Figura 2-3: Patrones semánticos definidos por el Sistema, Stratica et al. (2005) | 18 |
| Figura 2-4: Ejemplo del uso del patrón semántico “Attribute of object”, Stratica et al. (2005) | 18 |
| Figura 2-5: Ejemplo del uso del patrón semántico “Attribute of object of object”, Stratica et al. (2005) | 18 |
| Figura 2-6: Arquitectura del sistema ORAKEL, Cimiano et al. (2008) | 25 |
| Figura 2-7: Ejemplo de traducción de LN a SQL del Sistema de Afonso et al, 2008 | 26 |
| Figura 2-8: Ejemplo de tratamiento de estructuras complejas del Sistema de Afonso et al, 2008 | 27 |
| Figura 2-9: Tratamiento de la consulta realizado por el sistema propuesto por Jamil (2012). | 29 |
| Figura 2-10: Estructura del sistema presentado por Chaudhari, P. P. et al, 2013 | 31 |
| Figura 2-11: Interfaz gráfica de programación de BioBIKE, de Elhai et al. (2009) | 34 |
| Figura 2-12: Ejemplo de la salida de APP para una consulta en inglés Distelhorst et al. (2003) | 36 |
| Figura 2-13: Ejemplo de la división de una consulta ejecutada por GAPP para una consulta en inglés Distelhorst et al. (2003) | 37 |
| Figura 2-14: Ejemplo de preguntas anidadas, Distelhorst et al. (2003) | 38 |

| | |
|---|----|
| Figura 2-15: Arquitectura del sistema CSIR (Goldsmith et al., 2008) | 39 |
| Figura 2-16: Gráfico resumen de la utilización de herramientas y técnicas de LN en los sistemas de ILN analizados | 41 |
| Figura 2-17: Ejemplo de series de consultas de prueba para sistemas que aborden la búsqueda en preguntas contextuales, Vorhees (2001)..... | 42 |
| Figura 2-18: Ejemplo de la división de una consulta ejecutada en NITCR 2004, como parte de la subtarea 3, Kato et al. (2004). 46 | |
| Figura 3-1: Módulos principales y comunes entre Interfaces gráficos que utilizan PLN para acceso a bases de datos..... | 59 |
| Figura 3-2: Ejemplo de modelo de integración de QA para la respuesta de la pregunta: “¿Cuál es el precio de los productos Canon durante la temporada de rebajas?” | 64 |
| Figura 3-3: Extracto de la ontología de QA..... | 65 |
| Figura 3-4: Extracto de la ontología de DW..... | 66 |
| Figura 3-5: Mapeo entre ontologías QA y DW | 68 |
| Figura 3-6: Enriquecimiento de la ontología con <i>WordNet</i> | 68 |
| Figura 3-7: Resumen de los procesos básicos para la conexión a diferentes fuentes de datos | 69 |
| Figura 3-8: Tratamiento de la nueva consulta del usuario..... | 71 |
| Figura 3-9: Ejemplo de estructura <i>parsedSentence</i> | 76 |
| Figura 3-10: Ejemplo de extensión semántica de los conceptos en <i>NLQ_concepts</i> | 78 |
| Figura 3-11: Ejemplo de la representación de la no coincidencia de un concepto en <i>NLQ_concepts</i> | 79 |
| Figura 3-12: Diagrama del algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales..... | 82 |
| Figura 3-13: Código del algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales..... | 83 |
| Figura 3-14: Ejemplo de búsqueda de la anáfora en <i>NLQ_concepts</i> | 84 |

| | |
|---|----|
| Figura 3-15: Ejemplo de reglas de restricción y preferencia aplicadas en la resolución de la anáfora. | 85 |
| Figura 3-16: Ejemplo de pregunta con referencia anafórica. | 86 |
| Figura 3-17: Ejemplo de <i>NLQ_concepts</i> para: “ <i>Which traits are affected by them?</i> ”. | 86 |
| Figura 3-18: Ejemplo de estructuras de huecos almacenadas en <i>previous_questions</i> y <i>previous_answers</i> para las preguntas: “ <i>What QTLs are related to frost tolerance in wheat? In barley?</i> ” 87 | 87 |
| Figura 3-19: Preguntas y sus respectivas estructuras de huecos en <i>NLQ_concepts</i> para la reformulación de la consulta, una vez resuelta la referencia anafórica: “ <i>Which traits are affected by them?</i> ” 90 | 90 |
| Figura 3-20: Captura de pantalla de la pregunta contextual: “ <i>Which traits are affected by them?</i> ”, que sigue a las preguntas: “ <i>What QTLs are related to frost tolerance in wheat?</i> ”, “ <i>In barley?</i> ” 90 | 90 |
| Figura 3-21: Código del algoritmo para la resolución de la anáfora..... 92 | 92 |
| Figura 3-22: Diagrama del algoritmo para resolución de la anáfora..... 93 | 93 |
| Figura 3-23: Ejemplo de patrón sintáctico-semántico utilizado para validar si una pregunta está bien formada. Ejemplo de pregunta bien formada y pregunta con elipsis..... 94 | 94 |
| Figura 3-24: Ejemplo de estructuras de huecos guardadas en <i>NLQ_concepts</i> para segunda frase de la consulta..... 94 | 94 |
| Figura 3-25: Estructuras generadas tras la consulta de <i>previous_questions</i> para la pregunta “ <i>In barley?</i> ” 95 | 95 |
| Figura 3-26: Ejemplo de estructuras guardadas en <i>NLQ_concepts</i> para la segunda frase de la consulta tras la resolución de la elipsis. 96 | 96 |

| | |
|--|-----|
| Figura 3-27: Captura de pantalla de la pregunta contextual “ <i>In barley?</i> ”, que sigue a la pregunta: “ <i>What QTLs are related to frost tolerance in wheat?</i> ” | 97 |
| Figura 3-28: Código del algoritmo para manejar la resolución de la elipsis. | 98 |
| Figura 3-29: Extracto del modelo multidimensional para el escenario CEREALAB | 102 |
| Figura 3-30: Ontología correspondiente a la base de datos CEREAL LAB..... | 104 |
| Figura 3-31: Serie de preguntas propuestas para la aplicación de la propuesta. | 110 |
| Figura 3-32: Contenido de la estructura <i>parsed_sentence</i> luego de la aplicación del análisis sintáctico parcial a la consulta a).1. | 111 |
| Figura 3-33: Cuadro de mando para la serie de preguntas a), de la sección anterior..... | 122 |
| Figura 3-34: Cuadro de mando para la serie de preguntas b), de la sección anterior..... | 123 |

1. Introducción

Desde los inicios del 2000 se consideraba la información como un activo de las organizaciones (Oppenheim et al., 2003). En el 2011 un estudio del Grupo Gartner reportaba que el volumen de la información crecía a una tasa mínima del 59% anualmente. Y según el Estudio del Universo Digital de la EMC en el 2014 indicaba que para el 2020 el universo digital crecerá a 44 billones de gigabytes.

Además del crecimiento mencionado, este escenario implica heterogeneidad de fuentes y métodos de almacenamiento. La información se puede encontrar almacenada en bases de datos relacionales, base de datos textuales, o cualquier otro tipo de repositorio de información dependiendo del negocio y de las herramientas utilizadas para acceder a ella. Aunado a esto, también hay diversidad de lugares donde se puede alojar, por ejemplo físicamente en la misma organización o en la nube. Otro factor a considerar es el dispositivo de dónde se accederá, o que generará información. Ya no solo se genera información por personas u organizaciones, “las cosas” conectadas a través de Internet, también contribuyen a esta tarea.

Siendo la información un activo clave, aunado al crecimiento exponencial que muestra y la diversidad y heterogeneidad de su representación, se presenta un reto respecto al acceso de la misma: ¿cómo acceder a la información de manera oportuna, estando ésta almacenada en fuentes de datos diversas y heterogéneas?

Este reto planteado, genera una serie de dimensiones a las cuales atender. Hay que tener en cuenta el formato de la fuente de datos,

que a su vez acarrea un requisito sobre el nivel de conocimiento técnico del usuario (por ejemplo, si se va a acceder a una base de datos relacional, es necesario el conocimiento de la estructura y del lenguaje de consulta para la misma). En cambio si lo que se busca el usuario es información general, puede buscarse la almacenada en Internet, a través de un motor de búsqueda; si tuviese un nivel de especialización adecuado, en alguna base de datos específica; etc.

Para resolver el acceso a la información se han propuesto diferentes métodos, cada uno con características distintas y atendiendo diferentes aristas del problema, algunas propuestas más simples, otras más complejas, que además restringen el tipo de usuario e información al que se puede acceder. De entre todas las propuestas, destacamos las siguientes:

1.1. Recuperación de información (Information Retrieval, IR)

La IR es una disciplina relativamente joven. Sus inicios datan de los años 50, y sus raíces se sitúan en el uso de los catálogos definidos por los bibliotecólogos (Salton, 1968), donde a cada documento publicado se le asigna una categoría, de modo que el acceso a la información sea de manera expedita. A pesar de su corta vida, esta disciplina ha sufrido un crecimiento exponencial, evidenciado por el impacto que tiene en los quehaceres de la sociedad actual, por medio de la tarea que desempeñan los motores de búsqueda en Internet, los cuales permiten acceder a la gran cantidad de información que se encuentra en la Web, de manera ágil.

El término Recuperación de Información fue introducido formalmente en 1950 por Calvin N. Mooers, quien lo definió como: *“La búsqueda de información en un conjunto de documentos, realizada a partir de la especificación de un tema”*.

El objetivo primordial de la Recuperación de Información es brindar de la manera más precisa la información que el usuario ha solicitado, en el menor tiempo posible. En otras palabras, un sistema de IR apoya al usuario en la búsqueda de los documentos relacionados con su requerimiento de información.

Para lograr este objetivo se han realizado gran cantidad de propuestas. Por ejemplo, las que trabajan con Procesamiento de Lenguaje Natural como las aproximaciones que han sacado provecho del uso de información morfológica para usar el lema (forma canónica de la palabra) en vez del “stem” (raíz de la palabra) de las palabras u otras que utilizan diversas derivaciones morfológicas (Vilares et al., 2003). Otros sistemas usan técnicas de expansión de las preguntas sumando nuevos términos a la pregunta original obtenidos a partir de recursos léxicos, como WordNet, (Gonzalo et al., 1988) o (Arampatzis et al., 2000). También hay aproximaciones que utilizan WordNet, pero en su etapa de indexación (Rojas, et al., 2005a; 2005b). Se encuentran propuestas que indexan pares de palabras ((Zhai et al., 1997), (Mitra et al., 1997), (Strzalkowski et al., 1999)) o sintagmas (Arampatzis et al., 2000). Por otro lado, se pueden ver aproximaciones que han ido un nivel más adelante, sacando provecho de la información léxica y sintáctica, por medio del uso de los sintagmas nominales, preposicionales y verbales, unido con la resolución de problemas lingüísticos, como la anáfora (Ferrández et al., 2007, Rojas, et al., 2005c, Ferrández et. al, 2015).

1.2. *Búsqueda de respuestas (Question Answering, QA)*

Los sistemas de búsqueda de respuestas tienen como objetivo obtener el texto específico que responde a una pregunta expresada en lenguaje natural de manera precisa. Estos sistemas analizan y extraen las respuestas de aquellas zonas de los documentos de cuyo contenido es posible inferir la información. Estos sistemas debido a su complejidad, necesitan hacer uso de técnicas de Procesamiento de Lenguaje Natural.

Las primeras propuestas para sistemas de QA se desarrollaron en los años 60, y su estructura principal constituía un sistema experto con una interface en lenguaje natural que se aplicaba a un dominio restringido.

Las propuestas iniciales más relevantes fueron: BASEBALL y LUNAR. BASEBALL fue propuesto por Green et al., en 1961, y su dominio era la liga de béisbol de USA en el periodo de un año. LUNAR (Woods et al., 1972), por su parte, respondía a preguntas sobre el análisis geológico de las piedras lunares que trajeron las misiones Apollo. Los siguientes sistemas mantenían las características básicas de estos primeros: su núcleo era siempre una base de datos de conocimiento, definida manualmente por expertos del dominio.

Ejemplos de aplicaciones prácticas donde se pueden aplicar los sistemas de QA son:

- Sistemas de ayuda online.
- Sistemas de consulta de datos para empresas.
- Interfaces de consulta de manuales técnicos.

- Sistemas de búsqueda de respuestas generales de acceso público sobre Internet.

Algunas aproximaciones relevantes (Harabagiu et al., “FALCON: Boosting Knowledge for Answer Engines”, 2000), (Harabagiu et al., “Answering complex, list and context questions with LCC's Question-Answering Server” 2001) y (Soubbotin & Soubbotin. “Patterns of Potential Answer Expressions as Clues to the Right Answers”, 2001) o las desarrolladas en la Universidad de Alicante (Muñoz-Terol et al, 2008) (Vila et al., 2011), entre otras.

Los componentes principales de un sistema de QA son:

- Análisis de la pregunta.
- Recuperación de documentos o pasajes relevantes.
- Extracción de respuestas.

1.3. *Bases de datos relacionales (BD)*

Los sistemas de Bases de Datos están optimizados para el manejo de datos estructurados con una semántica bien definida, al contrario de los sistemas de IR. En estos sistemas el usuario introduce una consulta específica, expresada en álgebra relacional, obteniendo como salida, en forma tabular, los resultados que satisfacen dicho requerimiento sin posibilidad alguna de error.

Para utilizar este tipo de sistemas es necesario conocimiento técnico experto. Se debe tener acceso a la estructura de la BD y conocer la sintaxis que utiliza el sistema gestor de la misma.

Estos sistemas se han usado de forma comercial o por organizaciones que utilizan transacciones específicas

estandarizadas como bancos, empresas de todo tipo, organismos públicos, etc.

1.4. *Almacenes de Datos (Data Warehouse, DW)*

Los Almacenes de Datos son repositorios diseñados para facilitar la confección de los informes y la realización de análisis en las organizaciones. Se forman a partir de BD existentes y son de alto apoyo a la toma de decisiones de las empresas.

Al igual que las BD requiere de conocimiento técnico experto, ya que debe modelar soluciones a partir de BD existentes y generar una optimización y cruce de información, todo expresado con una sintaxis específica.

1.5. *Interfaces de lenguaje natural (Natural Language Interfaces, NLI)*

Como se mencionó al inicio de este capítulo, con la cantidad creciente de información en la actualidad, es necesario disponer de herramientas que permitan el acceso a bases de datos/repositorios de información híbridos y heterogéneos, sin que esto implique conocimiento previo específico por parte del usuario. Las NLI al contrario de las soluciones para acceso a información nombradas anteriormente, tratan de abordar esta problemática.

Un ejemplo de la necesidad de estas soluciones se da en el área de las investigaciones científicas, concretamente en el área de Ciencias de la Vida, donde se almacena la información en bases de datos especializadas (DataBases, DB), como las que abordan los temas de Genética y Biotecnología (Matos et al., 2010). Esa gran cantidad de DBs requiere que se optimicen las estrategias de

búsqueda de modo que el usuario pueda acceder a la información almacenada de manera agradable y eficiente (Jensen, Saric, & Bork, 2006; Altman et al., 2008). En este sentido, hay una necesidad de diseñar interfaces sencillas, que trabajen por ejemplo con conceptos de Biología Molecular complejos, para que faciliten a los biólogos la comprensión de los datos.

Si se requiere de formación técnica o se utilizan métodos poco intuitivos para la consulta de los datos, esto genera que el usuario se intimide y, por lo tanto, no saque provecho a la información almacenada. Todo ello ha provocado que las aplicaciones que disponen de un entorno gráfico o que utilizan métodos de búsqueda simple (como palabras clave) se hayan vuelto muy populares, aunque presentan limitaciones en los resultados obtenidos (Li et al., 2007).

Las búsquedas en lenguaje natural (LN) probablemente constituyen la manera más flexible y efectiva de examinar una base de datos biológica (Jamil, 2012). Dado que muchas de las DBs del área de Biología (por ejemplo, GenBank o Genome Browser) emplean interfaces gráficas y no están habilitadas para consultas arbitrarias, se han realizado esfuerzos para construir estrategias para acceder por medio de LN a DBs existentes. Podemos encontrar ejemplos de este tipo en los campos de Biomedicina y Biotecnología (e.g. Jamil, 2012; Goldsmith et al., 2009; Clegg & Shepherd, 2007; Distelhorst et al., 2003). Sin embargo, Cimiano et al. (2009) indican que aunque el paradigma basado en consultas de LN es considerado como el más intuitivo desde un punto de vista de uso, se ha demostrado que es el más difícil de realizar con eficacia. Las principales razones de esta dificultad son las siguientes:

- El LN es, sin duda, una tarea muy difícil debido a las ambigüedades que surgen en todos los niveles del análisis: morfológico, léxico, sintáctico, semántico y pragmático.
- Se requiere de una gramática amplia para que el sistema tenga una cobertura aceptable.
- El sistema debería ser adaptable a diferentes dominios sin un esfuerzo significativo.
- El LN debe ser preciso.

El campo de la Ingeniería Genética es una disciplina emergente, y se ha expandido a la biomedicina, agricultura y otros dominios relacionados (Aleksejeva, 2014). En esta Tesis, el caso de estudio que veremos en el Capítulo 3 está centrado en una empresa fitomejoradora (mejoradora de plantas), para la cual su principal objetivo es crear organismos genéticamente modificados (Genetically Modified Organisms, GMOs). De acuerdo con la Organización Mundial de la Salud, los GMOs son organismos a los que se les ha alterado su material genético (ADN), de manera que esa alteración no puede ser generada naturalmente (World Health Organization, 2002). Esta alteración se obtiene insertando secuencias de ADN de un organismo a otro. Esta estrategia de mejoramiento de plantas tiene un papel importante en el mercado mundial, hasta el punto que, en 2014, cinco compañías de semillas controlan el 35% del mercado mundial (Le Buanec, 2008) y el 33% de su producto son los GMOs (Rotolo et al., 2014; ISAAA, 2012; Meijerink & Danse, 2009; ETC, 2008). Es importante destacar que en el mercado de cereales, frutas, verduras y legumbres, la producción de semillas modificadas sigue creciendo.

En Biotecnología vegetal, el análisis de QTL, Quantitative Trait Locus, (*locus* -una posición fija en un cromosoma- de una carácter

cuantitativo) es muy útil, ya que permite identificar la acción, la interacción, el número y la ubicación precisa de las regiones cromosómicas que contienen uno o más genes implicados en características fenotípicas concretas (Falconer & Mackay, 1996; Kearsey, 1998; Lynch & Walsh, 1998; Miles & Wayne, 2008). De este modo, un QTL podría ser transferido en el laboratorio a partir de un organismo a otro para modificar uno o más rasgos particulares. Teniendo en cuenta que nuestro conocimiento de la función de los productos génicos está aumentando rápidamente, las bases de datos de QTLs tratan de recoger todas las relaciones entre las posiciones de los cromosomas y las características biológicas de muchos organismos.

En esta tesis presentamos una propuesta de un NLI. La misma mejora las propuestas existentes hasta el momento ya que incorpora la resolución de problemas lingüísticos diversos, como la anáfora y la elipsis. Además, es capaz de procesar una serie de preguntas encadenadas (serie de preguntas contextuales). La aproximación se aplica como ayuda en un Sistema de Toma de Decisiones (Decision Support System, DSS) en una empresa de Biotecnología. Se realiza una evaluación que prueba la efectividad de la aproximación, con una fortaleza que debe resaltarse ya que se conectan diversas bases de datos heterogéneas (información estructurada y datos no estructurados externos obtenidos de la web), sin que el usuario requiera conocimiento técnico experto.

1.6. Organización de la tesis

El documento está estructurado de la siguiente manera. En el capítulo 2, se resumen los trabajos más relevantes del área, se hace una descripción de ellos y se resumen sus principales

características. En el capítulo 3, presentamos la propuesta, que es una herramienta de consulta que utiliza la resolución de la anáfora y elipsis, apoyado por preguntas de contexto. Además, con el objetivo de aclarar nuestra propuesta, se expone la aplicación sobre el estudio de caso en el que se consulta la base de datos de dominio bilógico CEREALAB. Se concluye el documento con el resumen de nuestra contribución y nuestro trabajo futuro en el capítulo 4.



Universitat d'Alacant
Universidad de Alicante

2. Antecedentes

Los sistemas de apoyo a las decisiones integran una variedad de interfaces para la consulta de bases de datos o almacenes de datos. Hasta el momento, estas interfaces se han implementado principalmente como sistemas gráficos, que permiten al usuario de una manera más fácil y dinámica formular las necesidades de información requeridas. Sin embargo, estas interfaces gráficas presentan ciertas desventajas, como son la falta de flexibilidad a la hora de la formulación de consulta (ya que ésta se debe ajustar a la interfaz), sus limitaciones sobre el manejo de las consultas, y además algunas requieren que el usuario utilice el lenguaje propio de la herramienta, lo que implica un esfuerzo adicional y muchas veces representa una barrera de acceso.

Ante estas desventajas, se valora como alternativa con alto potencial el uso de interfaces que utilicen el lenguaje natural (LN), ya que permiten mayor flexibilidad para que el usuario formule sus necesidades de información sin necesidad de ajustarse a formatos preestablecidos. Sin embargo, estos sistemas presentan las desventajas propias del LN, como son la dificultad y el costo computacional del manejo del LN; la cobertura lingüística; y la representación real del requerimiento del usuario sin restar fluidez o naturalidad a la comunicación (como el trato adecuado del contexto y la utilización de fenómenos del discurso como anáfora y elipsis).

Las siguientes cuatro subsecciones abordan las diferentes propuestas/aproximaciones que se han presentado al respecto y la manera en que se trata de dar solución a los diferentes retos que se presentan para el correcto uso del LN: interfaces de LN,

preguntas contextuales, resolución de la anáfora y resolución de la elipsis.

2.1. *Interfaces que utilizan Lenguaje Natural*

El objetivo de esta subsección es analizar las propuestas de interfaces que utilizan lenguaje natural, y que de alguna manera hacen uso de la resolución de la anáfora y elipsis, o que utilizan preguntas contextuales.

Como ejemplo de estas propuestas está la presentada por Popescu et al. (2003), donde se describe el sistema PRECISE, que traduce de LN a SQL. En su trabajo tratan preguntas clasificadas como semánticamente tratables, que son aquéllas que siguen un patrón definido y no dan espacio para ambigüedades, que además no requieren el tratamiento de algún fenómeno del discurso, como la elipsis o anáfora.

La Figura 2-1 muestra la arquitectura de PRECISE:

- Analizador (*Parser*)
PRECISE utiliza como analizador el sistema Charniak y genera para cada pregunta un árbol que representa sus relaciones sintácticas, que serán utilizadas en los siguientes módulos.
- Segmentador (*Tokenizer*)
El segmentador divide la consulta teniendo en cuenta que debe ser la información necesaria para acceder a la base de datos y que son concretamente 3 elementos: relaciones, atributos y valores.

- Relación: Es la correspondencia entre el atributo y el valor y nacen del auxiliar de la pregunta {"qué", "dónde", "quién", "cuándo"}.
 - Atributo: Es una columna de la BD en una relación en particular. Un atributo es compatible con su relación.
 - Valor: Es el valor de un atributo. Un valor es compatible con su atributo y también con la relación que contiene dicho atributo.
- Lexicon
Recibe los tokens y recupera los elementos que coinciden con la base de datos.
 - Emparejador (*Matcher*)
Éste es el elemento diferenciador de su propuesta. Se reduce a un problema de emparejamiento de grafos: el problema de encontrar la interpretación semántica de los *tokens* que consideren ambiguos a su correspondiente en la base de datos.
 - Comprobación de equivalencias (*Equivalence Checker*)
Se valida si hay varias soluciones distintas al problema y si estas soluciones se traducen en consultas SQL distintas.

Si PRECISE encuentra 2 consultas SQL distintas, no emite una respuesta, ya que no puede tener la certeza de que cuál es la consulta correcta. Ante esto pide al usuario elegir entre 2 o más interpretaciones. Un ejemplo de esto puede observarse en la pregunta:

"¿Cuáles son los puestos de trabajo de analista de sistemas en Austin?"

"Analista de sistemas" podría referirse al puesto de *analista de sistemas*. Por otra parte, si se analiza por separado, podría tomar *sistemas*, como los sistemas que utilizan los diferentes equipos y *analista* como al puesto de analista.

Ante esto, el sistema envía al usuario sugerencias para que el usuario decida cuál es la interpretación correcta.

- Generador de consultas (*Query Generator*)
Toma los elementos generados por el módulo anterior y los convierte a SQL.

En la evaluación, su sistema identificó que no le era posible tratar el 20% de las consultas, ya que no eran semánticamente tratables, el 80% restante las respondió adecuadamente. Para ese 20% identificado como no tratable, se solicitaba que se reformularan las consultas.

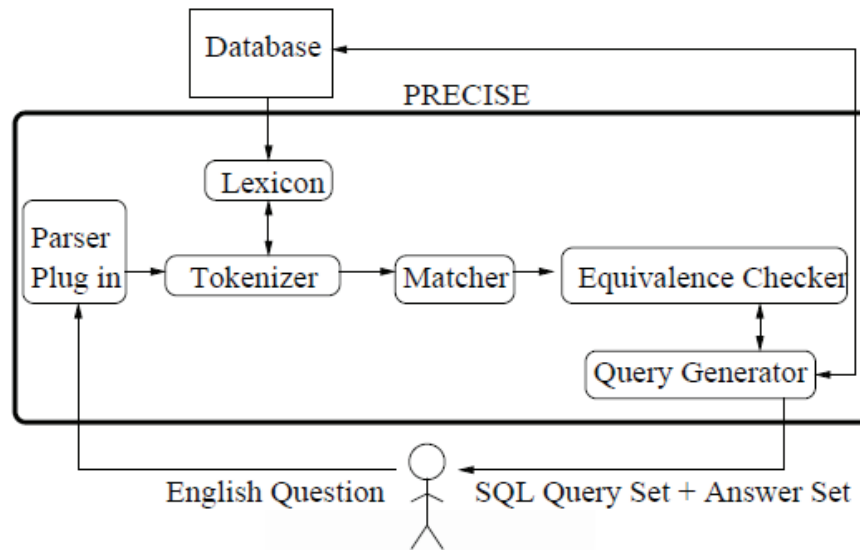


Figura 2-1: Arquitectura del Sistema PRECISE, Popescu et al. (2003)

Otra propuesta que aborda el mismo objetivo, realizar la traducción de LN a SQL de manera transparente al usuario, fue presentada por Stratica et al. (2005). Su sistema realiza el proceso de traducción utilizando análisis sintáctico y semántico (además se apoyan en *WordNet*). El sistema requiere de la creación de reglas propias del dominio en que se está trabajando.

En la Figura 2-2 se puede observar con detalle la arquitectura de la propuesta:

- **Análisis sintáctico**
Se realiza por medio de la herramienta de código abierto Link Parser, que devuelve por cada consulta un conjunto de todos los posibles árboles de análisis, clasificados en orden de probabilidad. Esto les permite crear varias consultas posibles, tomando en cuenta el nivel de confianza indicado por la herramienta.

- Análisis semántico
Utilizando como base los árboles generados en el punto anterior se compara cada uno de ellos con 3 plantillas definidas por ellos. De este modo se tiene en cuenta las posibles opciones que podrían surgir ocasionadas por la variación léxica de una misma consulta. Como ejemplo se tendría la misma salida en SQL para una consulta realizada de varias maneras:
 - ¿Quién es el autor del libro “Algoritmos”?
 - ¿Quién es el escritor del libro “Algoritmos”?
 - ¿Quién es el autor del recurso “Algoritmos”?
 - ¿Quién es el escritor del recurso “Algoritmos”?

Para construir la base de conocimiento semántico, el pre-procesador primero lee el esquema de la base de datos, identifica todos los nombres de las relaciones y nombres de los atributos y, por último, utiliza *WordNet* para crear una lista de hipónimos, hiperónimos y sinónimos para cada relación y el nombre de atributo. Como paso siguiente, el administrador del sistema puede modificarlo para reflejar los sentidos específicos en el contexto del dominio de la base de datos. El conjunto semántico ajustado se utiliza en tiempo de ejecución para encontrar qué nombres de relación tienen más probabilidades de ser los que se encuentran en la consulta dada.

En la Figura 2-3 aparecen patrones genéricos definidos por los autores y ejemplos de uso de los 2 primeros patrones pueden verse en las Figura 2-4 y 2-5.

Para el primer patrón se identifica claramente en la Figura 2-4 que el atributo “*address*” es el atributo del objeto “*author Mark Twain*”. El segundo patrón se visualiza en la Figura 2-5; en la segunda consulta, el atributo es “*name*” del objeto “*author*” y del objeto “*book Physics*”

- Generación de la consulta
Una vez identificados atributos, objetos y relaciones, se genera la consulta en SQL y se obtiene su respuesta de la base de datos.

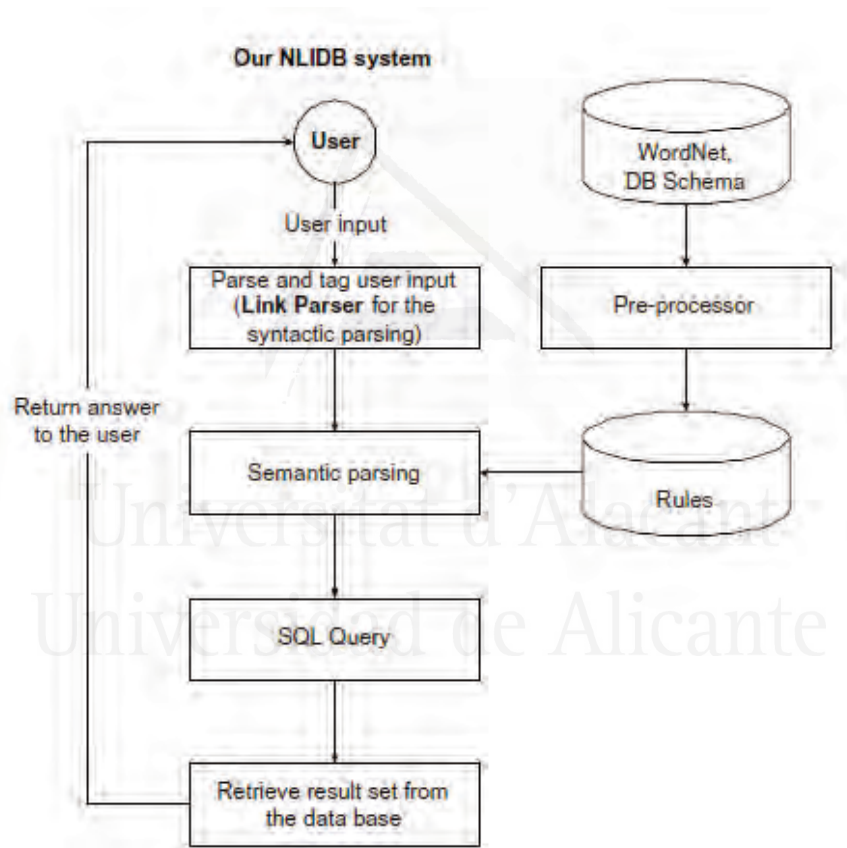


Figura 2-2: Arquitectura del Sistema propuesto por Stratica et al. (2005)

| |
|---|
| <p><attribute> of <object> <attribute> of <object> of <object> <action verb> <object> <attribute value></p> |
|---|

Figura 2-3: Patrones semánticos definidos por el Sistema, Stratica et al. (2005)

| Consulta en Leneuaie natural | Consulta en SQL |
|--|--|
| <p>List the address of the writer Mark Twain <i>Attribute Relation</i></p> | <pre>SELECT address FROM author WHERE name = Mark Twain</pre> |
| <p>What is the size of the book "Algorithms"? <i>Attribute Relation</i></p> | <pre>SELECT size FROM resources WHERE title = 'Algorithms'</pre> |

Figura 2-4: Ejemplo del uso del patrón semántico "Attribute of object", Stratica et al. (2005)

| Consulta en Lenguaje Natural | Consulta en SQL |
|---|--|
| <p>List the address of the author of the book 'Astronomy'. <i>Attribute Relation1 Relation2 Value of Default Attribute</i></p> | <pre>SELECT author.address FROM author, resource, resource_author WHERE resource.title = 'Astronomy' AND resource.resource_id = resource_author.resource_id AND author.author_id = resource_author.author_id</pre> |
| <p>What is the name of the author of the book 'Physics'? <i>Attribute Relation1 Relation2 Value of Default Attribute</i></p> | <pre>SELECT author.name FROM author, resource, resource_author WHERE resource.title = 'Physics' AND resource.resource_id = resource_author.resource_id AND author.author_id = resource_author.author_id</pre> |

Figura 2-5: Ejemplo del uso del patrón semántico "Attribute of object of object", Stratica et al. (2005)

La evaluación de este sistema fue realizada sobre una base de datos de una biblioteca, con preguntas de prueba generadas por el equipo, no con usuarios reales. En su mayoría, las repuestas generadas fueron correctas. La propuesta no soporta preguntas de contexto, ni tratan la resolución de la anáfora o elipsis. Como ejemplo, no pudieron dar respuesta a una pregunta en apariencia

simple como es: “*Show the address of Mark Twain*”, pero sí lo hacían si se especificaba una característica del objeto “*Show the address of **author** Mark Twain*”.

Dentro de los proyectos que tratan la resolución de la anáfora se puede citar el trabajo realizado por Li et al. (2005), donde presentan NaLIX, un interfaz de lenguaje natural de consulta genérica e interactiva, que realiza consultas a una base de datos XML. Este sistema está dividido en 2 tareas específicas:

- Transformación de la pregunta de LN a XML

Esta tarea se centra en obtener la correcta representación de la consulta en LN de modo que al hacer su transformación a XML se logre obtener la respuesta adecuada.

Esta transformación se realiza en 3 pasos:

- Clasificador

Con base en el árbol de análisis generado a partir de la consulta original, se identifica qué palabra o frase puede ser mapeada a la consulta en XML.

- Validador

Utilizando el árbol del paso anterior, comprueba que sea posible mapearlo a XML. También comprueba si es posible encontrar en la base de datos, los nombres de los elementos, atributos o valores contenidos en la consulta del usuario.

- Traductor

Al llegar a este paso, se asume que el árbol de análisis ya está bien formado para hacer el paso a la consulta en XML, según la estructura definida en la base de datos.

Es importante mencionar que en el proceso de traducción el sistema toma en cuenta la negación simple, los predicados de comparación, la cuantificación, el anidamiento, agregación, unión de valores, y la clasificación.

- Formulación de la consulta de manera interactiva

Para obtener una consulta que pueda ser transformada adecuadamente el sistema guía al usuario en un proceso interactivo para que vaya formulando la pregunta de manera que sea entendible por el sistema. Esto se realiza tomando como base la pregunta original y se envía al usuario sugerencias de la pregunta, formuladas de manera que se adecúe al sistema. En general, se toman como máximo 2 iteraciones para tener la consulta en lenguaje natural y que sea posible de interpretar por el sistema.

- Repositorio de consultas

Almacena automáticamente las consultas en lenguaje natural que han sido validadas exitosamente durante una sesión de búsqueda. El usuario puede acceder y gestionar el historial de consultas en cualquier momento desde la interfaz gráfica. También tiene la opción de guardar todo el historial de consultas, o una parte seleccionada de ella en un fichero.

- Generador de mensajes

Si la validación del árbol del análisis falla, el generador de mensajes emitirá mensajes de error y sugerencias basadas en la información proporcionada por el validador del árbol de análisis sintáctico.

El sistema también emite advertencias en algunas situaciones, junto con los resultados de la consulta, para que el usuario sea consciente de los problemas que pueden tener un impacto negativo en la traducción de la consulta. Por ejemplo, si existe un pronombre en una consulta, las advertencias se emitirán para recordar al usuario un posible problema en la resolución de la anáfora que podría incidir en la calidad de la búsqueda. Por otra parte, si existen varias coincidencias en la base de datos durante la comprobación de la existencia de un nombre de elemento, atributo o valor, se le da al usuario la oportunidad de elegir el que corresponda.

- Resultado de las consultas

NaLIX despliega los resultados según las necesidades de los usuarios:

- Resultados simples: Salida sencilla en formato texto.
- Resultados complejos: Vista de árbol, de modo que el usuario pueda navegar sobre los resultados.
- Resultados estructurados: Vista de plantilla, que permite realizar el análisis correspondiente.

- Expansión de términos basado en una Wordnet
En la interacción con el usuario para ir depurando su consulta, se utiliza Wordnet como ontología para la expansión de términos de la consulta.

Se puede concluir que, a pesar de que en el módulo de transformación de la consulta se tienen en cuenta la anidación, cuantificación, agregación y otros, se presentan deficiencias en la resolución de la anáfora y elipsis. Además, aunque se permita almacenar el historial de consultas, éstas no son tomadas como preguntas de contexto, sino que son una ayuda para que el usuario tenga una frase bien formada para consultar la base de datos.

Por su parte, Laukaitis y Vasilecas (2007) presentan una propuesta que pretende apoyar los portales web que distribuyen información. Para ello, proponen una arquitectura de LN de diálogo. Además, investigan la arquitectura de agentes distribuidos, donde cada agente administra su código y datos de manera remota y devuelve sólo la información que es requerida. Respecto al PLN utilizan análisis morfológico, sintáctico, léxico y semántico, obtenido de combinar su arquitectura con el sistema GATE (*General Architecture for Text Engineering*).

El aporte de este proyecto se da principalmente en el área de arquitectura del software, ya que ahonda sobre el uso que puede obtenerse del lenguaje de programación Java, sus herramientas y su combinación en diversas arquitecturas. Respecto al uso de PLN no presentan un aporte sustancial, ya que se circunscriben al uso de una herramienta existente (GATE), sólo muestran la unión de ésta en la arquitectura propuesta.

Respecto al sistema GATE, fue presentado por Cunningham et al. en el año 2000. Es una infraestructura robusta y escalable para PLN. Se ha diseñado de tal manera que cada módulo se pueda utilizar independiente de manera simple. Además, permite ser extendido y personalizado, según las necesidades del usuario. Este sistema maneja 3 tipos de componentes:

- Recursos de lenguaje
 - Lexicones
 - Corpus
 - Ontologías
- Recursos de procesamiento
 - Analizadores léxicos
 - Generadores de n-gramas
- Recursos visuales
 - Componentes de edición y visualización

Esta arquitectura es bastante robusta y proporciona un gran apoyo para no iniciar desde cero si se desea utilizar PLN en algún tipo de proyecto.

Con referencia a las interfaces de LN que hacen uso de ontologías, se puede mencionar el proyecto que desarrolló Cimiano et al. (2008). Diseñaron una interfaz denominada ORAKEL cuyo objetivo principal consistió en minimizar el esfuerzo de adaptar un sistema a un dominio específico.

La arquitectura de ORAKEL se puede visualizar en la Figura 2-6. Los componentes principales son los siguientes:

- Ontología de dominio
 - Se construye manualmente por un experto y es aquí donde se adapta el sistema a un dominio específico. Su rol es

estratégico en la aplicación ya que es el núcleo de todo el proceso de adquisición de léxico.

- **Lexicones**
El lexicón general y el de dominio específico son creados manualmente por el experto del dominio. Estos proporcionan las únicas fuentes que ORAKEL requiere para responder a las preguntas. Ambos tipos de léxicos son gramáticas lexicalizadas, y son utilizadas tanto para el análisis, como para la construcción de la semántica de las preguntas de entrada. Con estos componentes el sistema no requiere el uso de recursos léxicos o gramáticas externas.
- **Usuarios.** Son de dos tipos:
 - Los usuarios que generan consultas, común al resto de sistemas.
 - “*FrameMapper*”: se refiere a un experto en el dominio o un ingeniero de conocimiento, que esté familiarizado con la base de conocimientos subyacente. Su rol principal es ser “ingeniero de léxico”, de modo que interactúa con el sistema en la adquisición de léxico y la creación de léxicos específicos de dominio para adaptar el sistema al dominio específico.
- **Intérprete de la consulta (*Query interpreter*)**
Analiza la consulta formulada por el usuario y construye una consulta en forma lógica (LF), la fórmula con respecto a los predicados de dominio específico.
- **Convertidor de la consulta (*Query converter*)**
Recibe la consulta en su forma lógica y la traduce al lenguaje y formato definido en la base de conocimiento.

Es a través de este módulo, y con el uso de la ontología, que los creadores de ORAKEL aseguran la portabilidad e independencia de idioma y de dominio.

- Generador de la respuesta (*Answer generation*)
Realiza el proceso de deducción entre la consulta y la base de conocimiento.

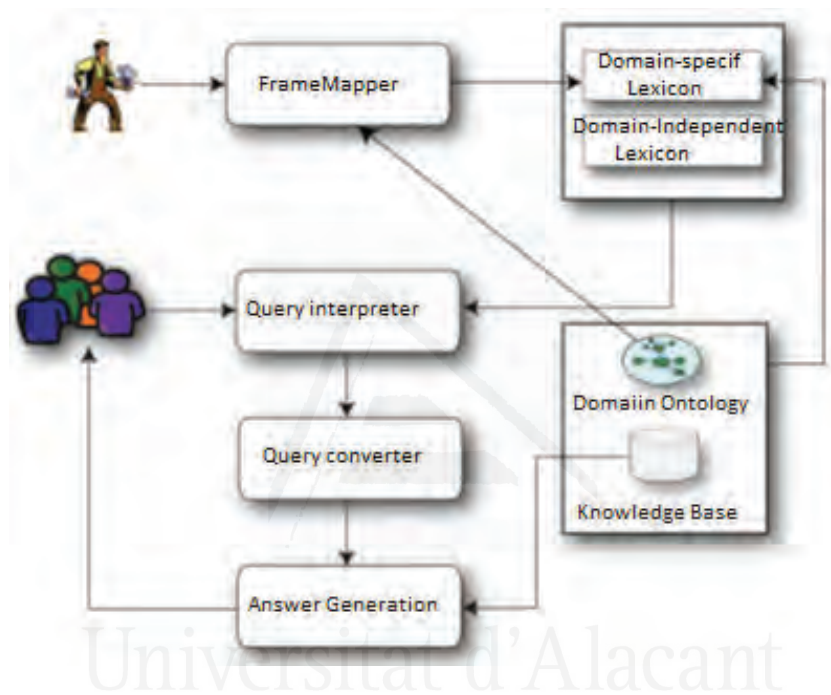


Figura 2-6: Arquitectura del sistema ORAKEL, Cimiano et al. (2008)

Aunque en el análisis que presentan los autores concluyen que no existe una alta complejidad en la creación de los lexicones y la ontología, es claro que requerir de conocimiento experto, siempre es una barrera para el uso de un sistema. Por ello, podemos indicar como principal desventaja de esta aproximación la necesidad de un experto para la generación manual del lexicón.

Dentro de las limitaciones señalan que aunque el sistema puede tratar consultas complejas, dejan pendiente de resolver cómo lidiar con: entradas no gramaticales, palabras desconocidas, resolución de la anáfora, elipsis y preguntas contextuales.

En el mismo año, 2008, Afonso et al. presentan un algoritmo bajo el paradigma de computación evolutiva que traduce de LN a SQL. Lo evaluaron sobre el idioma portugués (Brasil), e indican que sus resultados arrojan un 92% de exactitud en las interpretaciones realizadas. Su sistema combina el uso de análisis léxico y semántico. Una de las características principales de este sistema es que ellos utilizan la resolución de la anáfora y la elipsis.

En la Figura 2-7 se muestra un ejemplo de la traducción de LN a SQL de una oración. Se puede observar que en la oración se distinguen 3 componentes estructurales: Ps, Ta y R. Ps es la representación léxica de los elementos seleccionados; por su parte, Ta se refiere al objeto de la consulta, y corresponde al nombre de la tabla donde está almacenada esa información; respecto a R, define qué información se está buscando del objeto. Como es sabido, la información en LN no siempre se representa de la misma manera, por esto algunas veces los componentes deberán ser inferidos.

Frase en LN:

¿Hubo un alumno con alta calificación?

| | | |
|-----------------|---------------------|------------------------------------|
| Existe | Alumno | con alta calificación |
| SELECT | *FROM Alumno | WHERE Alumno.Califi >= 9 |
| P _{s1} | T _{a1} | R _{s1} |

Figura 2-7: Ejemplo de traducción de LN a SQL del Sistema de Afonso et al, 2008

Para estructuras complejas, los autores proponen la resolución de la anáfora y la elipsis. El algoritmo realiza primeramente una división de la oración en frases básicas. Para cada una encuentra los componentes estructurales mencionados anteriormente y se identifica la necesidad de información R_1 , de existir más de una se determina. En una pila se almacenan los nombres de las frases. Como último paso se analizan las frases con pronombres o referencias y los reemplazan por los nombres correspondientes almacenados en la pila. La Figura 2-8, muestra un ejemplo del tratamiento de las estructuras complejas.

Frase en LN:

¿Existe algún alumno con buena calificación, que vive en San José y su nota es mayor a 9?

| | |
|---|----------------------|
| ¿Existe algún alumno con buena calificación/ que vive en San José / su nota es mayor a 9? | |
| Pila | |
| <i>Nota</i> | mayor que 9 |
| <i>Alumno</i> | que vive en San José |

Figura 2-8: Ejemplo de tratamiento de estructuras complejas del Sistema de Afonso et al, 2008

Los investigadores concluyen que el aporte de su trabajo es la combinación de una manera dinámica de diferentes aplicaciones de tratamiento del lenguaje: análisis léxico, reglas sintácticas y semánticas, permitiendo esto responder de manera adecuada a la variabilidad del lenguaje.

Más recientemente y en la misma línea, basado en la tarea de hacer una interpretación de NL a SQL, se encuentran otras propuestas, por ejemplo la presentada por Jamil (2012). El trabajo

consta de un *plug-in* que utiliza LN y está orientado a consultar bases de datos biológicas. El método mapea las preguntas realizadas en LN a su equivalente semántico en SQL, para obtener la información solicitada por el usuario de diferentes bases de datos.

Una de las fortalezas de este proyecto es su independencia al esquema de la base de datos. Ésta es una de las principales barreras de los trabajos desarrollados para la traducción de NL a SQL, ya que si se realiza algún cambio en el modelo de la base de datos, debía cambiarse el sistema, ya que los mapeos hacia la base de datos serían distintos dependiendo del cambio que se realizara. Limitar el tipo de pregunta que se puede realizar es otro factor que se tuvo en cuenta en la realización de esta propuesta. Los autores indican que según el estado de la cuestión en esta área, esta restricción impactará directamente en la precisión de los resultados y en la usabilidad del sistema.

Según lo mencionado, se definen 3 tipos de pregunta permitidos por el sistema:

- Interrogativas
- Imperativas
- Declarativas

Se estipula que se deben utilizar preguntas que utilicen los términos (el sistema es para el idioma inglés): “*Does*”, “*Is*”, “*Why*”, “*How*” y “*What*”, entre otros. Las preguntas que utilizan “*Does*” e “*Is*”, principalmente son consultas del tipo existencial o de verificación y pueden ser resueltas haciendo uso del álgebra relacional. Respecto a las preguntas que utilizan: “*Why*” y “*How*”, generalmente implican deducción y requieren un alto grado de

razonamiento en la respuesta. Por sus parte el tipo imperativo y declarativo habitualmente utilizan términos como: “*List*”, “*Print*”, y “*Return*”, y éstas implican una selección.

La siguiente figura muestra el proceso que sigue el sistema, para el tratamiento (proceso y transformación) de las consultas:

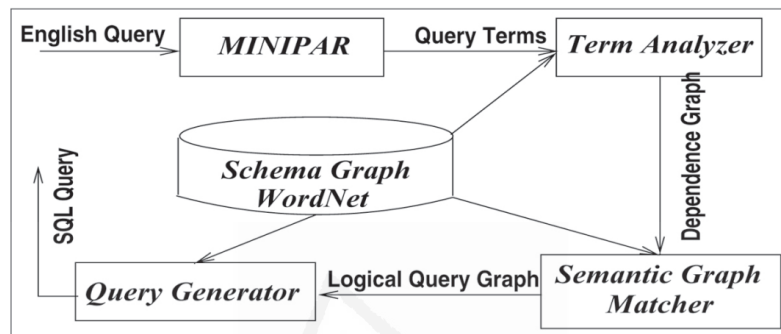


Figura 2-9: Tratamiento de la consulta realizado por el sistema propuesto por Jamil (2012).

Como se puede observar en la Figura 2-9, se presentan los siguientes componentes utilizados para el proceso de la consulta:

- **Analizador automático (parser)**
Utilizando MINIPAR se analiza la consulta y se generan los términos que son pasados al analizador de términos.
- **Analizador de términos**
Se encarga de la representación semántica de la consulta por medio de un grafo de dependencias. Utiliza *Wordnet* como apoyo para encontrar relaciones entre los términos.
- **Mapeo del grafo semántico**
Establece la correspondencia entre los términos y los conceptos, se representa en un grafo. La correspondencia se realiza con apoyo de una ontología que almacena la relación semántica respecto al esquema de base de datos.

- Generador de la consulta
Traduce el grafo a una consulta en lenguaje SQL que pueda ser entendido e interpretado por la base de datos.

Como conclusiones de su trabajo los autores resaltan la independencia de la estructura de la base de datos, por el uso que hacen del mapeo entre la semántica a través de una ontología. Entre sus trabajos futuros proponen la transformación en una vía SQL a LN. Es importante destacar que a pesar de ser una propuesta robusta y bastante completa, el sistema no resuelve la elipsis o la anáfora, ni se ocupa de preguntas contextuales.

En la misma línea que el trabajo anterior, habilitando el paso de LN a SQL, está la propuesta de Pranali, et al., en el 2013, quienes señalan que su propuesta se caracteriza por ser flexible y liviana. El término liviano hace referencia al uso que hacen de herramientas de lenguaje natural, ya que no utilizan análisis sintácticos o semánticos profundos. El paso de LN a SQL lo realizan extrayendo términos clave e indicadores de la consulta en inglés, para trasladarlos luego a una consulta SQL, pasando estas palabras por un pre-proceso y un post-proceso.

La Figura 2-10 muestra la estructura del sistema y se puede visualizar en detalle lo que se realiza en el preproceso y en el postproceso hasta acabar con la formación de la consulta en SQL.

Como se ilustra en la Figura 2-10, en el preproceso:

- Se decide el tipo de consulta, si es un SELECT o DELETE.
- Los números escritos en palabras se sustituyen por su representación numérica.
- Se sustituyen los operadores de comparación en texto por su representación.

- Se quita el apóstrofe y se convierte la expresión.
- Se eliminan las palabras que se considera que no aportan nada a la construcción de la expresión en SQL.
- Se extraen las cláusulas nominales. éstas pueden tener la clave para el nombre de la tabla o un nombre de un campo de una tabla.

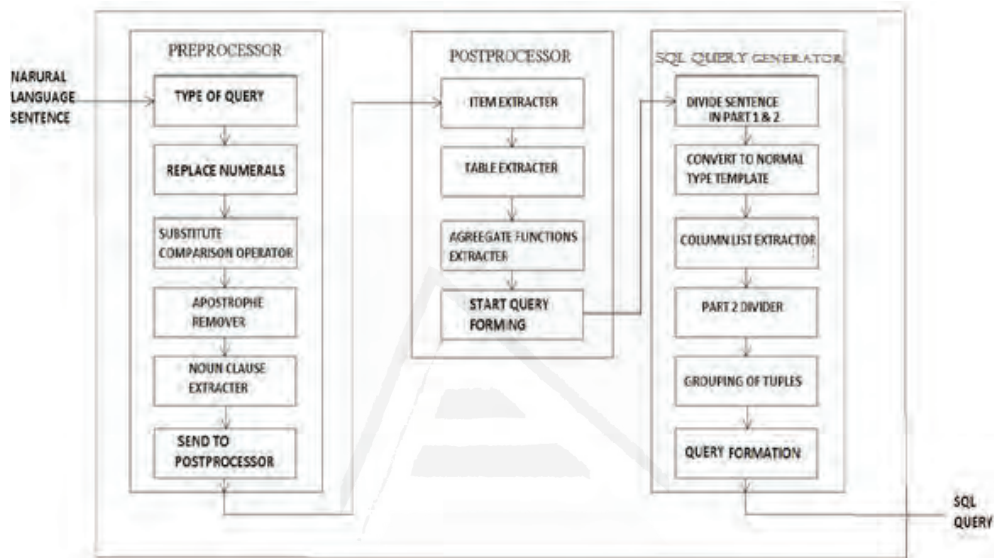


Figura 2-10: Estructura del sistema presentado por Chaudhari, P. P. et al, 2013

Por su parte, en el postproceso:

- El extractor de ítems, define qué debe extraerse de las tablas: valores numéricos u otro tipo de valores.
- Con el extractor de tabla, se identifica el nombre de la tabla.
- Se divide la consulta según aparecen los términos clave: dónde, quién, cuál, etc.

Respecto a la generación de la consulta en SQL, se observa en general que:

- Se divide la consulta.
- Luego se decide el tipo de consulta que es, para aplicar el patrón correspondiente.
- Se eligen las columnas previamente identificadas.
- Se genera la consulta.

Los autores identifican entre fortalezas de su sistema que cuenta con un método de actualización del diccionario de datos, que permite añadir una nueva palabra o frase, que podrá ser utilizada en la cláusula equivalente en SQL. La propuesta de Pranali no tiene en cuenta la resolución de la anáfora ni elipsis ni tampoco aborda las preguntas contextuales, ya que como enfatizan, optaron por una propuesta “liviana” respecto a PLN.

En cuanto a los interfaces que se ocupan de preguntas contextuales, se encuentra la propuesta de Elhai et al. (2009), BioBIKE (*Biological Integrated Knowledge Environment*), cuyo objetivo es permitir a los biólogos que accedan por ellos mismos a las grandes cantidades de información que existen en el área. Su condición de uso es el conocimiento base que deben tener éstos en computación.

BioBIKE está compuesto por 3 grandes componentes:

- Base de conocimiento biológico
Está compuesta por una serie de bases de datos especializadas, disponibles y distribuidas en Internet. BioBIKE las ha unificado y las almacena en un mismo servidor, para generar una mayor eficiencia de acceso al usuario. La siguiente tabla (tabla 2-1) muestra la composición de la base de conocimiento.

| Base de datos | Información contenida |
|---------------|--------------------------------------|
| CyanoBIKE | Cyanobacteria (42 genomas) |
| ParaBIKE | Eukaryotic parásitos (5 genomas) |
| StaphyloBIKE | Estafilococos (45 genomas) |
| StreptoBIKE: | Estreptococos (25 genomas) |
| ViroBIKE | Virus (1797 genomas, 20 metagenomas) |
| BIKE | Usado para educación (0 genomas) |

Tabla 2-1: Bases de datos que componen la base de conocimiento bilógico de BioBIKE

- Interfaz gráfico de programación
Es el componente creado para que el usuario incluya las opciones de búsqueda. Éstas deben definir el formato en que requieren la salida y de dónde requieren que sea extraída la información, entre otras cosas.

En este componente BioBIKE permite construir una serie progresiva de consultas (utilizando un lenguaje BIOBIKE), en las que cada una se debe basar en el resultado de la anterior. Este proceso es similar al utilizado en las preguntas contextuales simples, a diferencia de que el usuario es el responsable de la traducción de LN al lenguaje específico de BIOBIKE de cada pregunta.

Un ejemplo de la manera en que se presenta este interfaz se puede visualizar en la Figura 2-11. Como se puede apreciar, la interfaz requiere de conocimientos básicos de computación y a simple vista no es tan amigable y requiere

de tiempo para el aprendizaje para así dar un uso adecuado a la herramienta.

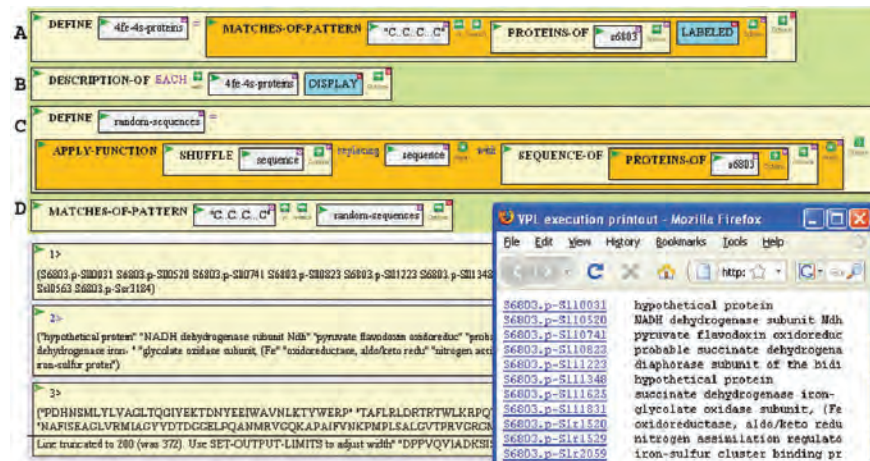


Figura 2-11: Interfaz gráfica de programación de BioBIKE, de Elhai et al. (2009).

- Conjunto de herramientas (ampliable)
 BIOBIKE proporciona acceso a varios programas que son comúnmente utilizados, como:
 - Blast: búsquedas secuenciales.
 - Clustal: múltiples alineamientos de secuencia.
 - RNAz: descubrimiento de secuencias RNA.
 - Phylip: construcción de árboles filogenéticos.

El principal problema con este enfoque es la rigidez de las preguntas, ya que deben plantearse en el lenguaje específico definido por el sistema. Es importante mencionar que BioBIKE no maneja la resolución de la elipsis ni la anáfora.

En Distelhorst et al. (2003), se describe una *plug-in* en LN restringido a una gran base de conocimientos: el *Foundational Model of Anatomy* (FMA). Cuenta con un interfaz llamado

GAPP, que permite manejar preguntas sencillas o anidadas, y éstas se analizan para ser traducidas a un formato: sujeto-relación-objeto.

El proceso que se sigue para procesar las consultas es el siguiente:

1. Análisis de la pregunta con diccionarios de dominio específico.
2. Con el resultado del análisis se pasa al lenguaje de consulta: *StruQL graph-searching*.
3. Se envía la consulta al servidor OQAFMA, y devuelve la salida en XML.

Este proceso está basado en una adaptación del programa *Apple Pie Parser* (APP), que analiza y representa las frases de idioma inglés con su estructura sintáctica. Para lograr la interpretación de la consulta de manera sujeto-relación-objeto, GAPP saca provecho de las regularidades que presenta el idioma inglés en su sintaxis, y por medio de la identificación de patrones en las frases y la combinación de palabras genera el componente *relación*.

APP genera la estructura sintáctica de cada pregunta entrante. Para cualquier frase en inglés, devuelve una estructura sintáctica anidada entre paréntesis y etiquetada. Un ejemplo de la salida que genera se ilustra en la Figura 2-12.

GAPP tiene como estrategia para simplificar el tratamiento de las frases la conversión de los sintagmas adjetivales y adverbiales en sintagma preposicionales. Aunque lingüísticamente esta práctica es poco natural, esta simplificación es bastante útil, ya que el patrón es bastante repetitivo y a menudo se producen en la misma posición en preguntas similares.

Frase en inglés:

What gives blood to the liver?

| Salida de APP |
|--|
| (S (NP what) (VP gives (NP blood) (PP to (NP the liver)))) ? |

Figura 2-12: Ejemplo de la salida de APP para una consulta en inglés
Distelhorst et al. (2003)

La estructura sintáctica de la oración permite a GAPP dividir la pregunta en tres partes, siguiendo el siguiente proceso:

- Sintagma nominal de nivel superior (NP) se marca como una entidad anatómica. El sistema decide después si este NP es una entidad conocida o desconocida.
- Sintagma Verbal (VP), se asume que la relación comienza en el sintagma verbal.
- Fin de la relación: para decidir dónde termina la región relación, GAPP utiliza un archivo definido por el usuario, diccionario de análisis. Utilizando el sintagma verbal como término de búsqueda, a través del diccionario, decide qué parte del sintagma verbal se puede considerar parte de la relación entre las entidades (sin cruzar a la otra entidad anatómica).

En la Figura 2-13, se visualiza la manera en que GAPP divide en 3 partes una pregunta.

Pregunta en inglés:
What gives blood to the liver?

| División de la consulta por GAPP | | |
|----------------------------------|--------------------------------------|-----------------------------------|
| (NP What) Entidad 1 | (VP gives (NP blood) (PP Relación | (NP the liver)))) ? Entidad 2 |
| (NP What) Desconocido | (VP gives (NP blood) (PP Relación | (NP the liver)))) ? Conocido |

Figura 2-13: Ejemplo de la división de una consulta ejecutada por GAPP para una consulta en inglés Distelhorst et al. (2003)

Respecto a las preguntas anidadas, el sistema utiliza el resultado que se obtenga para la consulta interna como entrada para la consulta de nivel superior. Como ejemplo, para la pregunta: "¿Qué parte del tórax contiene el pulmón", el sistema identifica 2 relaciones: "parte de" y "contiene". Ante esto GAPP realiza la consulta interna: "contiene el pulmón" y su salida la utiliza para "parte del tórax". La Figura 2-14 muestra gráficamente el tratamiento de una pregunta anidada.

Pregunta:
¿Qué parte del tórax contiene el pulmón?

| Identificación de preguntas anidadas | | |
|--------------------------------------|----------------------|-----------------------|
| Qué | parte del tórax | contiene el pulmón |
| | Pregunta 1 | Pregunta 2 |
| Qué parte del tórax Desconocido | Contiene Relación | el pulmón Conocido |

Figura 2-14: Ejemplo de preguntas anidadas, Distelhorst et al. (2003)

En lo referente a la evaluación del sistema, los autores indican que tienen dificultades para tratar preguntas con auxiliares “*Wh*” al inicio de la pregunta, y que podría ser resuelto de manera adecuada si utilizaran resolución de elipsis o la anáfora.

Por último, otra propuesta interesante es la de Goldsmith et al. (2008), que se enfrentan al acceso a bases de datos, pero lo abordan desde el punto de vista de la Recuperación de Información (RI). Presentan una tecnología de procesamiento del lenguaje natural que mejora el acceso a la base de datos MEDLINE de resúmenes científicos. Su sistema CSIR (*Cognition Search Information Retrieval*), mejora por medio de la utilización del PLN el acceso a los *abstracts* de la base de datos MEDLINE.

Es una propuesta con alto uso de herramientas de PLN, entre ellas están:

- Lexicon
- Base de datos semánticas
- Detección de patrones utilizando:
 - Sinonimia
 - Relaciones ontológicas
 - Desambiguación

El detalle la arquitectura del sistema propuesto se visualiza en la Figura 2-15.

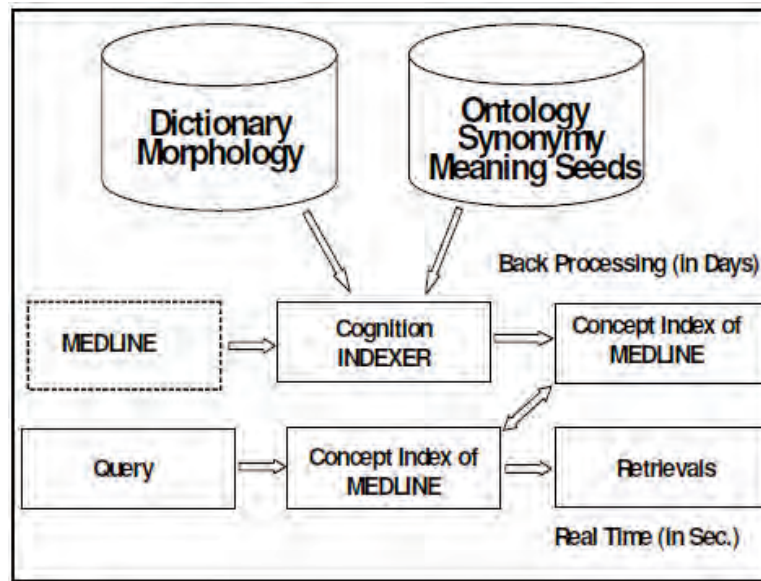


Figura 2-15: Arquitectura del sistema CSIR (Goldsmith et al., 2008)

CSIR contiene un amplio recurso semántico en inglés (mapa semántico), basado en los sentidos de las palabras, sinónimos, hiperónimos; además, posee una base de datos de sentidos que es utilizada para identificar casos particulares que se encuentren en el texto. El proceso de indexado se realiza fuera de línea y en él se almacenan conceptos del documento. Utiliza diccionario, mapa semántico, etiquetas morfológicas y sintácticas y una base de datos de sinónimos y relaciones ontológicas (ver Figura 2-15).

Para la búsqueda, como se aprecia en la parte inferior de la Figura 2-15, la consulta se interpreta según su sentido/concepto, y se compara con el índice generado de conceptos. Se puede clasificar la arquitectura de CSIR como una arquitectura basada en sentido. Respecto a la ontología que se muestra en la Figura 2-15, ésta es creada manualmente y pertenece al área de bioquímica y biología molecular.

El principal problema que se identifica en este enfoque es que las preguntas tienen una estructura muy simple (la mayoría de ellas frases simples) y el sistema no aborda cuestiones de estructura compleja o problemas del PLN.

| Sistema/ LN | Análisis léxico | Análisis sintáctico | Análisis semántico | Resol. elipsis | Resol. anáfora | Ontología/ Semántica | Domino específico | Preguntas contextuales |
|-----------------------|-----------------|---------------------|--------------------|----------------|----------------|----------------------|-------------------|------------------------|
| Popescu et al. | Sí | No | No | No | No | Sí | Sí | No |
| Stratica et al. | Sí | Sí | Sí | No | No | Wordnet | Sí | No |
| NaLIX | Sí | Sí | Sí | No | Sí | Wordnet | No | No |
| Laukaitis & Vasilecas | Sí | Sí | Sí | No | Sí | No | No | No |
| ORAKEL | Sí | Sí | No | No | No | Sí | Sí | No |
| Jamil | Sí | Sí | Sí | No | No | Sí | Sí | No |
| NLS to SQL | Sí | Sí | No | No | No | No | No | No |
| BioBIKE | No | No | No | No | No | No | Sí | Sí |
| Distelhorst et al. | Sí | Sí | Sí | No | No | Propia / Wordnet | Sí | Sí |
| Afonso et al. | Sí | Sí | Sí | Sí | Sí | No | No | No |
| CSIR | Sí | Sí | Sí | No | No | Sí | Sí | No |

Tabla 2-2: Cuadro comparativo de los sistemas de ILN analizados

Como muestra la Tabla 2-2 y se resume de manera gráfica en la Figura 2-16, la mayoría de los sistemas ILN no se ocupan de una resolución de la anáfora y elipsis y sólo algunos de ellos logran tratar las preguntas contextuales, pero con soluciones limitadas (por ejemplo, la restricción de tener que formular preguntas en el lenguaje específico definido por el sistema).

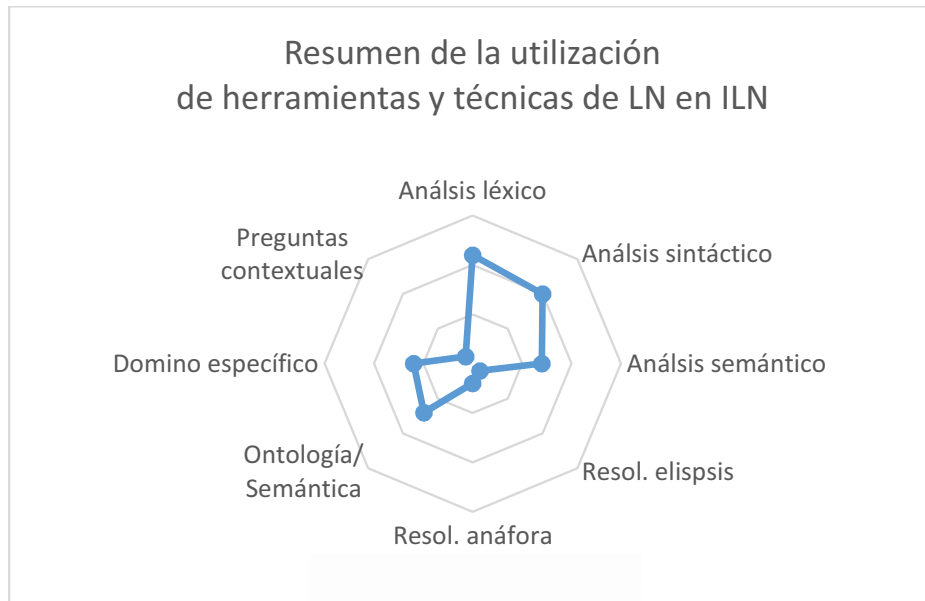


Figura 2-16: Gráfico resumen de la utilización de herramientas y técnicas de LN en los sistemas de ILN analizados

Referente al uso de ontologías, aunque un alto porcentaje de las propuestas las utiliza, son mayormente definidas de manera manual, lo que dificulta la portabilidad a otros dominios de las propuestas. Por su parte, la utilización de análisis léxico y sintáctico es lo más utilizado dentro de los proyectos.

2.2. Preguntas contextuales

La búsqueda de respuestas en dominio abierto tiene como objetivo obtener la respuesta específica y apropiada para una consulta escrita en lenguaje natural. Para esto se han generado foros o espacios en los que se motiva la presentación de iniciativas que permitan la extracción correcta de la información solicitada. Como ejemplo de esos espacios se encuentra la competencia TREC (*Text REtrieval Conference*) o NTCIR (*National Institute of Informatics Test Collection for IR Systems*).

Cuando se crean este tipo de espacios, como es natural, se amplía el rango de aplicación; por ello dentro de la búsqueda de respuestas, se ha propuesto una tarea que se presenta en la vida real: preguntas contextuales.

En este tipo de preguntas, el usuario tiene una necesidad de información como punto de partida, formulado por medio de una consulta; cuando se le retorna la información relevante al respecto, éste irá refinando la búsqueda según los resultados obtenidos. El proceso de refinamiento de la consulta se va dando por el descubrimiento de información que se obtiene de las respuestas obtenidas. Esta área requiere que los sistemas den seguimiento a los objetos del discurso a través de las diferentes consultas.

La Figura 2-17 es un ejemplo de una serie de preguntas realizadas en el TREC 2001, que permitían simular la interacción con el usuario.

| |
|---|
| <p>Q1-1: Which museum in Florence was damaged by a major bomb explosion in 1993?</p> <p>Q1-2: On what day did this happen?</p> <p>Q1-3: Which galleries were involved?</p> <p>Q2-1: What grape variety is used in Chateau Petrus Bordeaux?</p> <p>Q2-2: How much did the futures cost for the 1989 vintage?</p> <p>Q2-3: Where did the winery's owner go to college?</p> <p>Q2-4: What California winery does he own?</p> |
|---|

Figura 2-17: Ejemplo de series de consultas de prueba para sistemas que aborden la búsqueda en preguntas contextuales, Vorhees (2001)

Como se puede ver en el primer grupo de preguntas de la Figura 2-17, es necesaria la información de Q1-1 para resolver Q1-2, en específico reconocer la relación anafórica:

- a major bomb explosion in 1993 that damaged a museum in Florence?

De igual manera, existe una relación elíptica en Q1-3 que requiere ser resuelta:

- which galleries *in the museums in Florence* were involved *in a major bomb explosion in 1993?*

Dentro de las propuestas más relevantes para abarcar este tema se encuentran las presentadas en el TREC (Voorhees, 2001) y NTCIR (Fukumoto et al., 2003, 2004, 2007), dentro de las competiciones de Búsqueda de Respuestas.

Según indica Voorhees, en el 2001, en el informe que hizo sobre la tarea de búsqueda de respuestas contextuales, se presentaron 7 ejecuciones con resultados inesperados. No se halló relación entre la capacidad de responder correctamente a las preguntas de las últimas series respecto la capacidad de responder correctamente a las preguntas en las primeras series. En general, la primera pregunta permitía elegir el conjunto de los documentos que contenían las respuestas correctas y esto impactó positivamente en el rendimiento de los sistemas, más que la capacidad que demostraron de dar seguimiento a las preguntas.

Por su parte, con respecto a las propuestas en el NTCIR, la primera vez que lanzaron este tipo de tarea, en el 2003, Fukumoto describió los resultados que obtuvieron para la tarea que plantearon. Cada problema consistió en dos preguntas sucesivas;

la segunda pregunta, que contenía algunos elementos anafóricos, era el objeto a evaluar.

El sistema más preciso logró un 0,17 de puntuación. Además 14 de las 40 preguntas no fueron respondidas por ningún sistema.

También dentro del ámbito del NTCIR, Kato T. et al., en el 2004, describen la subtarea 3, llamada: “Búsqueda de respuestas para el acceso a información a través del diálogo”. En esta subtarea los sistemas debían responder en tiempo real para hacer la interacción posible. Además de interpretar correctamente una pregunta dada en el contexto de un diálogo específico, debían añadir información relevante no mencionada explícitamente por el usuario. Por otra parte, los sistemas debían ser capaces de plantear una pregunta si consideraban necesario alguna aclaración para resolver la ambigüedad en relación con el objetivo o intenciones del usuario, de este modo se participaría en un diálogo haciendo sugerencias y llevando al usuario hacia la solución del problema.

Esta tarea es importante para el presente trabajo, ya que entre estas diversas capacidades que se esperaba que resolviesen los sistemas y centrándose en el aspecto más fundamental, la interpretación de una pregunta dada en el contexto de un diálogo específico, la subtarea 3 medía las habilidades de procesamiento de los sistemas como la resolución de la anáfora y la elipsis. En este desafío, se solicita a los sistemas responder una serie de preguntas relacionadas. Esta serie de preguntas y las respuestas a esas preguntas comprendían el diálogo establecido. Aunque se suponía que era un diálogo de forma interactiva, la interacción solamente se simulaba; los sistemas deben contestar una serie de preguntas en un modo por lotes. Este tipo de simulación deja fuera aspectos que se dan en una situación real como la dinámica

inherente de diálogo, o la evolución de los diálogos, sin embargo, se plantea como una manera práctica para la evaluación objetiva.

A partir de los informes de los participantes, las técnicas empleadas para el procesamiento de contexto fueron en su mayoría bastante simples:

- La mayoría de los sistemas no analizan expresiones referenciales, sino que simplemente se trata cada consulta como una continuación de la anterior. Por ejemplo, los sistemas que utilizaban palabras claves usaban técnicas de añadir a la siguiente pregunta las palabras clave de la consulta anterior.
- Algunos sistemas utilizaban características con mayor grado de complejidad. Por ejemplo usando bigramas con asignación distinta de pesos según sea el término de la pregunta.
- El sistema que Kato et al. propusieron, agrega como palabras clave a la pregunta actual las respuestas obtenidas de la pregunta anterior.
- Otro sistema empleaba una forma más sofisticada de manejo de contexto:
 - Utilizaba análisis semántico y sintáctico superficial de preguntas.
 - Realizaba una división de la consulta en:
 - Entidad + Atributo + Expresión interrogativa. En la Figura 2-18 se puede ver un ejemplo del tratamiento que hacen a la consulta.
 - Cuando la pregunta actual no tiene ninguna expresión de referencia aparente, es decir, cuando tiene anáfora cero o existe una

elipsis, se hace una revisión entre las expresiones interrogativas de la consulta actual y de la anterior, para determinar si la entidad o el atributo debe ser suministrado como referente.

Cada técnica utilizada, como es obvio, tiene sus ventajas y desventajas. Sin embargo, la evaluación no fue suficiente para permitir evidenciar si existe una relación entre las técnicas empleadas y los resultados obtenidos.

Pregunta:

¿Quién es el presidente de los Estados Unidos?

| División de la consulta | | |
|-------------------------|------------|-------------------------|
| Entidad | Atributo | Expresión interrogativa |
| Estados Unidos | Presidente | Quién |

Figura 2-18: Ejemplo de la división de una consulta ejecutada en NITCR 2004, como parte de la subtarea 3, Kato et al. (2004)

También se ha trabajado sobre las preguntas contextuales en el CLEF tanto en la competencia de búsqueda de respuestas (QA), como en la sesión sobre la búsqueda de respuestas en la transcripción del habla (Question Answering on Speech Transcriptions, QAST) (Lamel et al. 2008). Uno de los retos consistió en responder preguntas basadas en consultas anteriores y sus respectivas respuestas.

Trabajar sobre datos generados del texto hablado genera nuevos retos, ya que la estructura es significativamente distinta al texto escrito, contiene muletillas propias de la persona que habla, o correcciones, inicios falsos, falta de fluidez, frases o palabras

cortadas, entre otros. Además de la gramática que es totalmente distinta a la que utilizamos al escribir. La diferencia de precisión de los sistemas sobre texto hablado se evidenció en la evaluación: a pesar de que los resultados para texto hablado no fueron malos, los resultados obtenidos sobre la colección de texto escrito generaron mejores valores.

Como caso de éxito para modelar el contexto se encuentra el presentado en el 2001, por Harabagiu et al., que utilizaron la resolución de referencias. Dentro de su propuesta ejecutan un proceso de resolución de la correferencia antes del reconocimiento del tipo de respuesta esperada. Identificaron la resolución de las siguientes formas de referencia (para inglés):

- Pronombres demostrativos
 - o “On what day did this happen?”;
- Pronombres en tercera persona
 - o “What California winery does he own?”;
- Pronombres posesivos
 - o “What was his first radio song?”;
- Referencia nominal
 - o “What executive from the company was a member of the Supreme Council in 1994?”;
- Formas nominales de los verbos
 - o “In what facility was it constructed? When was construction begun?”;
- Elipsis
 - o “How many species of spiders are there? How many Ø are poisonous to humans?”;
- Meronimia
 - o “Which museum in Florence was damaged by a major bomb explosion in 1993? Which galleries were involved?”.

El algoritmo de resolución de referencia que proponen, es distinto al utilizado para el procesamiento de diálogo o de discursos. En lugar de resolver las referencias por medio de la información del discurso, este sistema trata de identificar la pregunta o la respuesta que se espera que contenga el antecedente de la referencia. Una vez que se identifica, se realizan una combinación de las palabras clave de ambas preguntas.

Como ejemplo, utilizando la pregunta “Which museum in Florence was damaged by a major bomb explosion in 1993?” y ante la pregunta:

- How many people were killed?

El sistema identifica los antecedentes elípticos en la consulta.

Así que las palabras clave para la búsqueda serían:

- k2 = Florence, k3 = bomb, k4 = explosion, k1 = killed.

Por su parte en la competición de Búsqueda de Respuestas del TREC 2004 se tenía un tema simple para el que se definían un conjunto de preguntas relacionadas al mismo. Los sistemas presentados, en su mayoría, optaron por agregar el tema al final de cada pregunta. Otros lo que utilizaron fue el reemplazo del tema en las preguntas por cada pronombre que apareciese en la consulta. Este último tipo de solución se veía afectado cuando en lugar de utilizar pronombres en las preguntas, se utilizaba sintagmas nominales para referirse al objeto o tema de búsqueda, por ejemplo si el tema meta era el nombre de un conjunto musical:

- Forma de pregunta con pronombre
 - o Ésta ganó 1 premio *grammy*
- Forma de pregunta con sintagma nominal
 - o La banda ganó 1 premio *grammy*

Como se observa, en la segunda forma de realizar la consulta no se puede realizar el reemplazo de pronombres.

Dentro de las propuestas más elaboradas se presentaron sistemas que utilizaron la resolución de la anáfora, sin embargo, por la manera en que se plantearon las preguntas no fue posible evidenciar el impacto de la utilización de esta técnica en la precisión de las respuestas, ya que como se mencionó anteriormente, el conjunto de preguntas se daban con base en el tema específico y los pronombres de las preguntas se referían al mismo (Voorhees, 2005).

Del mismo modo, en el CLEF 2007 para la sesión de Búsqueda de Respuestas Multilingüe (Giampiccolo et al., 2007) se definió un grupo de preguntas relacionadas con cada tema. Los sistemas plantearon mayormente soluciones que utilizaban la relación de correferencia. Esta relación era identificada por la referencia anafórica presente en la consulta, ya fuera respecto al tema declarado implícitamente en la primera pregunta o en su respuesta. En general se identificó una disminución de la precisión en los sistemas cuando se presentaban preguntas enlazadas.

Otro ejemplo del uso del PLN se puede ver en el sistema de Matsuda & Fukumoto (2005). Ellos definieron 3 tipos de relaciones de referencia:

- Pronombres
- Anáfora cero
 - o Verbal
 - Ejemplo: *Tus amigos han venido. Me han dicho que volverán pronto*
 - o Modificador de sustantivos

- Ejemplo: *La hermana de Juan se ha comprado un coche, la de David también.*

Para el análisis del texto utilizaron análisis lingüístico. Además, para el reconocimiento de los casos de anáfora cero, utilizaron casos definidos y coocurrencia, basándose en el diccionario Japonés de coocurrencias, EDR. Los resultados de las propuestas que utilizaron PLN en el NTCIR del 2005 no fueron satisfactorios y estuvieron por debajo del rendimiento mostrado por los otros sistemas.

Se puede concluir que la mayoría de soluciones propuestas para tratar el tema de preguntas contextuales realizan la resolución de la anáfora y la elipsis de forma muy sencilla. Se basan en añadir las palabras clave de preguntas anteriores a la consulta subsiguiente.

2.3. *Resolución de la anáfora*

Como se pudo ver en la sección anterior, muchas de las propuestas optaron por la resolución de la anáfora para tratar de obtener mejores resultados a la hora de abordar las preguntas contextuales.

De manera general se puede clasificar estas técnicas en dos grandes grupos:

- Enfoques tradicionales
- Enfoques basados en el modelado del discurso (especializados en preguntas contextuales)

Por su relevancia para la presente investigación comenzaremos por los enfoques basados en el modelado del discurso.

Las investigaciones sobre este modelado se dirigen principalmente a 2 aspectos:

- ¿Qué información del discurso debe ser capturada?
- ¿Cómo puede ser representada esa información para su correcta interpretación lingüística? (Sun & Chai, 2007).

Para dar solución a ambos aspectos, a través de los años, se han desarrollado una serie de teorías, entre las que destacamos:

- Teoría de Hobbs (Hobbs, 1985)
- Teoría de la Estructura Retórica (Mann y Thompson, 1987)
- Teoría del Centrado (Grosz et al., 1995)
- Teorías para los diálogos (Grosz y Sidner, 1986)
- La teoría de la Representación del Discurso (Mann y Thompson, 1987)

Por su parte Chai y Jin en 2004 analizaron el rol que juega en la búsqueda de respuestas el modelado del discurso. Afirman que cada pregunta y su respuesta cuentan con un estado del discurso, respecto al discurso completo. Para la definición del estado del discurso definen 2 elementos:

1. El rol de las entidades en el discurso en una pregunta y en su respectiva respuesta.

Las entidades como el sintagma nominal, sintagma verbal y sintagma preposicional, entre otros, llevan distintos roles en las preguntas e indican cuál es el tema o foco de una pregunta en cuanto a la información en general que buscan en el discurso.

El tema hace referencia al “acerca de” de la pregunta y el enfoque a la perspectiva específica del tema.

2. Las transiciones del discurso.

Indican la manera en que los roles de discurso van cambiando de una pregunta a otra a medida que avanza la interacción. Además se puede visualizar cómo esos cambios reflejan el progreso de las necesidades de información de los usuarios. Los roles y transiciones del discurso determinan si el contexto es útil y, si es así, la manera de cómo utilizar el contexto para interpretar una pregunta.

Los autores consideran que la forma en que las preguntas se relacionan entre sí depende del cómo (la evolución de los temas de esas preguntas). Categorizan las transiciones de información en tres tipos: extensión del tema, exploración del tema y transición del tema.

Otra propuesta presentada sobre la resolución de la anáfora es la de Sun & Chai en el 2007. Los autores investigan el papel que desempeña el procesamiento del discurso y su implicación en la expansión de la pregunta para una secuencia de preguntas, sobre la tarea de recuperación de documentos (previo al QA). En su investigación examinan 3 modelos, basándose en la teoría para el procesamiento del discurso que generaron en 1995 Groz et al.:

1. Modelo de referencia, resolución de los pronombres de cada pregunta.
2. Modelo avanzado, añade términos de la pregunta anterior basado en lo que considera que pueda impactar en el futuro.

3. Modelo de transición, añade términos de manera selectiva de acuerdo a las transiciones identificadas entre preguntas adyacentes.

En cuanto a los enfoques tradicionales, además de los existentes para la resolución de la anáfora (Zheng et al. 2011), así como la resolución de la anáfora pronominal (Palomar et al., 2001), las descripciones definidas deben resolverse como se ha comentado anteriormente.

Por su parte en Vieira y Poesio (2000) los autores concluyen que las descripciones definidas no son generalmente anafóricas (descripciones que denotan la misma entidad del discurso que su antecedente), ya que la mitad de las veces son utilizadas para introducir una nueva entidad en el discurso. La heurística que proponen para determinar las referencias anafóricas es cuando se detectan las siguientes relaciones semánticas:

- Sinonimia (*una casa* frente a *la casa*)
- Generalización / hiperonimia (*un roble* contra *el árbol*)
- Especialización / hiponimia (*un árbol* contra *el roble*)
- Meronimia

Otra heurística utilizada es cuando tanto la descripción como el antecedente tienen el mismo núcleo nominal y modificadores compatibles (*una casa azul* frente a *la casa*); aunque esto no siempre se cumple, como por ejemplo cuando se utiliza un nombre propio (*Bill Clinton* frente a *el presidente*). Como ejemplo de heurísticas adicionales que han sido propuestas por los autores están la que determina que las descripciones definidas se encuentran en la aposición (*Glenn Cox, el presidente de Phillips Petroleum Co.*) y que las construcciones subordinadas suelen

introducir entidades nuevas, relativas al sintagma nominal al que modifican (*el hombre con más probabilidades de obtener la custodia de todo esto es un político de carrera llamado David Dinkins*).

2.4. Resolución de la elipsis

La elipsis es otro de los fenómenos del discurso que es necesario resolver para un correcto análisis del mismo.

Este tema no es reciente y se viene tratando desde hace ya mucho tiempo. Como ejemplo se encuentra el trabajo realizado en 1983 por Carbonell, donde analiza el manejo adecuado de los fenómenos del discurso más comunes (tales como anáfora, elipsis y otras maneras de abreviación de texto) pueden influir de manera positiva en la interacción con los usuarios. El autor concluye que los usuarios prefieren generar expresiones concisas o fragmentadas en lugar de expresiones largas. Usan generalmente frases independientes. A continuación se muestra un ejemplo:

Para la pregunta inicial:

- ¿Cuál es el precio de los 3 discos duros con más capacidad de un solo puerto?

Consultas que complementan la búsqueda:

- ¿Velocidad?
- ¿Los dos más pequeños?
- El precio de los 2 más pequeños
- El más pequeño con dos puertos
- ¿Velocidad de los de 2 puertos?
- Discos con 2 puertos

Como se puede observar, después de formular la pregunta inicial, las siguientes consultas son concisas y se realizan por etapas, según va surgiendo la necesidad de información, o sea, se realizan de manera fragmentada.

Para el análisis de las preguntas, el autor realiza la distinción entre la elipsis sintáctica y la semántica. La semántica no se refiere a información omitida como estructuras sintácticas incompletas sino como proposiciones incompletas semánticamente. Su propuesta para resolver ambos tipos de elipsis está basada en un enfoque de casos y una sustitución de reglas contextuales.

Por su parte, Díaz de Ilarraza et al. (1990) indican que los 2 mayores subproblemas de la resolución de la elipsis en sistemas de procesamiento de diálogo son:

1. El análisis de una sentencia elíptica
2. La reconstrucción de los fragmentos elididos

Los autores aplican la resolución de la elipsis detectando los elementos obligatorios ausentes en las preguntas. Cuando los valores de los descriptores obligatorios no están presentes, el sistema va a generar posibles instancias (soluciones) que podrían llenar los descriptores.

En cuanto a la elipsis sintáctica, distinguen entre:

1. Sustitución de una categoría sintáctica por otra.
2. Agregación de las cadenas de modificadores a una cadena central.

En su trabajo, Williams, S. (2000) presenta un componente que resuelve la anáfora y la elipsis para un sistema de lenguaje hablado

(español) que proporciona acceso al correo electrónico por medio del teléfono.

El autor ejecuta la resolución mediante el almacenamiento de una lista ordenada de los enunciados que aparecen en las preguntas anteriores. En esta lista, también se almacena un conjunto de entidades específicas de dominio, como “*mensaje, correo electrónico, encabezado o carpeta*”. Estas entidades específicas se buscan primero y si están presentes en la pregunta para resolver entonces se buscan las declaraciones restantes.

La propuesta de Tomioka, S. (2008) se centra en sintagmas verbales elididos (VP), que exigen la presencia del mismo VP o el significado del VP. El autor llega a la conclusión de que la definición de la similitud es una tarea muy difícil. Por lo tanto, propone una estrategia de recuperación de significado similar a la resolución de la anáfora.

3. Propuesta

En el presente capítulo se describe la propuesta para la interfaz en lenguaje natural. Se iniciará con el detalle sobre la arquitectura planteada. A continuación se detallará la solución integrada para el tratamiento de la anáfora y elipsis en preguntas contextuales explicando los algoritmos utilizados. Finalmente, se presenta un caso de prueba que permitirá de manera práctica mostrar el funcionamiento del interfaz y el impacto positivo que tiene la resolución de los problemas lingüísticos en los resultados obtenidos.

3.1. *Arquitectura propuesta*

Como se ha analizado en el capítulo anterior “Antecedentes” y se resume en la Figura 3-1, todos los interfaces propuestos tienen en común los siguientes procesos (el orden puede variar según la propuesta):

- Conexión e interacción con la fuente de información
Se genera una conexión con la fuente de información elegida y (según la propuesta) se generan ontologías u otros recursos para apoyar el acceso a las distintas fuentes de datos.
- Recepción de la consulta
Se recibe la consulta del usuario.
- Tratamiento de la consulta
Dependiendo del sistema, se realiza sobre la consulta un análisis léxico, sintáctico y/o semántico de modo que se pueda generar una interpretación de la consulta. De este

modo es posible representarla adecuadamente para realizar la consulta a las fuentes de datos definidas.

- Recuperación y presentación de resultados

Se muestran los resultados obtenidos. Según la propuesta, esta recuperación incluirá niveles distintos de procesamiento (utilizando fuentes o recursos intermedios de información).

En nuestra propuesta se mantienen los módulos generales de los interfaces en Lenguaje Natural. La diferencia principal radica en el enriquecimiento con información sintáctica y semántica que nos permite realizar una mejor resolución de problemas lingüísticos (como anáfora y elipsis) sacando provecho del uso de preguntas contextuales. Asimismo, la propuesta trata de aprovechar positivamente la experiencia del usuario ya que se realizan correcciones en la consulta y permite la interacción. Se le sugiere mejora en la consulta y se le permite flexibilidad en la selección de las fuentes de datos. Además, presenta una mejora en la presentación de los resultados por medio de un panel de control, en que el usuario cuenta con mayor flexibilidad para la visualización y manipulación de los mismos.

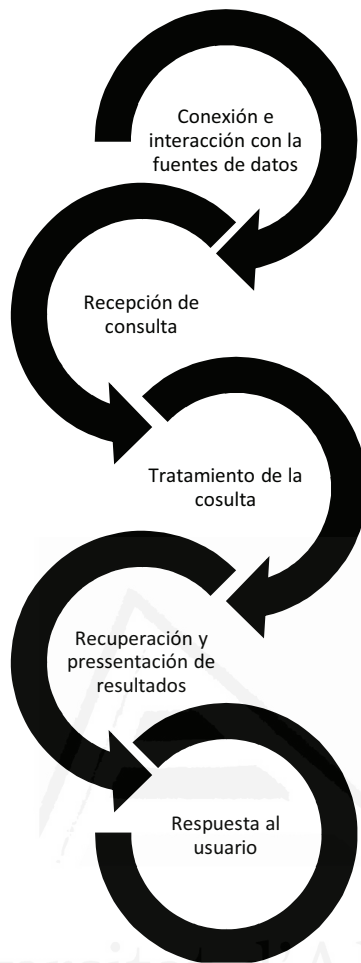


Figura 3-1: Módulos principales y comunes entre Interfaces gráficas que utilizan PLN para acceso a bases de datos.

En las siguientes subsecciones se detalla cada módulo de forma general.

3.1.1 Conexión e interacción con la fuente de datos

Como se mencionó en la Introducción, la generación de información crece de manera exponencial. Ya desde el 2011 el Grupo Gartner reportaba que el volumen de información crecía a

una tasa mínima de 59% anualmente. Ante tanta información se hace necesario poder contar con acceso a diferentes fuentes de datos, no sólo a base de datos estructuradas propias de una organización (información interna), sino que también se hace necesario consultar a fuentes de datos externas: artículos científicos, informes existentes y hasta correos electrónicos generados sobre un tema de estudio.

El tener acceso a diferentes fuentes puede considerarse una barrera para usuarios sin conocimiento técnico previo ya que en el caso de la consulta de una base de datos estructurada, el usuario debería conocer el lenguaje de consulta de la misma además de conocer la estructura que tiene definida. Por otro lado, si es una colección de documentos, su recuperación y revisión demandaría un lapso de tiempo alto.

Una vez que el usuario cuente con cada resultado obtenido, deberá unificar para poder concluir sobre la información obtenida y esto le agregará aún más complejidad a la tarea de búsqueda de información.

Por todo ello, además de permitir al usuario consultar una fuente de datos, por medio de consultas en lenguaje natural, es sumamente relevante que no se le limite el acceso una fuente específica, sino que pueda acceder simultáneamente a diferentes fuentes de datos, en diferentes formatos y que esto sea transparente para él.

El objetivo de la presente propuesta es dotar al usuario, sin conocimientos técnicos previos, de la opción de consultar información en diversas fuentes de datos y expresar esa necesidad en lenguaje natural. El cumplimiento de este objetivo es

precisamente una fortaleza de nuestro proyecto, ya que el sistema se ha definido de arquitectura abierta, donde se permite al usuario elegir diferentes fuentes de datos, aunque éstas estén en formatos distintos. Contar con la apertura hace que esta propuesta sea aún más atractiva, ya que no se condiciona al usuario a que tenga acceso a información restringida y limitada, sino que al acceder a diferentes fuentes de datos se enriquece el proceso de búsqueda de información y repercute de manera significativa en los resultados y en la experiencia del usuario.

En una arquitectura inicial y muy sencilla (con sólo dos fuentes de datos, una interna y otra externa) podemos considerar que el interfaz proporciona acceso a:

- Almacén de datos (DW)
- Sistema de búsqueda de respuesta (QA)

Se debe subrayar que por cada fuente de información seleccionada, se añade de manera dinámica un nodo al sistema. Esto permite tener múltiples fuentes de información, con la ventaja adicional que esos nodos pueden representar fuentes de datos heterogéneas entre ellos. Por ejemplo, a esta arquitectura inicial es posible añadirle nodos especializados en Recuperación de Información, Extracción de Información, Análisis de Sentimientos o incluso diversos nodos de QA, por ejemplo, cada uno específico para un dominio concreto (médico, medioambiental, legislativo, etc.)

En la Tabla 3-1 se puede visualizar un resumen de la definición de cada una de estas opciones para la arquitectura inicial con dos nodos.

| Fuente de datos | Descripción |
|-----------------------------------|---|
| Almacenes de datos | <ul style="list-style-type: none"> • Repositorio de datos históricos recopilado de base de datos heterogéneos de una organización. • Provee un modelo de datos común, que agrupa toda la información interna de una compañía, obtenida de diferentes fuentes, con el objetivo de facilitar el reporte y análisis de la misma. |
| Sistema de búsqueda de respuestas | <ul style="list-style-type: none"> • Representan los futuros motores de búsqueda de Internet. • Retorna la información específica buscada y los documentos en los cuales se localizó la misma. |

Tabla 3-1: Descripción de las fuentes de datos que se tiene acceso a través de la interfaz propuesta

Para el acceso a la fuente de datos, se realiza un proceso automático, y corresponde a la inicialización del sistema para cada fuente de datos. Se debe tener claro que por cada fuente de datos existe un nodo y que pueden existir múltiples de ellos.

A continuación se va a detallar la fase de inicialización de una forma general. Para una mejor comprensión de este proceso, se irá desarrollando en conjunto con cada etapa del proceso el seguimiento de un caso específico basado en un escenario de venta de productos electrónicos. Imaginemos que disponemos de la arquitectura inicial que cuenta con 2 fuentes de datos: nodo de QA y nodo de DW. En este contexto, el usuario formula la siguiente consulta:

“¿Cuál es el precio de los productos Canon durante la temporada de rebajas?”.

En la fase de inicialización, podemos distinguir 3 etapas relativas a las actuaciones en el nodo de QA, en el nodo de DW y en las relaciones entre los dos nodos (por medio de las equivalencias, mapeo, entre las ontologías creadas).

1. Actuaciones en el sistema de búsqueda de respuestas (QA)

Para el tratamiento de esta fuente, se hace necesario la realización de procesos separados:

1.1 Modelo de integración de QA

Este modelo contendrá la información de la respuesta esperada y la manera en que se integrará con la respuesta que se obtenga del DW. Hay que destacar, que este modelo de integración variará dependiendo del sistema de QA específico con el que se trabaje.

Esto es de suma importancia para la propuesta ya que, como se mencionó anteriormente, el usuario tendrá un sólo panel que resumirá la información solicitada, siendo para él transparente de cuál fuente fue tomada la información.

En la Figura 3-2 se puede ver el modelo de integración de QA que contiene la respuesta:

- El sintagma nominal y un *String* de tamaño fijo.
- El tipo de respuesta esperado. En la pregunta del ejemplo, la respuesta es de tipo económico.
- Las entidades detectadas.

Se identifica “productos Canon” como un objeto-producto electrónico.

- La URL o el documento que contiene el pasaje con la respuesta.

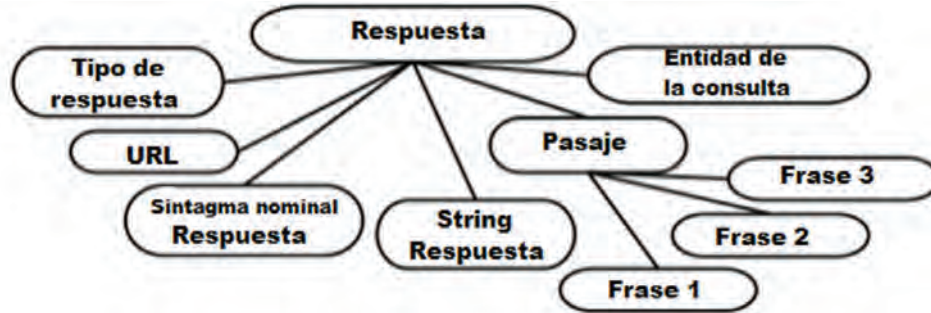


Figura 3-2: Ejemplo de modelo de integración de QA para la respuesta de la pregunta: “¿Cuál es el precio de los productos Canon durante la temporada de rebajas?”

1.2 Ontología de representación de QA

La ontología de QA contendrá la representación de los diferentes tipos de respuesta que se desea obtener de esta fuente de datos.

La Figura 3-3 muestra un extracto de la ontología de la respuesta, en donde un conjunto de los conceptos de nivel superior de *WordNet*: (objeto o persona) se utilizan con algunas extensiones: tipo económico o porcentual en el tipo numérico.



Figura 3-3: Extracto de la ontología de QA

2. Actuaciones en el nodo del Almacén de datos

2.1 Ontología de representación de DW

La ontología se crea con base en el esquema definido para el DW, captura sus hechos, dimensiones y relaciones. Se guarda la relación existente entre las diferentes tablas que componen el DW y sus atributos.

En la Figura 3-4 se muestra un extracto de la ontología de DW.

Universitat d'Alacant
Universidad de Alicante

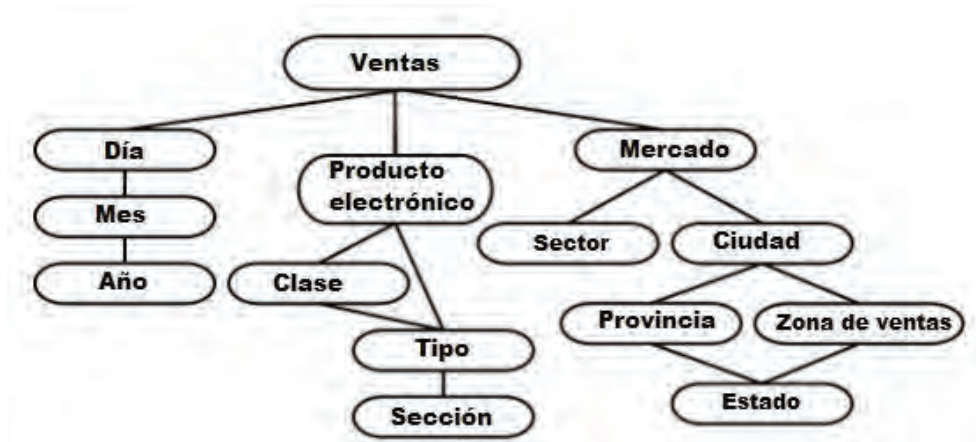


Figura 3-4: Extracto de la ontología de DW

3. Mapeo entre ontología de QA y ontología de DW

Este proceso se ejecuta de manera semiautomática y su objetivo es lograr una homogeneidad entre fuentes de modo que el acceso y generación de información se realice de manera fluida y esto influya directamente en el rendimiento del sistema.

De manera general, este proceso sigue los siguientes pasos:

3.1 Detección de clases/propiedades equivalentes entre ambas ontologías.

Primeramente se recuperan las equivalencias exactas. En la Figura 3-5 se puede ver las clases equivalentes: Día, mes y año, que se encuentran en las ontologías de DW y QA (se evidencia en Figura 3-4 y la Figura 3-3).

El resto de conceptos es mapeado utilizando la información de los recursos léxicos y semánticos utilizados en QA: *WordNet*, lexicones, diccionarios, glosarios, etc. Una vez que se generan, se solicita al usuario confirme las relaciones del mapeo propuesto. Por ejemplo, la equivalencia mostrada en el Figura 3-5 entre producto electrónico y objeto, se da gracias a la relación de hiperonimia existente en *WordNet* entre los términos producto y objeto; otra relación equivalente es la que existe entre Precio que está en DW contra económico en QA.

3.2 Se enriquecen los recursos léxicos y semánticos con las instancias de la ontología de DW.

En la Figura 3-6 se muestra un ejemplo del enriquecimiento de la ontología con *WordNet*, donde instancias de productos electrónicos almacenados en DW (Asus P5KPL-AM EPU, etc.) se agregan al recurso léxico. Con esta expansión, el sistema será capaz de tratar consultas basadas en estas nuevas instancias.

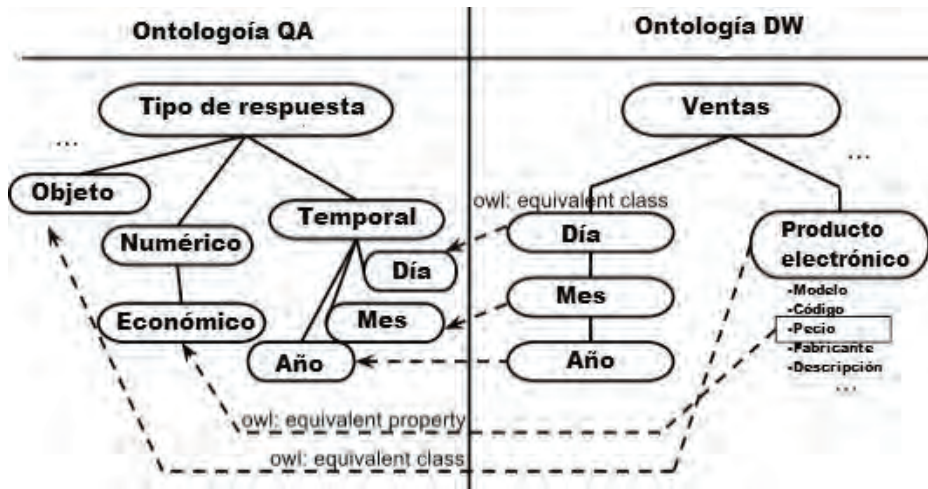


Figura 3-5: Mapeo entre ontologías QA y DW



Figura 3-6: Enriquecimiento de la ontología con *WordNet*

En la Figura 3-7 se muestra un resumen gráfico de los pasos para la conexión a las fuentes de datos. Se identifica, de manera general, cada componente que se genera y los recursos y agentes que intervienen en el mismo.

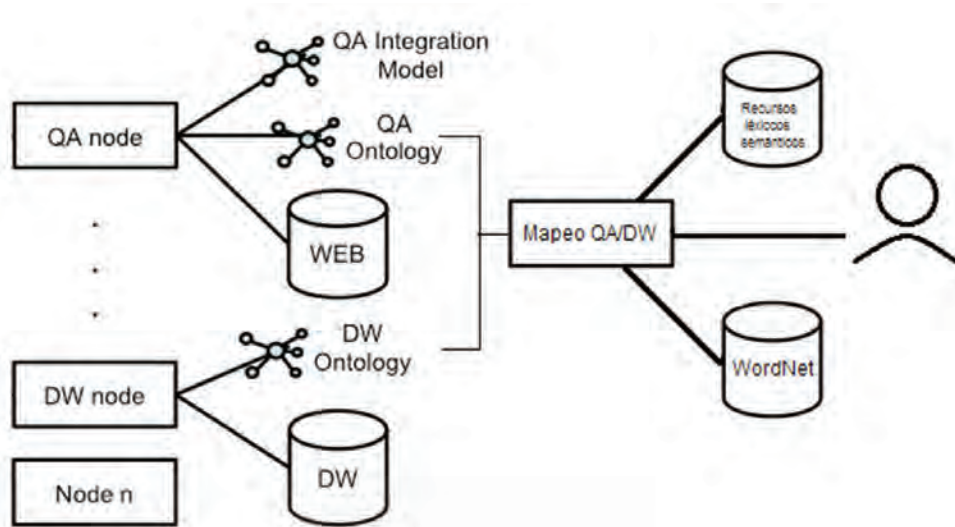


Figura 3-7: Resumen de los procesos básicos para la conexión a diferentes fuentes de datos

3.1.2 Recepción de la consulta

Como se ha mencionado en la subsección anterior, el objetivo de la presente propuesta es brindar flexibilidad al usuario a la hora de realizar una consulta; se evita de esta manera que deba tener conocimiento previo en tecnologías de información o que requiera de apoyo adicional para obtener los datos que requiere. Por lo tanto, la recepción y tratamiento de la consulta es uno de los diferenciadores de este proyecto.

En esta etapa, el usuario hace su solicitud de información expresada en lenguaje natural, no en un lenguaje específico indicado por el sistema. Conforme el usuario expresa su necesidad, la herramienta de apoyo al autor, sacando provecho de los

recursos generados en la etapa anterior, va sugiriendo al usuario si tiene algún error léxico, sintáctico o semántico. Todo este apoyo para esta etapa permite en gran medida asegurar el éxito de la recuperación acertada de los resultados.

3.1.3 Tratamiento de la consulta

Una vez capturada la consulta se inicia el tratamiento de la misma. Este proceso constituye el punto central de la tesis.

Como se ilustra en la Figura 3-8, la consulta debe pasar por análisis morfológico y sintáctico. Con base a los resultados de estos análisis, se sigue con la resolución de la anáfora y luego la elipsis; durante todo este proceso se va generando una serie de propuestas para apoyar al usuario a mejorar su pregunta. La herramienta de apoyo a la edición, permite tomar estos resultados, hacerlos llegar al usuario y esperar la respuesta sobre las consultas que el usuario considera se acercan a lo planteado. Durante todo el proceso se tiene cuidado de respetar las estructuras definidas en el DW (o cualquier fuente de información) de modo que al realizar la consulta sea bien estructurada y no se genere algún error de sintaxis.

El usuario también puede visualizar la interacción de las consultas previas y sus repuestas previas (con los procesos de resolución de anáfora y elipsis); esto permite generar contexto para apoyar estos procesos de resolución. En la sección 3.2 se tratará con detalle cada uno de estos procesos y sus interacciones.

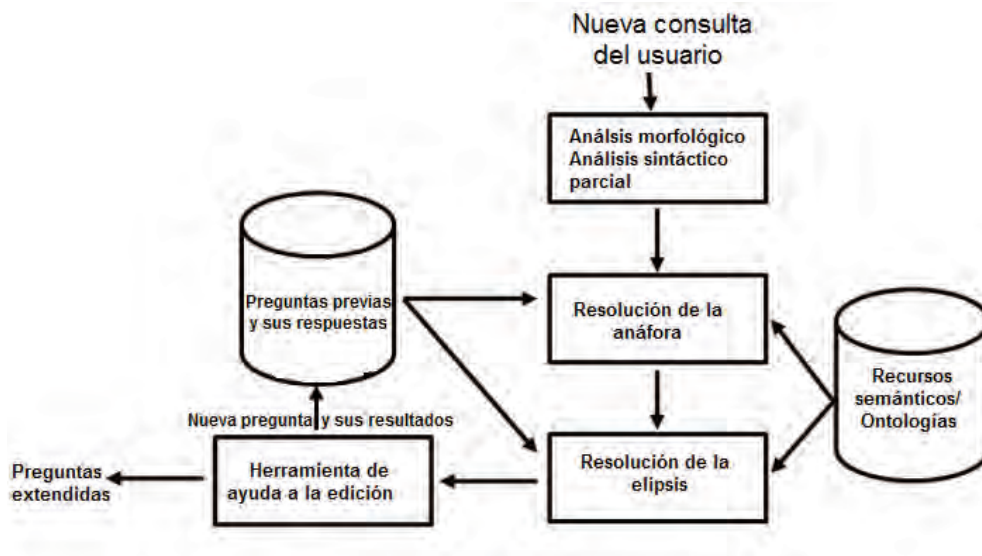


Figura 3-8: Tratamiento de la nueva consulta del usuario

3.1.4 Recuperación y presentación de resultados

Uno de los retos al contar con fuentes de datos híbridas es el lograr la estandarización de los resultados provenientes de las diferentes fuentes, de modo que al momento de dar la respuesta al usuario, le sea útil, entendible y que se le permita la manipulación de la misma.

Este reto ha sido asumido en esta propuesta, generando como salida un cuadro de mando, donde el usuario puede seleccionar y filtrar información. Pero la clave se resuelve en una etapa previa, por medio del tratamiento de la salida de las respuestas. Se cuenta con una base de datos diseñada para que los resultados generados por el sistema de QA se almacenen de modo estructurado y esto se define desde el módulo de integración de QA, en el momento en que se definen las fuentes de información. De esta manera se cuenta con una estandarización de los resultados.

La siguiente sección detallará el funcionamiento de cada componente de la arquitectura y evidenciará las mejoras sustanciales que se tiene al contar con el tratamiento de las preguntas contextuales, la resolución de la anáfora y la resolución de la elipsis.

3.2. Detalle del interfaz gráfico propuesto

En esta sección se presenta en detalle el interfaz gráfico (Graphical User Interface, GUI) propuesto. Se explicará cada fase, además de los algoritmos utilizados para el tratamiento de preguntas contextuales, resolución de la anáfora y la elipsis. Los ejemplos que se desarrollen para esta sección se basarán en casos de una empresa fitomejoradora (o mejoradora de plantas), que desea llevar a cabo nuevos programas de mejoramiento a través de experimentos con los nuevos avances en Genética. Dicha empresa utilizará una fuente interna (un DW) y también se consultarán fuentes externas de información (como por ejemplo Internet).

3.2.1 Fase de configuración

En esta fase se realiza la inicialización (explicada con detalle en la sección 3.1.) de los nodos correspondientes a las fuentes de datos accesibles al sistema y se preparan para la generación de las ontologías respectivas. Por ejemplo, la primera vez que se conecta al DW interno, se crea la ontología que describe su esquema y se realiza el mapeo con el resto de ontologías, lo que permite realizar la integración con el resto de nodos conectados a la plataforma.

Esta ontología se crea de manera semiautomática, con base en lo propuesto en 2010 por Santoso et al., y sigue 5 pasos:

- a. Por cada dimensión existente en el modelo conceptual del DW se crea un concepto.
- b. Cada concepto obtenido de la representación del DW, se relaciona con los niveles más finos (específicos) en cada dimensión.
- c. Se transforma cada relación *drill down* en una relación “tiene miembros” (*has_members*) en la ontología.
Se transforma cada relación *roll up* en una relación “es un miembro de” (*is_a_member_of*) entre conceptos ontológicos de la ontología.
- d. El cuarto paso es manual, lo realiza el usuario. Consiste en el enriquecimiento de la ontología con información semántica y, además, se realiza la revisión de las relaciones definidas en los pasos anteriores con el objetivo de mejorarlas o corregirlas.
- e. El último paso consiste en poblar la ontología de manera automática con la información guardada en el DW donde cada dato es guardado como una instancia del nodo correspondiente. Esta ontología se va a utilizar como la base de conocimiento y servirá de apoyo a la resolución de la anáfora, resolución de la elipsis y el uso de preguntas contextuales.

3.2.2 Solución integrada para el tratamiento de la anáfora, elipsis y preguntas contextuales

Esta subsección expone la propuesta utilizada para la resolución de la anáfora, elipsis y preguntas contextuales. Es importante recalcar que éste es el elemento diferenciador de la propuesta: se han incorporado de manera exitosa una serie de técnicas que han permitido mejorar la interpretación y manipulación de la consulta introducida por el usuario.

El GUI propuesto ha sido incluido en el trabajo realizado en 2012, por Llopis et al. Es utilizado como una herramienta de apoyo al autor que, de manera interactiva, da soporte en la construcción de la consulta, ya sea completando palabras, o indicando errores sintácticos en las mismas (por medio de subrayado o utilizando colores distintos en el texto). Este apoyo influye directamente en la usabilidad del sistema, ya que permite al usuario detectar desde el inicio errores en la formación de la consulta, ya sean errores léxicos o sintácticos; también se pueden indicar errores más complejos según las relaciones existentes en las tablas de las fuentes de datos.

El GUI, como se mencionó anteriormente, ha sido extendido para que pueda tratar la resolución de la anáfora, resolución de elipsis y de preguntas contextuales. Como se visualiza en la Figura 3-8, una vez se recibe la nueva consulta del usuario, se inicia analizándola léxicamente (a partir de ese paso se genera el lema e información léxica de cada término). A continuación, es analizada sintácticamente (de una forma parcial) obteniendo el árbol sintáctico y enriquecida semánticamente (a partir de los recursos semánticos y ontologías). Con todo ello se generan estructuras de datos con suficiente información que apoyan el proceso de resolución de la anáfora y elipsis (se ven con detalle en la siguientes subsecciones).

Los módulos de resolución atacan el problema del contexto teniendo en cuenta las preguntas previas y sus respuestas previas, que son almacenadas, como está ilustrado en la Figura 3-8. Este proceso de mejorar la consulta generará para el usuario una serie de preguntas extendidas que le permitirá seleccionar la que más

se ajusta a su petición, todo esto sin que el usuario deba conocer cuáles son las estructuras de las fuentes de datos elegidas.

Con la consulta seleccionada, a través del módulo Distribuidor/Integrador (presente en la arquitectura general del sistema completo) se obtiene la información que da respuesta a la consulta accediendo a las diferentes fuentes de datos presentes en la arquitectura. La información obtenida se estandariza y se presenta al usuario a través de un cuadro de mando que permite al usuario sacar provecho de la información que se le suministra sin que se requiera conocimiento experto para evaluar las salidas.

3.2.3 Detalle de algoritmos propuestos

La presente subsección aborda el detalle de los algoritmos propuestos para tratar con las preguntas contextuales y la resolución de la anáfora y elipsis.

El primer algoritmo que se presenta es el que se encarga de la interacción entre los diferentes módulos. Esta interacción se realiza utilizando como apoyo diferentes estructuras de datos. El segundo algoritmo aborda la resolución de la anáfora y el tercero la resolución de la elipsis.

Algoritmo 1: Algoritmo integrado para manejar: la anáfora, la elipsis y preguntas contextuales.

- Funcionamiento:

El algoritmo inicia tomando la pregunta del usuario en Lenguaje Natural (Natural Language Question, NLQ) y se realiza un análisis sintáctico parcial de la misma. Como resultado de este análisis, se identifican los sintagmas nominales, preposicionales y verbales. Las palabras no analizadas se saltan y el análisis continúa.

Cada sintagma extraído representa los principales conceptos de la consulta. Se almacenan secuencialmente (según su orden de aparición en la consulta) en la lista *parsedSentence*, que guarda para cada sintagma sus valores léxicos (en la estructura *conc*, por ejemplo el género o número), un identificador y la información sintáctica correspondiente. Además se guarda cada término de la consulta y su lema. Toda esta información es almacenada en una estructura denominada estructura de huecos, *slot structure* (Ferrández et al., 1999). Un ejemplo de la estructura *parsedSentence* se puede ver en la Figura 3-9. Mencionar que la estructura del tipo lista se representará entre corchetes “[“ y “]”.

```

Para la consulta:
“What QLTs are related to frost tolerance in wheat?”

Se genera la estructura parsedSentence:
[
  what
  np (conc (properName, plural), X, n(QLTs))
  vp (conc (plural), Y, v(are), v(related), prep(to))
  np (conc (commonName, sing), Z, n(frost), n(tolerance), T)
  pp (conc (), T, prep(in), U)
  np (conc (commonName, sing), U, n(wheat))
  ?
]

```

Figura 3-9: Ejemplo de estructura *parsedSentence*

En la Figura 3-9 podemos ver que se han detectado dos sintagmas nominales “QLTs” y “frost tolerance”; un sintagma preposicional “in wheat”; el sintagma verbal “are related to”; y las palabras aisladas “what” y “?”.

Una vez se tiene la estructura *parsedSentence*, el siguiente paso consiste en extenderla con la información semántica que se obtiene tanto de la ontología generada como de los recursos léxicos externos, tales como *WordNet*.

La estructura extendida se almacena en una lista de listas de estructuras de huecos, con nombre *NLQ_concepts*, que contiene sugerencias sobre reformulación de la consulta. Estas propuestas de preguntas serán presentadas al usuario por la herramienta de apoyo al autor.

Las estructuras de *NLQ_concepts*, que coincidan con las existentes en la ontología, serán identificadas como los conceptos más importantes para el resto del proceso de búsqueda. Por ejemplo, el concepto “QTL” es equivalente al nodo “QTL” en la ontología creada en el caso de estudio que veremos más adelante; por su parte, el concepto “frost tolerance” se encuentra como una instancia del nodo “Trait” en la mencionada ontología. Las coincidencias parciales se utilizan para la reformulación de la consulta.

Posteriormente, cada estructura de huecos se enriquece utilizando el recurso léxico *WordNet* utilizando las relaciones de hiperonimia, hiponimia, sinonimia y el resto de relaciones existentes en *WordNet*. Esta extensión puede ejemplificarse con la consulta: “*Which genes influence the resistance to plagues in wheat?*”; el concepto *plagues* no se encuentra en la ontología, sin embargo a través de la relación de sinonimia de *WordNet* se encuentra su relación con “*group/swarm of insects*” que es además un rasgo de “*wheat*”. Por otra parte, las instancias de rasgo:

“resistance to Russian wheat aphid” y “resistance to hessian fly” se refieren a plagas ya que *aphid* y *fly* son insectos. Por todo ello, las consultas que se propondrían al usuario serían:

- “*Which genes influence the resistance to Russian wheat aphid in wheat?*”
- “*Which genes influence the resistance to hessian fly in wheat?*”

La expansión semántica de cada concepto, implica un cambio en la estructura *conc*, que se ilustró en la Figura 3-9. Ésta se extiende agregando 2 campos más a *conc*: el primero tiene que ver con su coincidencia en la ontología y el segundo es una lista con los *synsets* relacionados con el concepto en *WordNet*. En la Figura 3-10 se puede ver un ejemplo. Si no se encuentra coincidencia en alguno, se indica por medio del símbolo “_” en la estructura. Un ejemplo se muestra en la Figura 3-11, de la no coincidencia de un término en *WordNet* (en este caso QTL no es encontrado en *WordNet*).

Para el concepto wheat:

Estructura en *parsedSentence*:

```
np (conc (commonName, sing), U, n(wheat))
```

Estructura en *NLQ_concepts* (luego de la extensión semántica):

```
np (conc (commonName, sing, [Species], [12142085, 078003545]), U, n(wheat))
```

Figura 3-10: Ejemplo de extensión semántica de los conceptos en *NLQ_concepts*

| |
|--|
| Para el concepto QLTs: |
| Estructura en <i>parsedSentence</i> : |
| np (conc (properName, plural), X, n(QLTs)) |
| Estructura en <i>NLQ_concepts</i> (luego de la extensión semántica): |
| np (conc (properName, plural, [QTL], [□]), X, n(QLTs)) |

Figura 3-11: Ejemplo de la representación de la no coincidencia de un concepto en *NLQ_concepts*

Una vez se genera *NLQ_concepts*, como se visualiza en la Figura 3-10, se aplica la resolución de la anáfora y resolución de la elipsis. Estos procesos reciben como entrada: *NLQ_concepts*, *previous_questions*, *previous_answers*, además la Ontología y *WordNet*. El detalle de estos algoritmos se abordarán en las próximas subsecciones.

- Variables:

Los parámetros de entrada y la descripción de cada uno se pueden ver en la Tabla 3-2:

| Parámetro | Descripción | Tipo |
|--------------------|--|------------------------------|
| NLQ | Nueva consulta del usuario | Cadena de caracteres |
| Ont | Ontología de DW | Ontología |
| WN | Recursos semánticos que utilizan en común todos los nodos, ejemplo WordNet | Recursos semánticos |
| previous_questions | Consultas previas que ha ido generando el usuario | Lista de listas de registros |
| previous_answers | Respuestas de las preguntas previas | Lista de listas de registros |

Tabla 3-2: Parámetros de entrada del algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales.

La Tabla 3-3 describe la variable local *parsedSentence*

| Variable | Descripción | Tipo |
|----------------|---|--------------------|
| parsedSentence | Se utiliza para almacenar el resultado del análisis léxico y sintáctico de la consulta. | Lista de registros |

Tabla 3-3: Variable local del algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales.

La Tabla 3-4 describe la variable de salida *NLQ_concepts*

| Variable | Descripción | Tipo |
|--------------|---|------------------------------|
| NLQ_concepts | Información léxica, sintáctica y semántica de cada concepto de la consulta. Además almacena propuestas alternativas de la consulta. | Lista de listas de registros |

Tabla 3-4: Variable de salida del algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales.

- Estructuras de datos que utiliza:

| Nombre de la estructura | Descripción | Tipo |
|---------------------------------|---|------------------------------|
| <code>parsedSentence</code> | Almacena para cada frase de la consulta: - valores léxicos - un identificador - información sintáctica - términos de la consulta - lema de cada término de la consulta. | Lista de registros |
| <code>NLQ_concepts</code> | Lista de listas que contiene la extensión de la estructura <i>parsedSentence</i> con información semántica de cada concepto de la consulta. Además esta extensión le permite generar consultas adicionales para que el usuario pueda mejorar la propuesta inicialmente. | Lista de listas de registros |
| <code>previous_questions</code> | Consultas previas que ha ido generando el usuario, representadas como una de listas de registros que contiene la información léxica, sintáctica y semántica de cada concepto de las consultas previas generadas por el usuario. | Lista de listas de registros |
| <code>previous_answers</code> | Respuestas de las preguntas previas. Tiene el mismo formato que <code>previous_questions</code> . | Lista de listas de registros |

Tabla 3-5: Estructuras utilizadas por el algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales.

La Tabla 3-5 resume las estructuras utilizadas por el algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales. Ejemplos detallados de todas estas estructuras tras los análisis morfológico, sintáctico y resolución de la anáfora se muestran en el Anexo 2.

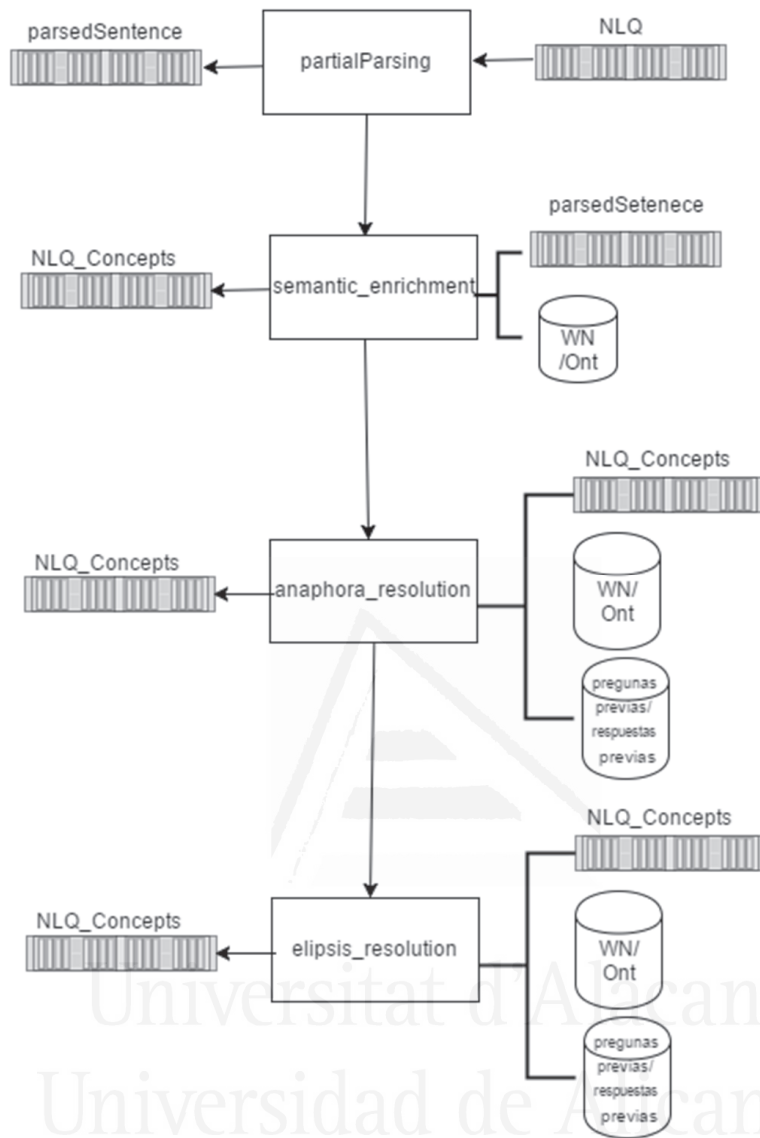


Figura 3-12: Diagrama del algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales.

- Código del algoritmo:
 En la Figura 3-13 se muestra el código correspondiente al algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales. El flujo del algoritmo se puede visualizar en la Figura 3-12.

```

ALGORITHM anaphora_ellipsis_context_questions_resolution

INPUT
NLQ: string;
Ont: Ontology;
WN: Semantic_resources;
previous_questions: list_of_list_of_slot_structures;
previous_answers: list_of_list_of_slot_structures;

OUTPUT
NLQ_concepts: list_of_list_of_slot_structures;

VAR
parsedSentence: list_of_slot_structures;

BEGIN
//análisis parcial de la consulta
parsedSentence = partialParsing(NLQ);

//Mapeo de frases apoyado con los recursos ontológicos y semánticos
NLQ_concepts = semantic_enrichment(parsedSentence, Ont, WN);

//detección y resolución de la anáfora
NLQ_concepts = anaphora_resolution(NLQ_concepts,
previous_questions, previous_answers, Ont, WN);

//detección y resolución de la elipsis
NLQ_concepts = ellipsis_resolution(NLQ_concepts,
previous_questions, previous_answers, Ont, WN);

END ALGORITHM

```

Figura 3-13: Código del algoritmo integrado para manejar la anáfora, elipsis y las preguntas contextuales.

Algoritmo 2: Algoritmo para la resolución de la anáfora

- **Funcionamiento:**

Como se puede ver en la Figura 3-13, el algoritmo para la resolución de la anáfora es el paso tras generar *NLQ_concepts* para la consulta actual. Tiene 3 parámetros de entrada: *NLQ_concepts*, *previous_questions*, y *previous_answers*. Estas 3 listas han sido enriquecidas con la información semántica, tal y como se ha descrito anteriormente.

El primer paso de este algoritmo es la extracción de antecedentes de las 3 listas de entrada. Consiste en el almacenamiento de las estructuras de huecos que son posibles soluciones de las anáforas. Por ejemplo, los sintagmas nominales se almacenarán como posibles soluciones para las anáforas pronominales.

El resultado de este proceso se almacena en una estructura llamada *list_antec* en la que la información está agrupada por consulta. En la sección Estructuras de datos que utiliza, se describe con detalle *list_antec*.

La búsqueda de una anáfora se realiza en *NLQ_concepts* de la pregunta actual a través de la información léxica y sintáctica (en la Figura 3-14 se muestra un ejemplo de cada una).

| |
|---|
| Información léxica Anáfora pronominal: pron (conc (thirdPersonl), ...) Información sintáctica Descripción definida: np (conc (...), ..., determiner (the), n(tolerance)) |
|---|

Figura 3-14: Ejemplo de búsqueda de la anáfora en *NLQ_concepts*

La resolución de la anáfora se realiza al sustituir la misma por la lista de las estructuras de huecos de las posibles soluciones o antecedentes a los cuales podría referirse (estos antecedentes se almacenan ordenados con base en la probabilidad o certeza). Esto se realiza por medio del

método *generation_of_reformulations* que genera nuevas formulaciones de la consulta.

En la presente propuesta se utiliza la resolución de la anáfora pronominal basada en la propuesta por Palomar et al. (2001) y la resolución de las descripciones definidas está basada en Muñoz et al. (2000).

En la resolución de la anáfora se utiliza el conocimiento morfológico, sintáctico y semántico almacenado en las estructuras de huecos generadas a partir de las consultas previas y de sus correspondientes respuestas. Para cada anáfora se almacenan ordenados sus posibles antecedentes distinguiendo entre restricciones y preferencias que se aplicarán a los antecedentes candidatos. Cada tipo de anáfora tiene un conjunto de restricciones y preferencias. En la Figura 3-15 se muestra un ejemplo de éstas.

| |
|--|
| <p>Anáfora pronominal:</p> <p>Restricción</p> <p><code>anaphor.constraint()</code>: concordancia en número/género. Restricciones c-command.</p> <p>Descripción definida:</p> <p>Preferencias</p> <p><code>anaphor.preference()</code>: heurísticas del tipo coincidencia del núcleo del sintagma nominal o la coincidencia de los modificadores del sintagma nominal</p> |
|--|

Figura 3-15: Ejemplo de reglas de restricción y preferencia aplicadas en la resolución de la anáfora.

Para ejemplificar el proceso utilizaremos la consulta que se describe en la Figura 3-16 compuesta por 3 preguntas. La primera es una pregunta bien formada, la segunda requiere

resolución de elipsis y la tercera resolución de la anáfora. Nos centraremos en la tercera pregunta. Como se puede observar, el usuario solicita información acerca de los rasgos que son afectados por los “*QTLs*” que fueron encontrados como respuesta de la segunda pregunta.

Pregunta con referencia anafórica:
 What QTLs are related to frost tolerance in wheat? In barley? Which traits are affected by them?

Figura 3-16: Ejemplo de pregunta con referencia anafórica.

La Figura 3-17 muestra las estructuras de huecos almacenadas en *NLQ_concepts* para la tercera pregunta.

[which
 np (conc (commonName, plural, [Trait], [04616059]), A, n (traits))
 vp (conc (plural, __, [00137313, 00019448, 02677097]), B, v (are),
 v(affected))
 pp (conc (_, _), C, prep (by), D)
 pron (conc (thirdPerson, plural, __, __), D, **them**)
 ?]

Figura 3-17: Ejemplo de *NLQ_concepts* para: “*Which traits are affected by them?*”.

En el momento en que se tiene las estructuras, se detecta una expresión anafórica: pron (conc (thirdPerson, plural, __, __), D, **them**).

Para resolver la anáfora se requiere de contexto. Como se muestra en el Figura 3-18, se tienen 2 preguntas previas que permiten apoyar la resolución de anáfora.

| previous_questions | previous_answers |
|---|--|
| <pre>[[what np (conc (properName, plural, [QTL], _), X, n(QTLs)) vp (conc (plural, _, [00713167, 02724417, 02458103]), Y, v(are), v(related), prep(to)) np (conc (commonName, sing, [Trait], [05033410]), Z, n(frost), n(tolerance), T) pp (conc (_, _), V, prep(in), W) np (conc (commonName, sing, [Species], [12142085, 07803545]), W, n(wheat)) ?], [what np (conc (properName, plural, [QTL], _), X*, n(QTLs)) vp (conc (plural, _, [00713167, 02724417, 02458103]), Y*, v(are), v(related), prep(to)) np (conc (commonName, sing, [Trait], [05033410]), Z*, n(frost), n(tolerance), T) pp (conc (_, _), V, prep(in), W) np (conc (commonName, sing, [Species], [07803093, 12123244]), W, n(barley)) ?]]</pre> | <pre>[[np (conc (properName, sing, [QTL], _), X1, n(QWin.ipk-6A))], [np (conc (properName, sing, [QTL], _), X2, n(QWsv.DiMo-5H.1)) np (conc (properName, sing, [QTL], _), X3, n(QWsv.DiMo-5H.2))]]</pre> |

Figura 3-18: Ejemplo de estructuras de huecos almacenadas en *previous_questions* y *previous_answers* para las preguntas:

“*What QTLs are related to frost tolerance in wheat? In barley?*”

Estas 2 preguntas con sus respuestas se encuentran almacenadas en la estructura *previous_questions*, *previous_answers*, según corresponde. Como se abordó en la sección anterior, ambas estructuras son listas de listas. Por ejemplo *previous_questions* = [[estructuras de huecos de Q1], [estructuras de huecos de Q2]]. En la Figura 3-18

se muestra el contenido de *previous_questions* y *previous_answers* para las 2 primeras preguntas.

En el proceso de resolución de la anáfora tras su detección, la lista de posibles antecedentes está formada sólo por el sintagma en plural “QTLs”. Dado que esta solución es el sintagma nominal que delimita el objetivo de búsqueda de la segunda pregunta, la lista de soluciones debe estar conformada por la estructura de huecos de “QTLs” y las estructuras de huecos de las respuestas de la segunda pregunta que se encuentran ya almacenadas en *previous_answers: QWsv.DiMo-5H.1, QWsv.DiMo-5H.2*. Una vez encontrados los antecedentes se procede a la sustitución de los mismos. En este caso se le mostrará al usuario las consultas para que utilizando la herramienta de apoyo al autor, seleccione la que considere pertinente. Las preguntas que se generan y su respectiva representación en *NLQ_concepts*, se encuentran en la Figura 3-19. En la Figura 3-17 se muestra una captura de pantalla con las preguntas de contexto para que el usuario pueda elegir.

- Variables:
Los parámetros de entrada y la descripción de cada uno se pueden ver en la Tabla 3-2. Estos parámetros son los mismos que utiliza el algoritmo integrado para manejar la anáfora, elipsis y preguntas contextuales.

La variable de salida es *NLQ_concepts*, descrita en la Tabla 3-4, que dentro de todo el sistema puede considerarse como un variable global que almacena y registra los diferentes procesos a los que se ve sometida la consulta.

Como variable local el algoritmo de resolución de la anáfora utiliza *list_antec*, su descripción se encuentra en la Tabla 3-6.

| Variable | Descripción | Tipo |
|------------|--|--------------------|
| list_antec | Lista de antecedentes candidatos para resolución de la anáfora; se obtiene del análisis de NLQ_concepts, previous_questions, previous_answers. | Lista de registros |

Tabla 3-6: Variable local del algoritmo para la resolución de la anáfora.

- Estructuras de datos que utiliza:
Como se pudo ver en la sección anterior, este algoritmo comparte variables con el algoritmo general, por lo que en la Tabla 3-7 se describe la estructura adicional que utiliza este algoritmo: *list_antec*. El resto han sido descritas en la Tabla 3-5.

```

[
  [ // pregunta: "Which traits are affected by QTLs?"
    which
    np (conc (commonName, plural, [Trait], [04616059]), A, n(traits))
    vp (conc (plural, __, [00137313, 00019448, 02677097]), B, v(are), v(affected))
    pp (conc (_, _), C, prep(by), D)
    np (conc (properName, plural, [QTL], __), X*, n(QTLs))
    ?
  ],
  [ // pregunta: "Which traits are affected by QWsv.DiMo-5H.1?"
    which
    np (conc (commonName, plural, [Trait], [04616059]), A, n(traits))
    vp (conc (plural, __, [00137313, 00019448, 02677097]), B, v(are), v(affected))
    pp (conc (_, _), C, prep(by), D)
    np (conc (properName, sing, [QTL], __), X*, n(QWsv.DiMo-5H.1)),
    ?
  ],
  [ // pregunta: "Which traits are affected by QWsv.DiMo-5H.2?"
    which
    np (conc (commonName, plural, [Trait], [04616059]), A, n(traits))
    vp (conc (plural, __, [00137313, 00019448, 02677097]), B, v(are), v(affected))
    pp (conc (_, _), C, prep(by), D)
    np (conc (properName, sing, [QTL], __), X*, n(QWsv.DiMo-5H.2))
    ?
  ],
]
]

```

Figura 3-19: Preguntas y sus respectivas estructuras de huecos en *NLQ_concepts* para la reformulación de la consulta, una vez resuelta la referencia anafórica: “*Which traits are affected by them?*”.

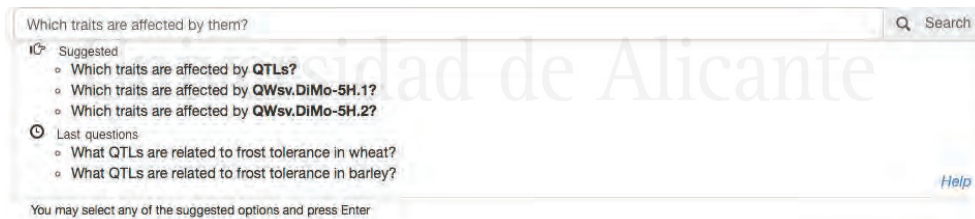


Figura 3-20: Captura de pantalla de la pregunta contextual: “*Which traits are affected by them?*”, que sigue a las preguntas: “*What QTLs are related to frost tolerance in wheat?*”, “*In barley?*”.

| Nombre de la estructura | Descripción | Tipo |
|-------------------------|---|--------------------|
| <i>list_antec</i> | Almacena para cada consulta: -todos las estructuras de huecos que encuentre en NLQ_concepts, previous_questions y previous_answers, que podrían dar solución a la anáfora | Lista de registros |

Tabla 3-7: Estructuras utilizadas por el algoritmo de resolución de la anáfora.

- Código del algoritmo:

En la Figura 3-21 se muestra el código correspondiente al algoritmo para la resolución de la anáfora. El flujo del algoritmo se puede visualizar en el Figura 3-22.


```

ALGORITHM anaphora_ resolution

INPUT
NLQ_concepts: list_of_list_of_slot_structures;
Ont: Ontology;
WN: Semantic_resources;
previous_questions: list_of_list_of_slot_structures;
previous_answers: list_of_list_of_slot_structures;

OUTPUT
NLQ_concepts: list_of_list_of_slot_structures;

VAR
list_antec : list_of_list_of_indexes_of_slot_structures;

BEGIN

list_antec = antecedent_extraction(NLQ_concepts,
previous_questions, previous_answers);

For each anaphor (NLQ_concepts) do
For each anaphor.constraint () do
list_antec =anaphor.constraint (NLQ_concepts, list_antec, On, WN);
For each anaphor.preference () do
list_antec =anaphor.preference (NLQ_concepts, list_antec, On, WN);
End for

NLQ_concepts = generation_of_reformulations (list_antec)

END ALGORITHM

```

Figura 3-21: Código del algoritmo para la resolución de la anáfora.

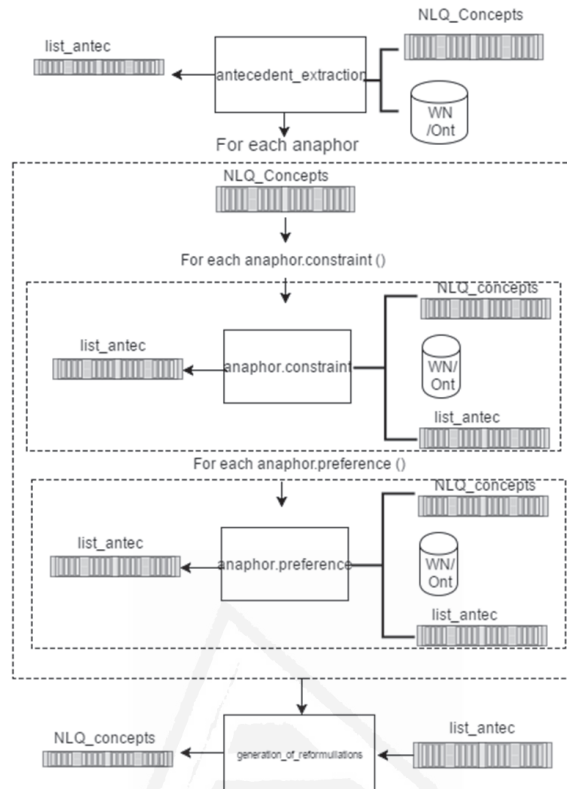


Figura 3-22: Diagrama del algoritmo para resolución de la anáfora.

Algoritmo 3: Algoritmo para la resolución de la elipsis

- **Funcionamiento:**

El algoritmo para la resolución de la elipsis realiza una serie de verificaciones necesarias para realizar el proceso.

La primera puede verse en la condición al inicio del algoritmo cuyo propósito es validar si existe la necesidad o no de la resolución de la elipsis.

Se utiliza el método desarrollado por Ferrández et al., en 2009, para determinar si la consulta está bien formada o, por el contrario, además de la pregunta base existe una segunda pregunta definida en base a la primera. La salida

de este método es booleano y será verdadero si la consulta está bien formada, falso en caso contrario. En la Figura 3-23 se ilustra uno de los patrones sintácticos-semánticos que se utiliza para verificar si una pregunta está bien formada. En ella se observa cómo en la segunda pregunta de la consulta del usuario (que se mostró en la Figura 3-16) se detecta la omisión de una partícula interrogativa, un sintagma nominal y un sintagma verbal (respecto al patrón). Por otra parte, el sistema detecta “*barley*” como una instancia de “*Species*”.

| |
|---|
| Pregunta bien formada: What QTLs are related to frost tolerance in wheat? Patrón: [What] + [NP of QLT type] + [VP] + [set of phrases]? Pregunta con elipsis: What QTLs are related to frost tolerance in wheat? In barley? Patrón: [What] + [NP of QLT type] + [VP] + [set of phrases]? [PP In + [NP of Species type]] |
|---|

Figura 3-23: Ejemplo de patrón sintáctico-semántico utilizado para validar si una pregunta está bien formada. Ejemplo de pregunta bien formada y pregunta con elipsis.

En la Figura 3-24 se observa *NLQ_concepts* para esta frase.

| |
|---|
| Pregunta: What QTLs are related to frost tolerance in wheat? In barley? NLQ_concepts: [pp (conc (_, _), V, prep (in), W) np (conc (commonName, sing, [Species], [07803093, 12123244]), W, n (barley)) ?] |
|---|

Figura 3-24: Ejemplo de estructuras de huecos guardadas en *NLQ_concepts* para segunda frase de la consulta.

Siguiendo con el algoritmo, enunciado en la Figura 3-28, tras la detección de la elipsis se realiza una revisión paralela entre la pregunta anterior y la pregunta actual (utilizando el procedimiento tradicional empleado en QA). Con ello queremos decir que hay términos que deben aparecer de forma obligatoria en la pregunta: un sintagma verbal y sintagmas opcionales categorizados (como la partícula interrogativa a la que le sigue un sintagma nominal que restringe el tipo de información buscada y sintagmas adicionales). Con respecto al paralelismo, se emparejan las estructuras de huecos con aquéllas de la misma categoría sintáctica y semántica.

Para el caso de la pregunta con elipsis de la Figura 3-24, en la segunda frase “*In barley?*”, como se mencionó anteriormente, existe una omisión del sintagma verbal. Este sintagma verbal omitido se recupera de *previous_questions*, de igual manera la partícula interrogativa “*what*” y el sintagma nominal adyacente, que realiza la función de delimitar el objetivo de la consulta: “*QTL*”. Con respecto a la recuperación del resto de estructuras de huecos elididas, en este ejemplo existe una ambigüedad (como se puede ver en la Figura 3-25) para restablecer un *np* o un *pp*.

| | |
|----|---|
| 1- | np (conc (commonName, sing, [Trait], [05033410]), Z, n(frost), n(tolerance), T) |
| 2- | pp (conc (_, _), T, prep (in), U) np (conc (commonName, sing, [Species], [07803093, 12123244]), U, n (wheat)) |

Figura 3-25: Estructuras generadas tras la consulta de *previous_questions* para la pregunta “*In barley?*”

Esta ambigüedad es resuelta debido a la coincidencia de la estructura sintáctica (un sintagma preposicional, *pp* con la preposición “in”) y la categoría semántica (compatibilidad semántica, cuando existe emparejamiento semántico entre las estructuras de huecos; por ejemplo por una relación de hiperonimia o sinonimia). En este caso la coincidencia se dio con [*Species*].

El resultado final que se almacena en *NLQ_concepts* puede verse en la Figura 3-26.

| |
|---|
| <p>Pregunta: What QTLs are related to frost tolerance in wheat? In barley?</p> <p>NLQ_concepts: [what (*) np (conc (properName, plural, [QTL], _), X*, n(QTLs)) vp (conc (plural, _, [00713167, 02724417, 02458103]), Y*, v (are), v (related), prep (to)) np (conc (commonName, sing, [Trait], [05033410]), Z*, n (frost), n(tolerance), T) pp (conc (_, _), V, prep (in), W) np (conc (commonName, sing, [Species], [07803093, 12123244]), W, n (barley)) ? (*)]</p> |
|---|

Figura 3-26: Ejemplo de estructuras guardadas en *NLQ_concepts* para la segunda frase de la consulta tras la resolución de la elipsis.

En la figura se observa que los identificadores con “*” hacen referencia a las frases agregadas, las cuales deben ser presentadas al usuario para que las seleccione.

En la Figura 3-27 se muestra lo que se le presentaría al usuario en el caso de la pregunta que se ha venido analizando.

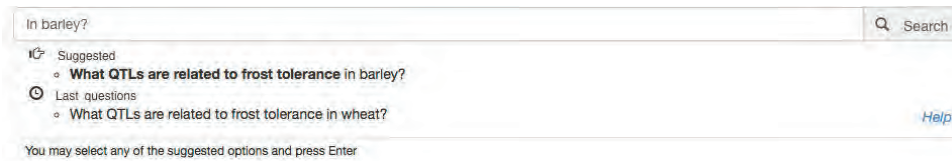


Figura 3-27: Captura de pantalla de la pregunta contextual “*In barley?*”, que sigue a la pregunta: “*What QTLs are related to frost tolerance in wheat?*”

- Variables:

Al igual que el algoritmo anterior, los parámetros de entrada y la descripción de cada uno se pueden ver en la Tabla 3-2. Estos parámetros son los mismos que utiliza el algoritmo integrado para manejar la anáfora, elipsis y preguntas contextuales.

La variable de salida es *NLQ_concepts*, descrita en la Tabla 3-4, que dentro de todo el sistema puede considerarse como un variable global que almacena y registra los diferentes procesos a los que se ve sometida la consulta.

- Estructuras que utiliza:

Al igual que el algoritmo anterior, comparte variables con el algoritmo general, la descripción de las mismas se encuentra en la Tabla 3-5.

- Código del algoritmo:

El código del algoritmo de la resolución de la elipsis se enuncia en la Figura 3-28.

```

ALGORITHM ellipsis_resolution

INPUT
NLQ_concepts: list_of_list_of_slot_structures;
Ont: Ontology;
WN: Semantic_resources;
previous_questions: list_of_list_of_slot_structures;
previous_answers: list_of_list_of_slot_structures;

OUTPUT
NLQ_concepts: list_of_list_of_slot_structures;

VAR
list_antec : list_of_list_of_slot_structures;

BEGIN

If not QA_complete_sentece (NLQ_concepts);
For each phrase (NLQ_concepts, previous_questions) do
If (previous_questions.phrase isA participle and
NLQ_concepts.phrase isNotA participle);
add_phrase (NLQ_concepts, previous_questions.phrase);
If (previous_questions.phrase isA vp and
NLQ_concepts.phrase isNotA vp)
add_phrase (NLQ_concepts, previous_questions.phrase);
If (previous_questions.phrase isA np and
NLQ_concepts.phrase isNotA np and
not compatible(NLQ_concepts, previous_questions.phrase))
add_phrase (NLQ_concepts, previous_questions.phrase);
If (previous_questions.phrase isA pp and
NLQ_concepts.phrase isNotA pp)
add_phrase (NLQ_concepts, previous_questions.phrase);
End for
End if

END ALGORITHM

```

Figura 3-28: Código del algoritmo para manejar la resolución de la elipsis.

3.3. *Caso de estudio*

La presente sección desarrollará un caso de estudio en el cual se ejecuta el sistema propuesto y se desarrolla en 4 subsecciones: descripción general del caso de estudio,

generación de la ontología fundamentada en la base de datos CEREALAB, análisis de una serie de preguntas de contexto procesadas por medio del sistema propuesto y algunos ejemplos de cuadros de mando generados en las pruebas realizadas.

3.3.1 Descripción del caso de estudio

El escenario propuesto para la prueba aquí documentada está basado en una empresa fitomejoradora (que trabaja en la mejora de plantas) que desea llevar a cabo nuevos programas de mejoramiento a través de experimentos con los nuevos avances en Genética.

Utilizamos el *framework* propuesto por Peral et al., en 2015, en el que el tomador de decisiones del programa de cría puede acceder fácilmente a los datos externos sobre los rasgos agronómicos y elaborar protocolos moleculares para diseñar nuevos cultivos modificados genéticamente, con el fin de aumentar la productividad de la industria de las semillas.

Como se describió en la sección anterior, nuestra herramienta de apoyo al autor, se ocupa de la resolución de la anáfora y elipsis y de preguntas de contexto. Esto está integrado en el módulo de interfaz gráfico de usuario de este marco. En nuestro ejemplo concreto, el usuario plantea las preguntas en Lenguaje Natural para consultar la base de datos CEREALAB (Milc et al., 2011). La base de datos CEREALAB almacena datos genotípicos y fenotípicos obtenidos por el proyecto CEREALAB que han

integrado fuentes de datos ya existentes, con el fin de crear una herramienta para los criadores y genetistas de plantas.

Esta base de datos puede ayudar a desentrañar la genética de rasgos fenotípicos que tiene además una importancia económica. Apoya también en la identificación y la elección de marcadores moleculares asociados a los rasgos clave y en la elección de los parentales deseados para los programas de mejoramiento.

La base de datos se divide en tres subesquemas que corresponden a las especies de interés: el trigo, la cebada y el arroz. Cada subesquema se divide en dos subontologías, respecto a los datos genotípicos y fenotípicos, respectivamente. La decisión de utilizar CEREALAB, a pesar de la existencia de algunas bases de datos diseñadas para almacenar y gestionar tanto los datos fenotípicos y genotípicos, como AppleBreed (Antofie et al., 2007) o PlantDB (Exner et al., 2008), entre otros, se debe a que esas bases de datos están diseñadas para almacenar los datos experimentales y los datos disponibles son generalmente restringidos a los llevados a cabo por los desarrolladores/usuarios y no dan la posibilidad de aprovechar la información ya disponible que reside en otras fuentes de datos. Por otra parte, CEREALAB es la primera base de datos específica para la cría de trigo, la cebada y el arroz destacando que estos cultivos son fundamentales en la agricultura mundial en la actualidad.

El modelo correspondiente para el escenario se muestra en la Figura 3-29. Está basado en un perfil de UML para el modelado de DWs presentados en Luján-Mora et al.

(2006). Éste captura la estructura de la información inicial a analizar.

Se distinguen cuatro dimensiones diferentes en el modelo: *DNA Sequence*, *Trait*, *Species* y *Effects*.

- En primer lugar, la dimensión de *DNA Sequence* captura la información con respecto a los *QTLs* y los genes involucrados en los diferentes rasgos mostrados por las diversas especies de plantas. La dimensión *DNA Sequence* está compuesta por tres niveles de jerarquía, cada uno de ellos identificado por el código científico correspondiente o el nombre dado al elemento. Los *QTLs* y genes se agrupan en sus cromosomas correspondientes que representan el nivel más alto de agregación.
- La dimensión *Trait* captura los rasgos afectados por la presencia de los *QTLs*/genes. Los rasgos se identifican por el código asignado a ellos. Algunos ejemplos pueden ser "Resistencia a la helada" o "Granos por espiga". Los rasgos también pueden tener una descripción y pueden estar relacionados con otros rasgos, capturado por medio del atributo

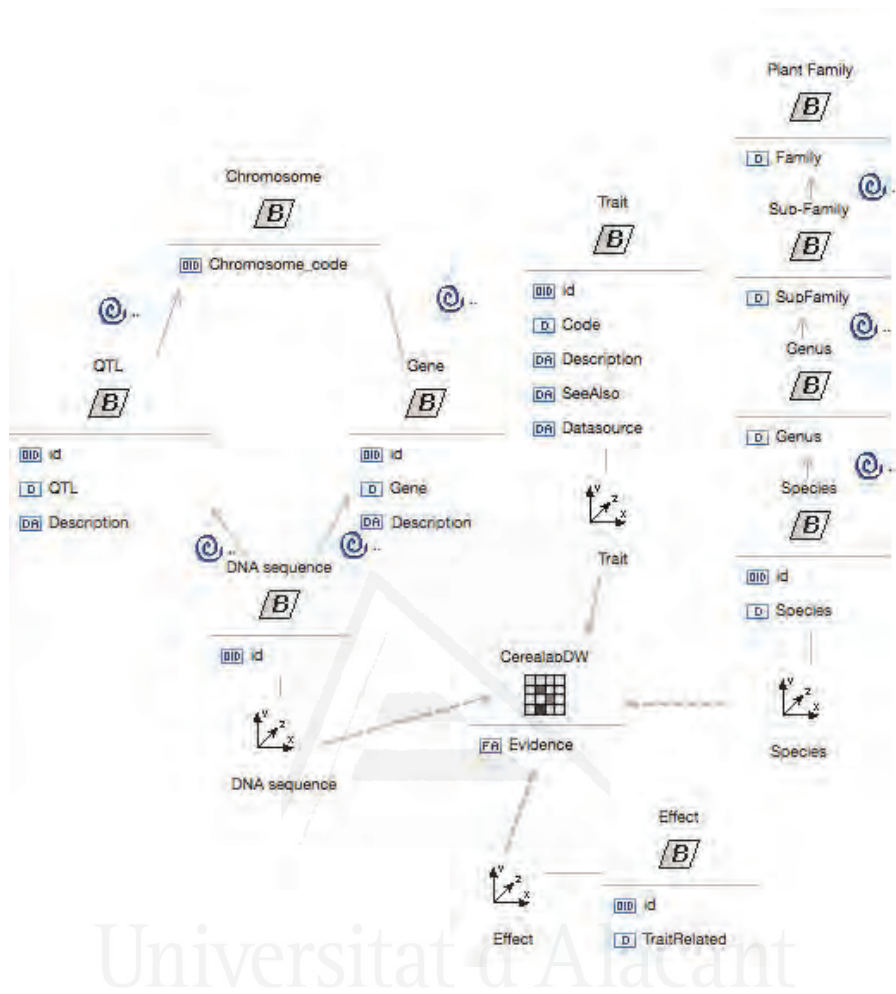


Figura 3-29: Extracto del modelo multidimensional para el escenario CEREALAB

“*SeeAlso*”. Por último, si el rasgo se ha extraído de una fuente de datos, se almacena en el atributo “*DataSource*”.

- La dimensión *Species* captura la información sobre las variedades de plantas que tienen los *QTLs/Genes*. Esta dimensión contiene toda la

información acerca de cada variedad, incluyendo el género, subfamilia y la familia. Cada uno de estos niveles incluye el identificador del grupo al que pertenece la variedad. En nuestro caso, sólo se almacena información sobre trigo, cebada y arroz en el nivel más alto de la jerarquía, aunque podría añadirse información adicional referente a otros grupos.

- La dimensión *Effect* captura el efecto que un determinado *QTL/Gene* tiene sobre un rasgo de una especie (*Trait of a Species*). La razón para incluir esta dimensión de manera separada se debe a que la mayoría de las tecnologías de *DW* están diseñadas para contener valores numéricos dentro de un hecho. El efecto de los *QTL/Genes* en los rasgos puede ser muy variado, como cambiar los colores o cambiar los porcentajes de ciertos elementos químicos presentes en las plantas, por lo que se ha añadido esta dimensión para almacenar esta información. Por último, nuestro hecho incluye una medida que da una idea de cuánta evidencia hay en términos de la cantidad de estudios que apoyan el efecto de un *QTL/Gen* en el rasgo de una especie vegetal. Esta información se obtiene a partir de los datos internos de la empresa. La medida se agrega con la adición de la evidencia encontrada sobre el rasgo que se ve afectado por un *QTL/Gen*.

3.3.2 Generación de la ontología

En esta subsección se ilustra la generación de la ontología con base en la base de datos utilizada en este caso de estudio CERELAB.

La generación de la ontología, como se explicó en la subsección 3.2.2.3, se realiza en 5 pasos y su resultado se muestra en la Figura 3-30.

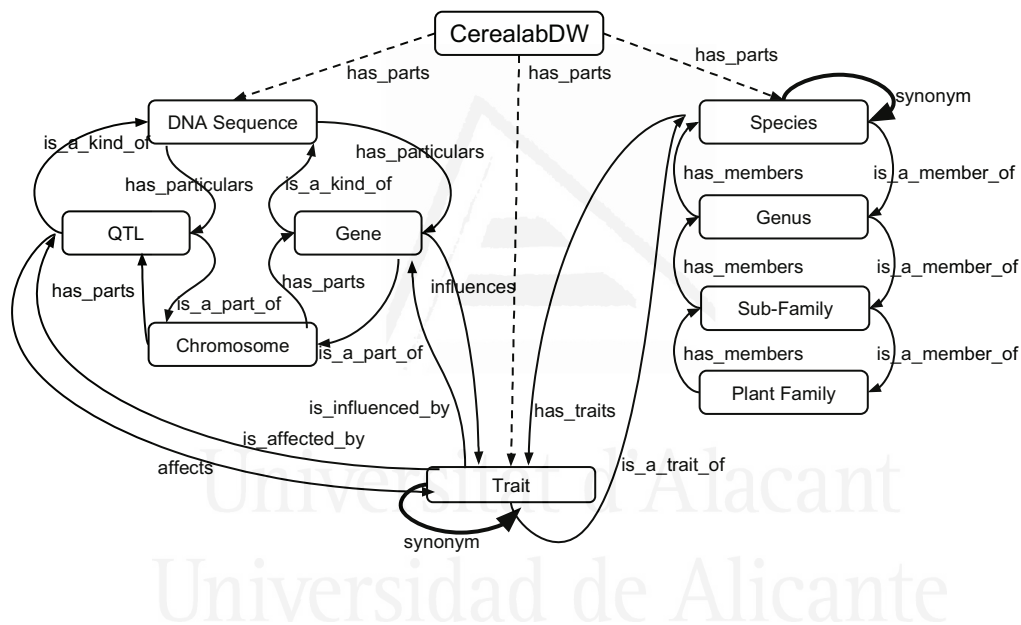


Figura 3-30: Ontología correspondiente a la base de datos CERELAB

Los pasos son los siguientes:

1. La dimensión de la secuencia de DNA se transforma en los conceptos de: DNA Sequence, Gene, QTL y Chromosome; la dimensión Species se transforma en los conceptos de Species, Genus,

Sub-Family y Plant Family; del mismo modo se crea el concepto Trait.

2. En el segundo paso el nodo CereabDW se relaciona por medio de la relación “*has_members*” con los niveles más finos de granularidad de las 3 dimensiones: *DNA Sequence*, *Species* y *Trait*.
3. Durante la tercera etapa, se incluyen en la ontología todas las relaciones “*drill down*” y “*roll up*” como relaciones “*has_members*” y relaciones “*is_a_member_of*” respectivamente; por ejemplo, se agrega el enlace “*is_a_member_of*” de *Species* a *Genus* y uno “*has_members*” de *Genus* a *Species*.
4. En la cuarta etapa, el diseñador cambia el nombre de las relaciones creadas (si procede) y enriquece la ontología mediante la inclusión de las relaciones a través de diferentes dimensiones. Podemos dividir estas relaciones en dos grupos principales, las relaciones estándar (similares a las utilizadas en WordNet) y relaciones específicas (nuevas relaciones especiales para el dominio de la Genética y Biotecnología). En la Tabla 3-8 se muestra un resumen de las relaciones enunciadas en los siguientes párrafos.

Con respecto al primer grupo de relaciones, se incluyen las siguientes relaciones:

- Se utiliza la relación de sinonimia, que permite a un mismo concepto ser expresado en diferentes palabras; como ejemplo los

términos “*durum wheat*” y “*Triticum durum*” son instancias del mismo concepto en *Species*.

- Las relaciones “*has_parts / is_a_part_of*” indican que existe una relación de meronimia; por ejemplo un *Chromosome* consiste en varios *Genes* (*has_parts*) mientras que un *Gene* “*is_a_part_of*” un *Chromosome*.
- Las relaciones “*has_particulars / is_a_kind_of*” se utilizan para especificar relaciones de hiponimia / hiperonimia; como se ilustra con *QTL* y *DNA Sequence*: un *QTL* es un tipo de *DNA Sequence* (*is_a_kind_of*).
- Las relaciones *has_members / is_a_member_of* son utilizadas para mostrar una relación de pertenencia, como se muestra en los nodos relacionados con el nombre de las *Species*; por ejemplo, un *Genus* tiene varias *Species* (*has_members*), mientras que una *Species* “*is_a_member_of*” *Genus*. Ejemplos concretos de esta relación en Cerealab son: los *QTLs* *QWsv.DiMo-5H.1* y *QWsv.DiMo-5H.2* son “*parts of*” del cromosoma 5H en la cebada. Otro ejemplo es que la *Species* *Triticum durum* “*is a member of*” del Genus *Triticum*.

En relación con el segundo grupo de relaciones específicas, para un dominio en particular, se utilizan las siguientes:

- Las relaciones *has_traits* / *is_a_trait_of* son utilizadas para denotar que cada *Species* tiene varios *Traits* (“*has_traits*”).
- *Influences* / *is_influenced_by* se usa para determinar que un *Gene* influye de manera decisiva en un *Trait*.
- La relaciones *affects* / *is_affected_by* es similar a la anterior pero más apropiada para el término *QTL*.

Ejemplos de esas relaciones son:

La *Species* *Hordeum vulgare* “*has_traits*” “frost tolerance”, “aleurone colour”, y “1000 kernel weight” entre otras.

Los *Genes* *Dn1*, *Dn2*, *Dn3* y *Dn4* entre otros *influence* the “*resistance to Russian wheat aphid*” *Trait* en el trigo.

Los *QTLs* *AQGD028*, *AQGD029* y *AQGD030* *affect* el color del pericarpio (*Trait*) en el arroz.

| Relaciones basadas en WordNet | | |
|--------------------------------------|--------------------------------------|--|
| Relación | Denotado por | Ejemplo |
| sinonimia | Varias palabras un mismo concepto | Instancias de <i>Species</i> “ <i>durum wheat</i> ” y “ <i>Triticum durum</i> ” |
| meronimia | <i>has_parts / is_a_part_of</i> | un <i>Chromosome</i> consiste en varios <i>Genes</i> |
| hiponimia / hiperonimia | <i>has_particulars/ is_a_kind_of</i> | <i>QTL</i> es un tipo de <i>DNA Sequence</i> |
| pertenencia | <i>members/ is_a_member_of</i> | <i>QTLs QWsv.DiMo-5H.1</i> y <i>QWsv.DiMo-5H.2</i> son “ <i>parts of</i> ” del cromosoma 5H en la cebada |
| Relaciones propias de dominio | | |
| Relación | Denotado por | Ejemplo |
| Rasgos | <i>has_traits/is_a_trait_of</i> | <i>Species Hordeum vulgare</i> “ <i>has_traits</i> ” “ <i>frost tolerance</i> ” |
| Influencia | <i>Influences / is_influenced_by</i> | <i>Genes Dn1, Dn2, Dn3, y Dn4</i> ... <i>influence the “resistance to Russian wheat aphid”</i> |
| Afecta | <i>affects / is_affected_by</i> | <i>QTLs AQGD028, AQGD029, y AQGD030</i> <i>affect</i> the color de pericarpio en el arroz |

Tabla 3-8: Resumen de las relaciones para creación de la ontología

La ontología creada se formalizó utilizando *Web Ontology Language* (OWL), siguiendo las recomendaciones de W3C (Dean & Schreiber, 2004; Patel-Schneider, Hayes & Horrocks, 2004). Utilizamos Protégé 4 para crear las ontologías (<http://protege.stanford.edu/>).

3.3.3 Serie de preguntas contextuales

En este apartado, se muestra la aplicación de nuestra propuesta (ver Algoritmo 1, sección 3.3.1) a una serie de preguntas. La propuesta se ha implementado basada en el interfaz desarrollado en Llopis et al. (2012). Por otra parte, para la resolución de la anáfora se utilizó la herramienta que se propuso en Palomar et al. (2001) y Muñoz et al. (2000), que actúa sobre la salida de un etiquetador, TreeTagger para el idioma inglés y Maco + para el idioma español, y del analizador SUPAR (Ferrández et al., 1999).

Esta herramienta de resolución de la anáfora logra una precisión del 81% en los pronombres en español, 74% en los pronombres en inglés y 78% en las descripciones definidas. Estas cifras se basan en los resultados obtenidos al aplicar esta resolución en los corpus de propósito general.

Con respecto a la aplicación del *framework* de Peral et al. (2015), éste utiliza la plataforma de BI de código abierto *Pentaho*, que provee las herramientas para trabajar con *OLAP*, por medio del servidor *Mondrian OLAP*. El servidor *OLAP* se conecta a un servidor *MySQL 5.6 DBMS* que almacena los datos para el análisis.

La serie de preguntas propuestas fue planteada por dos expertos en Biotecnología y Agricultura, que tenían conocimiento previo sobre la base de datos CEREALAB. Se puede ver la Figura 3-31, donde están enunciadas las consultas.

- | | |
|----|--|
| a) | 1. Tell me the QTLs related to awn colour in wheat. |
| | 2. And to glume colour? |
| | 3. To semolina one? |
| b) | 1. Does any QTL affect on days to flowering in wheat? |
| | 2. In which chromosome are they located? |
| c) | 1. I want to increase grains per spike in barley. Which QTLs are related to this? |
| | 2. What other traits are affected by them? |
| d) | 1. I would like to know the QTLs related to rust in wheat |
| | 2. Which genes influence the resistance to plagues in wheat? |
| | 3. Are there other plagues in wheat? |
| e) | 1. What QTLs are related to the kernel weight in barley? |
| | 2. Are there any biotechnology companies that have made Genetic Engineering with these QTLs in barley? |
| | 3. Are there currently any transgenic barley varieties in the market? |
| f) | 1. What QTLs are related to frost tolerance and resistance to fusarium in barley? |
| | 2. Since there is no common QTL, are there any studies showing these traits are related? |
| g) | 1. Are there currently any transgenic wheat? |
| | 2. Is it in the market? |
| | 3. What is its price? |

Figura 3-31: Serie de preguntas propuestas para la aplicación de la propuesta.

A continuación se detalla el proceso de resolución, utilizando la propuesta.

Proceso de resolución de la serie de preguntas a)

En la Figura 3-32 se presenta el contenido de la estructura *parsed_sentence*, que se genera luego de realizar un análisis sintáctico parcial a la consulta a).1: “*Tell me the QTLs related to awn colour in wheat*”.

```

Consulta a).1:
“Tell me the QTLs related to awn colour in wheat”

[ vp (conc (_, X, v(tell))
  pron (conc (firstPerson, sing), Y, me)
  np (conc (properName, plural), Z, determiner(the), n(QTLs))
  vp (conc (_, U, v(related), prep(to))
  np (conc (commonName, sing), V, n(awn), n(colour))
  pp (conc (), W, prep(in), A)
  np (conc (commonName, sing), A, n(wheat))
  ?
]

```

Figura 3-32: Contenido de la estructura *parsed_sentence* luego de la aplicación del análisis sintáctico parcial a la consulta a).1.

Como se puede observar en la Figura 3-32, luego del análisis sintáctico parcial se obtienen 3 sintagmas nominales:

- “the *QTLs*”
- “awn colour”
- “wheat”

No hay ninguna anáfora pronominal de tercera persona para resolver. Hay una descripción definida (“los *QTLs*”) que no pueda ser resuelta, ya que es el primer NP, de la primera consulta, por esto no hay antecedentes posibles a los cuales se pueda hacer referencia. En el mapeo de conceptos con la ontología estas 3 estructuras se corresponden con el nodo de la ontología “*QTL*”, una instancia del nodo “*Trait*” y una instancia de “*Species*” respectivamente.

La consulta no contiene una elipsis, por lo tanto no es necesario reformularla.

La consulta es enviada al distribuidor/integrador, que devuelve la respuesta: QRaw.ipk-1A, QRaw.ipk-1D.

Para la pregunta a).2: “And to glume colour?”, luego del análisis sintáctico parcial:

- “to glume colour” es etiquetado como *Trait*.
- No se detecta anáfora.
- Se detecta una elipsis, ya que la consulta no contiene un verbo:
 - Se resuelve mediante la comparación de las estructuras de la pregunta actual con las de la anterior. Los conceptos que faltan son “*QTL*” y “*Species*”.
 - La frase se completa teniendo en cuenta la estructura sintáctica de la frase anterior (es decir, el concepto que no se restaura a partir de la frase anterior es el que coincide con alguno de la frase actual: “*awn colour*” vs “*glume colour*”).
 - La frase propuesta luego de la revisión es: “*Tell me the QTLs related to glume colour in wheat?*”
- La respuesta que se presenta al usuario es: *QRg.ipk-1D*.

Con respecto a la pregunta a).3: “*To semolina one?*”:

- Se analiza sintácticamente la frase.
- Se encuentra el sintagma nominal “*semolina one*”.
- Se detecta la necesidad de aplicar resolución de la anáfora:
 - La anáfora se resuelve y se sustituye por su antecedente “*semolina colour*”, debido a la similitud semántica (*Trait*) entre “*semolina*” y “*glume*”.
- Al igual que la anterior consulta, ésta no contiene un verbo, por lo tanto se detecta la existencia de una elipsis.

Se realiza la comparación con la frase anterior y se detecta que hacen falta los conceptos:

- “QTL”
- “Species”
- Con la estructura sintáctica similar a la frase anterior, la consulta reformulada es: “Tell me the QTLs related to semolina colour in wheat”.
- La respuesta para el usuario es: *QY.ucw-1B*, *QY.ucw-6A*, *7A-QY.ucw*, *QY. UCW-7B*.

Proceso de resolución de la serie de preguntas b)

En la primera pregunta: “*Does any QTL affect on days to flowering in wheat?*”:

- Se identifican 3 NP:
 - “*QTL*”
 - “*days to flowering*”
 - “*wheat*”
- No se encuentra ninguna expresión anafórica.
- Los conceptos *QTL*, una instancia de *Trait* y una instancia de *Species* se han asociado para cada NP respectivamente.
- No se detecta la existencia de elipsis.
- Se entrega al Distribuidor/Integrador la respuesta: *QFlt.ipk-3A*, *QFlt.wak-2D*.

En la pregunta b).2: “*In which chromosome are they located?*”, una vez se realiza el análisis:

- El módulo de resolución de la anáfora detecta un pronombre de tercera persona, plural.

- El antecedente se busca en los conceptos de la oración actual, la anterior y las respuestas de la pregunta anterior.
- Ésta es una anáfora muy ambigua, por lo tanto la lista de posibles antecedentes es amplia y está encabezada por el único NP en plural: “days”, seguido por el resto de los NPs según probabilidad de certeza (muy influenciada por la proximidad a la anáfora).
- No se detecta elipsis.
- Por último, el sistema genera las preguntas según los antecedentes elegidos como solución, de las que el usuario ha seleccionado:
 - “*In which chromosome are QFlt.ipk-3A located?*”
 - “*In which chromosome are QFlt.wak-2D located?*”

Proceso de resolución de la serie de preguntas c)

De las preguntas de la serie (c), el usuario desea conocer si los QLTs que afectan un rasgo (granos por espiga) pueden tener efectos colaterales sobre otros rasgos, por ejemplo el peso del grano.

Para la primera consulta: “*I want to increase grains per spike in barley. Which QTLs are related to this?*”:

- El sistema localiza “this” y lo clasifica como un pronombre de tercera persona singular. Esto es un caso especialmente difícil de resolver ya que el antecedente no es un NP sino un sintagma verbal. En este caso, el usuario debe resolver de forma manual a través de la herramienta de apoyo a la

edición, reformulando la pregunta como: “*Which QTLs are related to increase grains per spike in barley?*”

- En la siguiente etapa, el sistema no encuentra un nodo en la ontología que coincida con el NP “*grains per spike*”. Por su parte el etiquetado del NP usando WordNet presenta como sinónimo el concepto “*kernels per spike*”, que se encuentra en la ontología como una instancia de “*Trait*”.
- Los restantes NPs se identifican en la ontología.
- No se presenta elipsis.
- Las respuestas que se obtienen son: : *QKer.pil-1H*, *QKps.BIKy-1H*, *QKps.BIKy-3H*, *QKps.BIKy-4H*, *QKps.BIKy-5H*, *QKps.BIKy-6H*, *QKps.TyVo-2H*.

En lo referente a la pregunta c).2, “*What other traits are affected by them?*”:

- Después de la ejecución del análisis sintáctico parcial, el pronombre “ellos” se resuelve y se sustituye por sus posibles antecedentes. Para la generación de antecedentes se toma en cuenta preguntas y respuestas anteriores. Luego se identifican todos los conceptos.
- La pregunta no contiene elipsis.
- Se reformula la pregunta, con base en cada respuesta simple de las preguntas previas:
 - “What other traits are affected by QTLs?”
 - “What other traits are affected by QKer.pil-1H?”
 - “What other traits are affected by QKps.BIKy-1H?”
 - “What other traits are affected by QKps.BIKy-3H?”, etc.

Proceso de resolución de la serie de preguntas d)

Continuando con la ejecución, para la consulta d).1, “*I would like to know the QTLs related to rust in wheat*”:

- Se detectan 3 NPs:
 - “*QTLs*”
 - “*rust*”
 - “*wheat*”
- No se encuentra relación anafórica.
- Todos los conceptos coinciden con un solo nodo simple de la ontología, excepto “*rust*”, que coincide con 3 instancias de “*Trait*”:
 - “*resistance to leaf rust*”
 - “*resistance to stem rust*”
 - “*resistance to stripe rust*”
- No se encuentra alguna relación de elipsis.
- Se sugiere al usuario una consulta por cada instancia de “*Trait*”:
 - “*I would like to know the QTLs related to resistance to leaf rust in wheat.*”
 - “*I would like to know the QTLs related to resistance to stem rust in wheat.*”
 - “*I would like to know the QTLs related to resistance to stripe rust in wheat.*”

Para la consulta d).2, “*Which genes influence the resistance to plagues in wheat?*”:

- Se analizan 3 NPs.
- No se detecta la presencia de alguna anáfora.

- Los conceptos “Genes” y “wheat” son localizados en la ontología; “plagues” no se encuentra.
- Luego de la extensión por medio de WordNet del término no encontrado, se identifica “*group/swarm of insects*” como sinónimo de “plague”, y esto implica que es un “Trait” de *wheat*.
- Utilizando las relaciones de WordNet y los conceptos de la ontología, el sistema determina que los rasgos:
 - “resistance to Russian wheat aphid”
 - “resistance to hessian fly”
 hacen referencia a plagues porque “aphid” y “fly” son insectos.
- No se detecta ninguna elipsis.
- Se sugiere una consulta por cada “Trait”:
 - “*Which genes influence the resistance to Russian wheat aphid in wheat?*”
 - “*Which genes influence the resistance to hessian fly in wheat?*”

La consulta d).3, “*Are there other plagues in wheat?*” se resuelve como las consultas anteriores. Se debe destacar que las preguntas se clasifican siempre como la búsqueda de conceptos marcados por el primer NP en la pregunta, en vez de de ser una pregunta cuya respuesta es sí/no. Además, es importante conocer que dentro del marco utilizado (Peral et al., 2015), el usuario puede decidir si desea consultar datos externos con el fin de responder a la pregunta. En ese caso, el nodo de QA de la arquitectura propuesta es el responsable de proporcionar la respuesta.

Proceso de resolución de la serie de preguntas e)

Referente a la serie de preguntas e), el usuario solicita información acerca de los competidores (datos externos) sobre un tema específico. Para la segunda consulta: “*Are there any biotechnology companies that have made Genetic Engineering with these QTLs in barley?*”, una vez realizado el análisis sintáctico parcial:

- Los 4 NPs detectados son:
 - “*biotechnology companies*”
 - “*Genetic Engineering*”
 - “*these QTLs*”
 - “*barley*”.
- Se realiza la resolución de la anáfora para la expresión “*these QTLs*” y se reemplaza por la respuesta de la pregunta previa: *QGwe.HaTR-5H.1*, *QGwe.HaTR-5H.2*, *QGwe.HaTR-7H.1*, etc.
- Se reformula la pregunta con base en cada respuesta:
 - “*Are there any biotechnology companies that have made Genetic Engineering with QGwe.HaTR-5H.1 in barley?*”
 - “*Are there any biotechnology companies that have made Genetic Engineering with QGwe.HaTR-5H.2 in barley?*”
 - “*Are there any biotechnology companies that have made Genetic Engineering with QGwe.HaTR-7H.1 in barley?*”, etc.
- El sistema no puede obtener estas respuestas en la base de datos CEREALAB por lo que se hace necesario que el usuario decida si desea consultar alguna fuente de información externa.

Proceso de resolución de la serie de preguntas f)

En la serie f) el usuario busca relaciones entre diferentes rasgos fenotípicos. Por ejemplo, si se busca que la cebada sea resistente a las heladas, se podría encontrar que también sea resistente al fusarium.

Para la primera pregunta de la serie: “*What QTLs are related to frost tolerance and resistance to fusarium in barley?*”, se detectan 2 NPs, que son coincidentes con instancias del nodo “*Trait*”: “*frost tolerance*” y “*resistance to fusarium*”.

Proceso de resolución de la serie de preguntas g)

En la serie de consultas g), el usuario desea información sobre los competidores, en específico del precio de productos, con el fin de determinar precios competitivos para sus productos. En estas preguntas, el usuario está buscando el precio del trigo transgénico. El sistema es incapaz de encontrar este tipo de información en la base de datos CEREALAB, por lo tanto el usuario tiene que decidir si consulta una fuente de datos externa que contenga esa información.

Con el fin de probar la robustez y la utilidad de la propuesta se pidió a los dos expertos ampliar el conjunto inicial de preguntas. De esta solicitud se obtuvo un conjunto de 149 preguntas (detalladas en el Anexo 1). Para la evaluación se utilizaron diferentes medidas que permiten calcular el rendimiento de la propuesta. Se combinaron dos tipos de medidas: las medidas clásicas utilizadas en las conferencias iCLEF hasta el año 2005

para calcular la precisión de un sistema y medidas centradas en el usuario, utilizadas en iCLEF desde 2006. La elección de estas medidas se ha realizado de este modo porque esta competición está centrada en la perspectiva del usuario aplicada a las capacidades de búsqueda, simulando la interacción con el usuario que proponemos en nuestro trabajo.

Por parte de las medidas clásicas, se utiliza la medida "accuracy" (Gonzalo et al., 2006) Se obtuvo un 81.4% de precisión. Se detectó que los principales errores se debieron a la detección incorrecta de los conceptos de la consulta.

La medida se calcula como la fracción de consultas de las cuales el usuario obtiene la información solicitada. Se cuenta con un lapso de 3 minutos. Si el usuario dentro de ese tiempo no obtiene la consulta correcta, se considera que no fue resuelta, similar a lo aplicado en iCLEF 2004 y 2005. Se toman en cuenta sólo si los resultados de la consulta son correctos. Es importante señalar que esta evaluación no midió la resolución de la anáfora y elipsis.

En relación con las medidas centradas en el usuario se utilizaron las medidas de satisfacción del usuario y las medidas que conciernen a la reformulación de las consultas. Para medir la satisfacción del usuario se aplicó una encuesta (basada en iCLEF 2006). Cada pregunta se puede puntuar entre 1 y 5 según su nivel de satisfacción, donde 1 es el puntaje menor y 5 el mayor. El resultado obtenido fue de 4.3 de satisfacción del usuario, de los 5 puntos que se podían obtener.

La encuesta mide:

- Feliz:
 - ¿Está satisfecho con la manera en que se realizó la tarea?

- Completo:
 - ¿Está satisfecho con lo encontrado, o hubiese continuado si no existiera el límite de tiempo?
- Calidad:
 - Comparando las propuestas del sistema con un conjunto dado. ¿Alguno de los mostrados es mejor que los resultados recuperados?

Basados en lo propuesto en el iCLEF 2009 (Gonzalo et al., 2010) por el Swedish Institute of Computer Science (SICS), se utilizaron diferentes medidas relacionadas con la reformulación de la consulta. En la Tabla 3-9 se detalla cada una y los resultados obtenidos en la ejecución de la propuesta.

| Medida | Descripción | Resultado |
|---|---|--|
| Exactitud | Fracción de preguntas para las que hay al menos una reformulación correcta ofrecida por nuestro sistema. | 75.6% Los errores principalmente se dieron a fallos en la resolución de la anáfora o elipsis. |
| Número medio de reformulaciones | Mide el promedio de reformulaciones que se realizan por consulta. | 1.4 reformulaciones por pregunta 71.1% de las preguntas fueron reformuladas correctamente en la primera iteración. |
| Tiempo promedio utilizado para responder una consulta | Mide el tiempo promedio utilizado para responder a una consulta; toma en cuenta el tiempo utilizado por el sistema para hacer reformulaciones de la consulta. | 35.7 segundos por pregunta. |

Tabla 3-9: Medidas utilizadas para evaluar el proceso de reformulación de la consulta en el sistema propuesto

3.3.4 Cuadro de mando presentado al usuario

Este apartado tiene como propósito presentar algunos ejemplos de los cuadros de mando generados en las pruebas del sistema. En la parte superior de la Figura 3-33 y Figura 3-34 se muestra la

herramienta de apoyo al usuario, que está organizada de la siguiente manera:

- La pregunta actual, propuesta inicial del usuario (por ejemplo:
 - En la Figura 3-33: “*To semoline one?*”
 - En la Figura 3-34: “*In which chromosome are they located?*”
- Las preguntas sugeridas para la reformulación, generadas luego de aplicar la resolución de la anáfora y elipsis.
- Las últimas preguntas planteadas por el usuario.

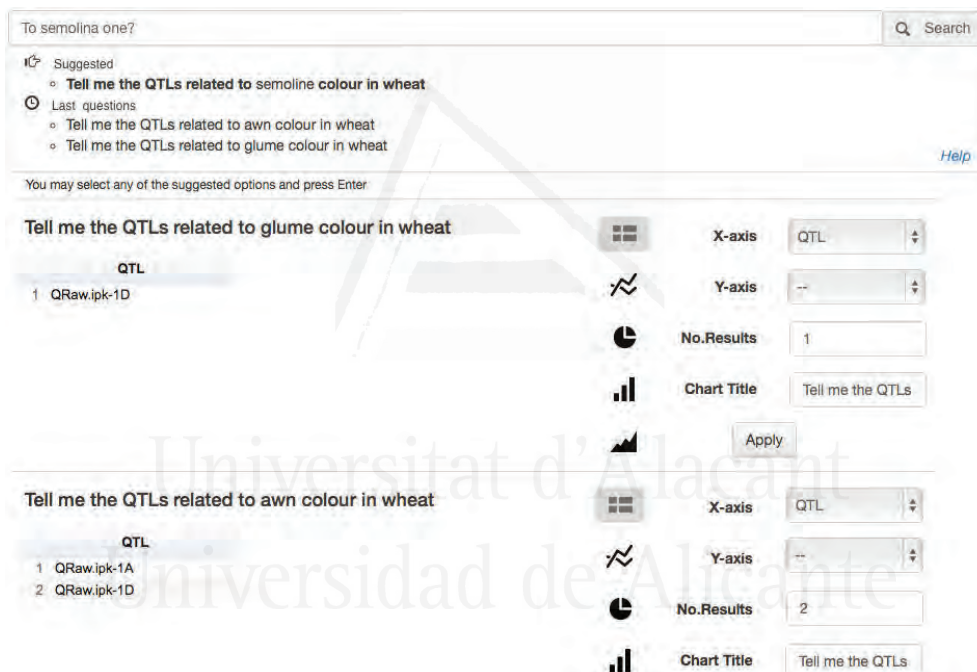


Figura 3-33: Cuadro de mando para la serie de preguntas a), de la sección anterior

Como se puede visualizar en la parte inferior de estas figuras, se presentan los cuadros de mando con la solución obtenida del resultado de las preguntas anteriores de la serie. Para cada pregunta anterior planteada por el usuario hay un panel de control.

Cada cuadro de mando tiene como título la pregunta planteada por el usuario y luego muestra la información extraída de los nodos origen (datos internos de la base de datos, o datos externos desde la web). Abajo, en la parte derecha, se presentan detalles adicionales sobre la información extraída.

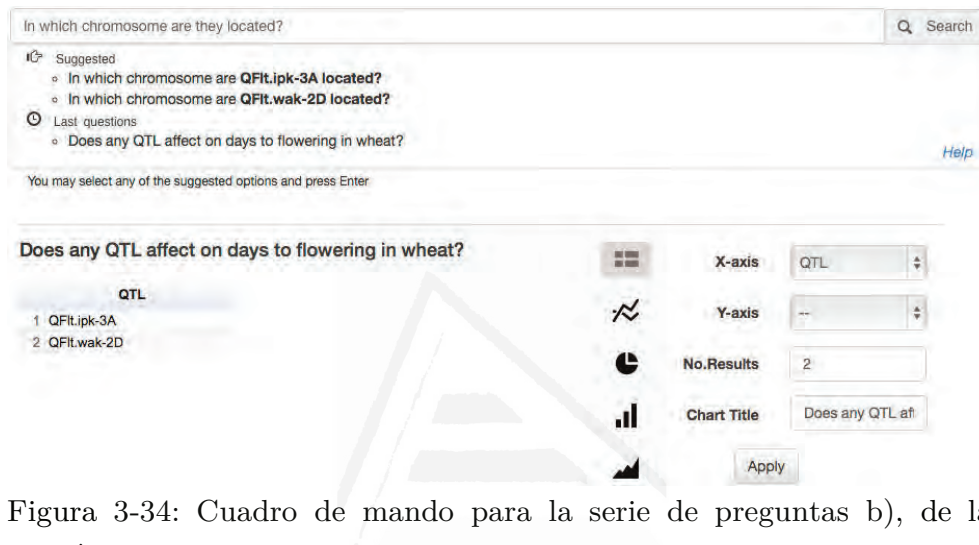


Figura 3-34: Cuadro de mando para la serie de preguntas b), de la sección anterior.

4. Conclusiones y trabajos futuros

Acceder a la información de manera efectiva es un reto de gran relevancia en un mundo donde el volumen de información que se genera crece de manera exponencial. Este acceso no trata sólo de contar con esa información sino obtener provecho de la misma.

En el área de investigación y de negocios se genera de igual manera grandes cantidades de hallazgos relevantes. Si se tuviera el acceso oportuno a los mismos, se podrían tomar decisiones acertadas, ya sea para la mejora de una investigación que podría representar un impacto en salud o en mejora de la vida, o para la mejora de un producto o servicio (como para una inversión clave de una organización).

Sin embargo, cuando se trata de acceder a la información, ésta no puede ser accedida de manera espontánea, ya que se presenta la barrera de la heterogeneidad y la diversidad de las fuentes de datos. El reto del acceso a la información ha sido abordado por diferentes propuestas, como hemos visto a través de este trabajo; nuestra propuesta trata específicamente del acceso a través de un interfaz en Lenguaje Natural (NLI).

La propuesta presentada ha extendido el NLI existente que se creó en 2012 por Llopis, et al. Nuestro NLI permite acceder a fuentes de datos heterogéneas. Además que se apoya al usuario en la construcción de la consulta, sin que éste deba tener conocimiento técnico previo, ni se le somete a conocer un lenguaje de consulta específica, sino que por medio del lenguaje natural realiza la consulta y con la herramienta de apoyo al usuario se le ayuda en la construcción de la misma. Este apoyo se logra con la

incorporación de la resolución de problemas lingüísticos diversos, como la anáfora y la elipsis. Además, el sistema es capaz de procesar una serie de preguntas encadenadas (serie de preguntas contextuales) que es la forma natural de proceder de un usuario ante unos requerimientos de información.

El usuario no sólo recibe apoyo en la etapa de la construcción de la consulta, también en la salida de resultados, que se maneja a través de un cuadro de mando donde el usuario obtiene de diferentes fuentes de datos la información de manera estructurada para que pueda manipularla.

El interfaz para su funcionamiento es respaldado por una serie de procesos que interactúan entre las diferentes fuentes. Entre estas fuentes se realizan unos mapeos o equivalencias de modo que se establece una “comunicación” entre ellas y de esta manera se puede responder exitosamente a la necesidad de información presentada por el usuario.

Por otra parte, se logra superar uno de los principales problemas de los NLI: la dificultad de adaptación a nuevos dominios. Esto es resuelto mediante ontologías que se generan de una manera semi-automática. De nuevo esto respalda el objetivo del proyecto de evitar en lo posible la necesidad de la intervención de un usuario con conocimiento técnico previo.

Esta herramienta ha sido probada en un escenario de un Sistema de Toma de Decisiones (DSS) en el área de Biotecnología y agricultura, en donde diferentes usuarios consultan la base de datos CEREALAB a través de una serie de preguntas de contexto, alcanzando una alta satisfacción del usuario en la interacción con

el DSS, lo que impacta positivamente en el proceso de toma de decisiones.

Las preguntas que se utilizaron para probar esta aproximación se plantearon por expertos en Biotecnología y agricultura que tenían conocimiento previo sobre la base de datos CERREALAB.

El proceso de prueba consistió:

- Conjunto de preguntas iniciales.
- Ampliación del conjunto de preguntas; se generaron en total 149 preguntas.
- Se cuenta con un lapso máximo de 3 minutos para dar el resultado correcto. Si no se obtiene en ese lapso se toma como fallido el resultado.

Con base en la medida "accuracy" (Gonzalo et al., 2006), se obtuvo un 81.4% de precisión y los principales errores detectados se debieron a la detección incorrecta de los conceptos de la consulta.

Respecto a la tarea de reformulación de la consulta se obtuvo:

- 75% de exactitud (fracción de preguntas para las que hay al menos una reformulación correcta).
- 1.4 reformulaciones promedio por pregunta.
- 71.1% de las preguntas fueron reformuladas en la primera iteración.
- 35.7 segundos en promedio para dar respuesta a una pregunta.

Como se puede observar en los resultados, el sistema presenta un buen rendimiento, ya que presenta una exactitud superior al 80% en la solución correcta de las preguntas. Respecto a la tarea de

reformulación de la consulta, se tiene claro que se debe trabajar en la disminución del tiempo de respuesta, sin embargo es compensando con la correcta reformulación en la primera iteración de más de un 70% de las consultas.

4.1. *Trabajos futuros*

Dentro de los trabajos futuros identificados se encuentran:

- Mejorar el tiempo promedio de generación de la respuesta, esto permite al sistema ser más atractivo para los usuarios.
- Adaptar la conexión a más fuentes de datos. Ya se cuenta con conexión a BD relacionales, DW, QA, sin embargo en Internet siguen creciendo las fuentes de información, como las generadas por redes sociales.

Esta adaptación requiere de un análisis exhaustivo, ya que se generan factores como la temporalidad, la falta de contexto, la informalidad de los textos y la cantidad de mensajes.

Un ejemplo de esto es Twitter, donde los mensajes son de 140 caracteres, el contexto que se genera podría ser una barrera para el análisis de relaciones anafóricas o elípticas.

- Se prevé probar otras técnicas de resolución de la anáfora como por ejemplo el sistema Berkeley en Durett & Klein en 2013.
- Se espera incorporar técnicas de desambiguación de los términos.
- Además como parte del proyecto se incorporará un grupo de la Universidad Hispanoamericana de Costa Rica para hacer crecer el conocimiento del área en este país.

5. Referencias

- Afonso, A., Brito, L. C., & Vale, O. (2008). An Evolutionary Method for Natural Language to SQL Translation. *Lecture Notes in Computer Science*, 432-441.
- Aleksejeva, I. (2014). EU experts' attitude towards use of GMO in food and feed and other industries. *Procedia - Social and Behavioral Sciences*, 110, 494-501.
- Altman, R., Bergman, C., Blake, J., Blaschke, C., Cohen, A., Gannon, F., . . . Valencia, A. (2008). Text mining for biology - the way forward: opinions from leading scientists. *Genome Biology*, 9, 1-15.
- Antofie, A., Lateur, M., Oger, R., Patocchi, A., Durel, C., & Van de Weg, W. (2007). A new versatile database created for geneticists and breeders to link molecular and phenotypic data in perennial crops: the AppleBreed DataBase. *Bioinformatics*, 23(7), 882-891.
- Arampatzis, A., van der Weide, T., Koster, C., & Bommel, v. (2000). Linguistically motivated Information Retrieval. (A. Kent, Ed.) *Encyclopedia of Library and Information Science*, 201-222.
- Chai, J. Y., & Jin, R. (2004). Discourse Structure for Context Question Answering. *Proceedings of the HLT-NAACL 2004: Workshop on Pragmatics of Question Answering*, (págs. 23-30).
- Chaudhari, P. P. (2013). Natural Language Statement to SQL Query Translator. *International Journal of Computer Applications*, 18-22.
- Cimiano, P., & Minock, M. (2009). Natural language interfaces: What is the problem - A data-driven quantitative analysis. *Proceedings of the NLDB*, (págs. 192-206).

- Cimiano, P., Haase, P., Heizmann, J., Mantel, M., & Studer, R. (2008). Towards portable natural language interfaces to knowledge bases - the case of the ORAKEL system. *Data & Knowledge Engineering*, 325-354.
- Clegg, A., & Shepherd, A. (2007). Benchmarking natural-language parsers for biological applications using dependency graphs. *BMC Bioinformatics*, 1-17.
- Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., & Wilks, Y. (2000). Experience of using GATE for NLP R/D. *Proceedings of the Workshop on Using Toolsets References 2000 and Architectures To Build NLP Systems at COLING-2000*, (págs. 1-8).
- Dean, M., & Schreiber, G. (s.f.). *OWL Web Ontology Language Reference*. Obtenido de W3C Recommendation: <http://www.w3.org/TR/owl-ref/>
- Díaz de Ilarraza, A., Rodríguez Hontoria, H., & Verdejo, F. (1990). A Mechanism for ellipsis resolution in dialogued systems. *Proceedings of the 13th conference on Computational linguistics (COLING '90)*, (págs. 452-454).
- Distelhorst, G., Srivastava, V., Rosse, C., & Brinkley, J. (2003). A prototype natural language interface to a large complex knowledge base, the Foundational Model of Anatomy. . *Proceedings of the AMIA Annual Symposium*, (págs. 200-204).
- Elhai, J., Taton, A., Massar, J., Myers, J., Travers, M., Casey, J., . . . Shrager, J. (2009). BioBIKE: a web-based, programmable, integrated biological knowledge base. . *Nucleic Acids*, 28-32.
- EMC, & IDC. (1 de abril de 2014). *Estudio del universo digital de EMC con investigación y análisis por IDC*. Obtenido

- de EMC: <http://mexico.emc.com/leadership/digital-universe/index.htm#2014>
- ETC (Action Group on Erosion, T. a. (2008). Who owns nature? Corporate Power and the Final Frontier in the Commodification of Life. . *Communiqué*, 100, 52.
- Exner, V., Hirsch-Hoffmann, M., Gruissem, W., & Hennig, L. (2008). PlantDB - a versatile database for managing plant research. *Plant Methods*.
- Falconer, D., & Mackay, T. (1996). *Introduction to Quantitative Genetics* (4 ed.). Green, Longmans.
- Ferrandez, A., Peral, J., De Gregorio, E., Trujillo, J., Maté, A., Ferrández, L., & Rojas, Y. (2015). An authoring tool for decision support systems in context questions of ecological knowledge. *Ecological Informatics*, 30, 328-344.
- Ferrández, A., Rojas, Y., & Peral, J. (2007). The successful application of Natural Language Processing for Information Retrieval. *Journal of Computer Science & Technology*, 79-85.
- Ferrández, S., Toral, A., Ferrández, O., Ferrández, A., & Muñoz, R. (2009). Exploiting Wikipedia and EuroWordNet to Solve Cross-Lingual Question Answering. *Information Sciences*, 3473-3488.
- Giampiccolo, D., Forner, P., Herrera, J., Peñas, A., Ayache, C., Forascu, C., . . . Sacaleanu, B. S. (2007). Overview of the CLEF 2007 Multilingual Question Answering Track. *Proceedings of the CLEF 2007 Workshop*, (págs. 200-236).
- Goldsmith, E., Mendiratta, S., Akella, R., & Dahlgren, K. (2009). Natural language query in the biochemistry and molecular biology domains based on cognition search. *Proceedings of the AMIA Joint Summits on Translational Science.*, (págs. 32-37).

- Gonzalo, J., Clough, P., & Vallin, A. (2006). Overview of the CLEF 2005 Interactive Track. *Proceedings of the CLEF 2005 Workshop. LNCS 4022*, (págs. 251-262).
- Gonzalo, J., Peinado, V., Clough, P., & Karlgren, J. (2010). Overview of iCLEF 2009: Exploring Search Behaviour in a Multilingual Folksonomy Environment. *Proceedings of the CLEF 2009 Workshop. LNCS 6242*, (págs. 13-20).
- Gonzalo, J., Verdejo, F., Chugur, I., & Cigarran, J. (1998). Indexing with WordNet Synsets can Improve Text Retrieval. *Proceedings of the Workshop on Usage of WordNet for NLP* (págs. 1-11). ACL.
- Green, W., Chomsky, C., & Laugherty, K. (1961). BASEBALL: An automatic question answerer. *Proceedings of the Western Joint Computer Conference*, (págs. 219-224).
- Grosz, B., & Sidner, C. (1986). Attention, intention, and the structure of discourse. *Computational Linguistics*, 175-204.
- Grosz, B., Joshi, A., & Weinstein, S. (1995). Centering: a framework for modeling the local coherence of discourse. *Computational Linguistics*, 203-225.
- Harabagiu, S., Moldovan, D., Pasca, M., Mihalcea, R., Surdeanu, M., Bunescu, R., . . . Morarescu, P. (2000). FALCON: Boosting Knowledge for Answer Engines. *Ninth Text REtrieval Conference*.
- Harabagiu, S., Moldovan, D., Pasca, M., Surdeanu, M., Mihalcea, R., Girju, R., . . . Bunescu, R. (2001). Answering Complex, List and Context Questions with LCC's Question-Answering Server. *Proceedings of the Tenth Text REtrieval Conference (TREC-10)*, (págs. 35-451).

- Hobbs, J. (1985). *On the coherence and structure of discourse. Technical report no. CSLI-85-37, Center for the Study of Language and Information, Stanford University.*
- ISAAA. (*International Service for the Acquisition of Agri-biotech Applications*). *Pocket K N° 16. . .* (2012).
Obtenido de Global status of commercialized biotech/GM crops.: <http://www.isaaa.org/resources/publications/pocketk/16/default.asp>.
- Jamil, H. (2012). A natural language interface plug-in for cooperative query answering in biological databases. *BMC Genomics.*, 1-12.
- Jensen, L., Saric, J., & Bork, P. (2006). Literature mining for the biologist: from information retrieval to biological discovery. *Nature Reviews Genetics*, 119-129.
- Kato, T., Fukumoto, J., & Masui, F. (2004). Question Answering Challenge for Information Access Dialogue – Overview of NTCIR4 QAC2 Subtask 3. *Proceedings of the NTCIR-4 Workshop Meeting.*
- Lamel, L., Rosset, S., Ayache, C., Mostefa, D., Turmo, J., & Comas, P. (2008). Question Answering on Speech Transcriptions: the QAST evaluation in CLEF. *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, (págs. 28-30).
- Laukaitis, A., & Vasilecas, O. (2007). Natural language as programming paradigm in data exploration domain. *Information Technology and Control*, 30-36.
- Le Buanec, B. (2008). Evaluation of the seed industry during the past 40 years. . *SEED News*.
- Li, Y., Chaudhuri, I., Yang, H., Singh, S., & Jagadish, H. (2007). Enabling domain awareness for a generic natural language interface. *Proceedings of the Association for the*

- Advancement of the Artificial Intelligence (AAAI*, (págs. 833-838).
- Li, Y., Yang, H., & Jagadish, H. (2005). NaLIX: an interactive natural language interface for querying XML. *Proceedings of the SIGMOD*, (págs. 900-902).
- Llopis, M., & Ferrández, A. (2012). How to make a natural language interface to query databases accessible to everyone: an example. *Computational Standard & Interfaces*, 470-481.
- Luján-Mora, S., Trujillo, J., & Song, I. (2006). A UML profile for multidimensional modeling in data warehouses. *Data and Knowledge Engineering*, 725-769.
- Lynch, M., & Walsh, B. (1998). *Genetics and Analysis of Quantitative Traits*. Sunderland, MA: Sinauer Associates, Inc.
- Mann, W., & Thompson, S. (1987). *Rhetorical structure theory: a theory of text organization*. Technical report no. ISI/RS-87-190, Information Sciences Institute, University of Southern California.
- Matos, S., Arrais, J., Maia-Rodrigues, J., & Oliveira, J. (2010). Concept-based question expansion for retrieving gene related publications from MEDLINE. *BMC Bioinformatics*, 1-9.
- Matsuda, M., & Fukumoto, J. (2005). Answering Questions of IAD Task using Reference Resolution of Follow-up Questions. *Proceedings of the Fifth NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access*.
- Meijerink, G., & Danse, M. (2009). *Riding the Wave: High Prices, Big Business? The Role of Multinationals in the International Grain Markets*. Report 2009-031. Project

- code 40789*. Netherland: LEI Wageningen UR, The Hague.
- Milc, J., Sala, A., Bergamaschi, S., & Pecchioni, N. (2011). A genotypic and phenotypic information source for marker-assisted selection of cereals: the CEREALAB database. *Database*.
- Miles, C., & Wayne, M. (2008). Quantitative Trait Locus (QTL) analysis. . *Nature Education*, 208.
- Mitra, M., Buckley, C., Singhal, A., & Cardie, C. (1997). An analysis of statistical and syntactic phrases. *Proceedings of the 5th International Conference "Recherche d'Information Assistee par Ordinateur"*, (págs. 200–214).
- Muñoz-Terol, R., Puchol-Blasco, M., Pardiño, M., Gómez Soriano, J., Roger, S., Vila, K., . . . Martínez-Barco, P. (2008). AliQAn, Spanish QA System at Multilingual QA@CLEF-2008. *CLEF (Working Notes)*.
- Oppenheim, C., Stenson, J., & Wilson, R. S. (2003). Studies on information as an asset I: Definitions. *Journal of Information Science*, 159-166.
- Palomar, M., Ferrández, A., Moreno, L., Martínez-Barco, P., Peral, J., Saiz-Noeda, M., & Muñoz, R. (2001). An Algorithm for Anaphora Resolution in Spanish Text. *Computational Linguistics*, 545-567.
- Palomar, M., Ferrández, A., Moreno, L., Martínez-Barco, P., Peral, J., Saiz-Noeda, M., & Muñoz, R. (2001). An Algorithm for Anaphora Resolution in Spanish Text. . *Computational Linguistics*, 545-567.
- Patel-Schneider, P., Hayes, P., & Horrocks, I. (2004). *OWL Web Ontology Language Semantics and Abstract Syntax*. Obtenido de W3C Recommendation: <http://www.w3.org/TR/2004/REC-owl-semantics-20040210/>

- Peral, J., Ferrández, A., De Gregorio, E., Trujillo, J., Maté, A., & Ferrández, L. (2015). Enrichment of the phenotypic and genotypic DataWarehouse analysis using Question Answering systems to facilitate the decision making process in cereal breeding programs. *Ecological Informatics*, 203-216.
- Popescu, A., Etzioni, O., & Kautz, H. (2003). Towards a theory of natural language interfaces to databases. *Proceedings of the 8th international conference on Intelligent user interfaces (IUI'03)*, (págs. 149-157).
- Rojas, Y., Ferrández, A., & Navarro, J. (2005). Indexing Entities to Improve Information Retrieval Performance. En B. Prasad (Ed.), *Proceedings of the 2nd Indian International Conference on Artificial Intelligence, IICAI 2005*, (págs. 790-809). Pune.
- Rojas, Y., Ferrández, A., & Navarro, J. (2005). Using WordNet to Improve Accuracy in Information Retrieval. (A. Gelbukh, C. Márquez, & O. Camacho, Edits.) *Advances in Artificial Intelligence and Computer Science. Research on Computing Science* (14), 1279-190.
- Rojas, Y., Ferrández, A., & Peral, J. (2005). Aplicación del Procesamiento de Lenguaje Natural en la Recuperación de Información. *Sociedad Española para el Procesamiento del Lenguaje Natural. Procesamiento del Lenguaje Natural*(34), 17-30.
- Rótolo, G., Francis, C., Craviotto, R., Viglia, S., Pereyra, A., & Ulgiati, S. (2014). Time to re-think the GMO revolution in agriculture. . *Ecological Informatics*.
- Salton, G. (1968). *Automatic Information Organization and Retrieval*. New York: Editorial McGraw-Hill.
- Santoso, H., Haw, S., & Abdul-Mehdi, Z. (2010). Ontology extraction from relational database: concept hierarchy as

- background knowledge. . *Knowledge Based Systems*, 457-464.
- Stratica, N., Kosseim, L., & Desai, B. (2005). Using semantic templates for a natural language interface to the CINDI virtual library. . *Data & Knowledge Engineering*, 4-19.
- Strzalkowski, T., Lin, F., Wang, J., & Perez-Carballo, J. (1999). Evaluating Natural Language Processing Techniques in Information Retrieval. *Natural Language Information Retrieval*, 7, 113–145.
- Sun, M., & Chai, J. Y. (2007). Discourse Processing for Context Question Answering Based on Linguistic Knowledge. *Knowledge-Based Systems. Special Issue on Intelligent User Interfaces*, 511-526.
- Sun, M., & Chai, J. Y. (2007). Discourse Processing for Context Question Answering Based on Linguistic Knowledge. *Knowledge-Based Systems: Special Issue on Intelligent User Interfaces*, 511-526.
- Sun, M., & Chai, J. Y. (2007). Discourse Processing for Context Question Answering Based on Linguistic Knowledge. *Knowledge-Based Systems. Special Issue on Intelligent User Interfaces*, 511-526.
- Tomioka, S. (2008). A step-by-step guide to ellipsis resolution. (K. Johnson, Ed.) *Topics in Ellipsis*, 210-228.
- Vieira, R., & Poesio, M. (2000). An Empirically Based System for Processing Definite Descriptions. *Computational Linguistics*, 539-593.
- Vila, K., Mazón, J., & Ferrández, A. (2011). Model-driven Development for Adapting Question Answering Systems to Restricted Domains. *Journal of Research and Practice in Information Technology*, 141-158.

- Vilares, J., Alonso, M., Ribadas, F., & Vilares. (2003). COLE experiments at CLEF 2002 Spanish monolingual track. *Lecture Notes in Computer Science*, 2785, 265–278.
- Voorhees, E. M. (2001). Overview of the TREC 2001 Question Answering Track. *Proceedings of the Tenth Text REtrieval Conference (TREC-10)*, 42-51.
- Voorhees, E. M. (2005). Overview of TREC 2004 Question Answering Track. *Proceedings of the 13th Text REtrieval Conference (TREC 2004)*, 52-62.
- Williams, S. (2000). Anaphoric reference and ellipsis resolution in a telephone-based spoken language system for accessing email. (S. Philip, & T. McEnery, Edits.) *Corpus-based and Computational Approaches to Discourse Anaphora*, 171-188.
- Woods, J. D., J.S., Marvin, U., & Powell, B. (1970). Lunar Anorthosites. *Science* 30, 167, págs. 602 – 604.
- World Health Organization. (2002). Obtenido de Foods derived from modern technology: 20 questions on genetically modified foods:
<http://www.who.int/foodsafety/publications/biotech/20questions/en/index.php>
- Zhai, C., Tong, X., Milic-Frayling, N., & Evans, D. (1997). Evaluation of Syntactic Phrase Indexing- CLARIT NLP Track Report. En K. Harman, & E. Voorhes (Ed.), *Proceedings of the Fifth Text REtrieval Conference(TREC-5)*, (págs. 347–358).
- Zheng, J., Chapman, W. W., Crowley, R., & S., S. G. (2011). Coreference resolution: A review of general methodologies and applications in the clinical domain. *Journal of Biomedical Informatics*, 1113-1122.

6. Anexos

6.1. *Anexo 1 - Consultas propuestas por expertos*

Detalle de las consultas propuestas por 2 expertos, para la evaluación de la propuesta.

- | | |
|---|--|
| a.1. Tell me the QTLs related to awn color in wheat. | e.1. What QTLs are related to the kernel weight in barley? |
| a.2. And to glume color? | e.2. Are there any biotechnology companies that have made Genetic Engineering with these QTLs in barley? |
| a.3. To semolina one? | e.3. Are there currently any transgenic barley varieties in the market? |
| b.1. Does any QTL affect on days to flowering in wheat? | |
| b.2. In which chromosome are they located? | |
| c.1. I want to increase grains per spike in barley. Which QTLs are related to this? | f.1. What QTLs are related to frost tolerance and resistance to Fusarium in barley? |
| c.2. What other traits are affected by them? | f.2. Since there is no common QTL, are there any studies showing these traits are related? |
| d.1. I would like to know the QTLs related to rust in wheat. | g.1. Are there currently any transgenic wheat? |
| d.2. Which genes influence the resistance to plagues in wheat? | g.2. Is it in the market? |
| d.3. Are there other plagues in wheat? | g.3. What is its price? |

h.1. What QTLs are related to awn type in barley?

h.2. And to lemma color?

h.3. To aleurone one?

i.1. Does any QTL affect to protein content in barley?

i.2. In which chromosome are they located?

j.1. I want to increase resistance to common bunt in wheat.

Which QTLs are related to this?

j.2. What other traits are affected by them?

k.1. I would like to know the QTLs related to biomolecules content in barley

(proteins/lipids/beta glucan).

k.2. Which genes influence the resistance to Fusarium in barley?

k.3. Are there other plagues in barley?

l.1. What QTLs are related to the grain yield in wheat?

l.2. Are there any biotechnology companies that have made Genetic Engineering with these QTLs in wheat?

q.2. What other traits are affected by them?

l.3. Are there currently any transgenic wheat varieties in the market?

m.1. What QTLs are related to grain yield and plant height in wheat?

m.2. Since there is no common QTL, are there any studies showing these traits are related?

n.1. Are there currently any transgenic barley?

n.2. Is it in the market?

n.3. What is its price?

o.1. Tell me the QTLs related to the number of kernels per spike in wheat.

o.2. And to resistance to BYDV?

o.3. To Septoria tritici?

p.1. Does any QTL affect on deoxynivalenol accumulation in wheat?

p.2. In which chromosome are they located?

q.1. I want to increase hull cover in barley. Which QTLs are related to this?

v.2. In which chromosome are they located?

r.1. I would like to know the QTLs related to tan spots in wheat.

r.2. Which genes influence the resistance to Russian wheat aphid in wheat?

r.3. Are there other plagues in wheat?

s.1. What QTLs are related to the friability in barley?

s.2. Are there any biotechnology companies that have made Genetic Engineering with these QTLs in barley?

t.1. What QTLs are related to resistance to Schizaphis and resistance to Powdery mildew in barley?

t.2. If there is no common QTL, are there any studies showing these traits are related?

u.1. Tell me the QTLs related to P/L ratio in wheat.

u.2. And to resistance to Powdery mildew?

u.3. To stem rust one?

v.1. Does any QTL affect to resistance to SBWMV in wheat?

w.1. I want to decrease leaf rust seedling in barley. Which QTLs are related to this?

w.2. What other traits are affected by them?

x.1. I would like to know the QTLs related to aspect of kernels in wheat.

x.2. Which genes influence the resistance to black point in wheat?

x.3. Are there other fungal infections in wheat? Which QTLs are involved in the resistance?

y.1. What QTLs are related to the heading date in barley?

y.2. Are there any biotechnology companies that have made Genetic Engineering with these QTLs in barley?

z.1. What QTLs are related to spike morphology and stripe rust in barley?

z.2. If there is no common QTL, are there any studies showing these traits are related?

aa.1. If there is some transgenic variety of rice, what traits are improved?

aa.2. Are there novel QTLs in the literature which are related to a certain trait in rice?

ab.1. Are there novel genes in the literature which are related to a certain trait in rice?

ab.2. In wheat?

ac.1. What QTLs are related to flowering in rice?

ac.2. And to panicle length?

ac.3. To brown rice one?

ad.1. Does any QTL affect on days to flowering in rice?

ad.2. In which chromosome are they located?

ae.1. I want to increase grain yield in rice. Which QTLs are related to this?

ae.2. What other traits are affected by them?

af.1. I would like to know the QTLs related to amylose content in rice.

af.2. Which genes influence the brown rice length in rice?

af.3. Are there other brown rice dimensions which can be affected?

ag.1. What QTLs are related to the plant height in rice?

ag.2. Are there any biotechnology companies that have made Genetic Engineering with these QTLs in rice?

ag.3. Are there currently any transgenic rice varieties in the market?

ah.1. What QTLs are related to panicle length and brown rice length in rice?

ah.2. Are there any studies showing these traits are related?

ai.1. What QTLs are related to hectolitic weight in barley?

ai.2. And to lipid content?

ai.3. To grain nitrogen one?

aj.1. Does any QTL affect on net blotch in barley?

aj.2. In which chromosome are they located?

ak.1. I want to decrease spikelet sterility in rice. Tell me the QTLs related to this trait.

ak.2. What other traits are affected by them?

al.1. I would like to know the QTLs related to spike row number in barley

al.2. Which genes influence the resistance to *Diuraphis* in barley?

al.3. Are there other plagues in barley?

am.1. What QTLs are related to the kernel texture in wheat?

am.2. Are there any companies working with these QTLs in wheat?

an.1. What QTLs are related to aromaticity and spikelet sterility in rice?

an.2. Which studies showing these traits are related?

ao.1. What QTLs are related to rachilla hair length in barley?

ao.2. And to awn type?

ao.3. To its roughness?

ap.1. Does any QTL affect to leaf scald in barley?

ap.2. In which chromosome are they located?

aq.1. I want to modify glume color in wheat. Which QTLs are related to the glume?

aq.2. What other traits are affected by them?

ar.1. I would like to know the QTLs related to spot blotch in barley.

ar.2. Which genes influence the resistance to *Schizaphis* in barley?

ar.3. Which other insects affecting barley?

ar.4. Which QTLs are involved in the resistance?

as.1. What QTLs are related to semolina color and kernel color in wheat?

as.2. Which studies showing these traits are related?

at.1. Show me the QTLs of interest for grain yield in wheat.

at.2. Show me the QTLs which are related simultaneously to resistance to *Fusarium* and to *Schizaphis* in barley.

- au.1. Are there any trait related to awn in barley?
au.2. In wheat?
- av.1. Are there any genes responsible of pericarp color in rice?
av.2. Of grain yield?
av.3. In wheat?
av.4. In barley?
- aw.1. What QTLs are related to lodging in rice?
aw.2. And to flowering time?
aw.3. To maturity one?
- ax.1. Are there more than 1 gene responsible of the grain yield in wheat?
ax.2. Which of these genes are involved in the grain yield in barley?
ax.3. In rice?
bd.2. Which are the relevant genes in the frost tolerance in wheat?
- ay.1. Does any QTL affect on aromaticity in rice?
ay.2. In which chromosome are they located?
- az.1. I want to obtain a Fusarium-resistant variety of wheat. What genes are suitable/appropriate/useful for this work?
- ba.1. What attributes are dependent on gene Sh3 in Hordeum?
ba.2. Which feature depends on gene blx1 in Hordeum?
- bb.1. I would like to know the QTLs related to panicle blast in rice.
bb.2. Which genes are related to the leaf blast in rice?
- bc.1. What QTLs are related to brown rice shape in rice?
bc.2. And to brown rice length?
bc.3. To the plant one ?
- bd.1. Tell me the QTLs linked to the awn color in wheat.
be.1. 1. Does any QTL affect to grain weight in rice?
be.2. In which chromosome are they located?

bf.1. Search for genes associated to grain nitrogen in wheat.

bf.2. In barley.

bg.1. I want to avoid lodging in rice. Which QTLs are related to lodging?

bg.2. What other traits are affected by them?

bh.1. What genes are underlying the lodging event in wheat?

bh.2. What ecological process is underlined by the gene CHL7 in rice?

bi.1. I would like to know the QTLs related to the development of rice plant.

bi.2. Which genes influence the pericarp color in rice?

bi.3. Are those genes related to the color of other parts?

bj.1. What trait is influenced by gene DP1 in rice?

bj.2. What biological feature is affected by QTL C1AS16 in rice?

6.2. Anexo 2- Trazas de ejecución de las consultas

Se presentan las trazas de ejecución de diferentes series de consultas

En primera instancia se presenta la salida luego de realizar el análisis morfológico (POS Tagger y Tree Tagger).

Luego se presenta la salida del análisis sintáctico parcial.

La última subsección contiene la salida del proceso de resolución de la anáfora.

6.2.1 Salida del análisis morfológico de las series de la a.1 a la r.3

```
:- x2(tipoWBB), conf(di, DI), cat([DI, '\', 'main_brown.pl'], R, _),
consult(R),abolish(fichInterfaz/1),assertz(fichInterfaz(R)),!
```

```
o(1,'frase_Brown',[w('Tell','tell','VB'),w('me','me','PPIO1'),w('the','the','DT'),w('QTLs',
,'qtls','NP'),w('related','relate','VBN'),w('to','to','TO'),w('awn','awn','NN'),w('color','col
our','NN'),w('in','in','IN'),w('wheat','wheat','NN'),w('!','!','SENT')]).
o(2,'frase_Brown',[w('And','and','CC'),w('to','to','TO'),w('glume','glume','NN'),w('color',
,'colour','NN'),w('?','?','SENT')]).
o(3,'frase_Brown',[w('To','to','TO'),w('semolina','semolina','NN'),w('one','one','NN'),w('
?','?','SENT')]).
o(4,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('!','!','SENT')]).
o(5,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('!','!','SENT')]).
o(6,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('!','!','SENT')]).
o(7,'frase_Brown',[w('Does','do','VBZDO'),w('any','any','DT'),w('QTL','qt1','NP'),w('aff
ect','affect','VB'),w('on','on','IN'),w('days','day','NNS'),w('to','to','TO'),w('flowering','flo
wering','VBG'),w('in','in','IN'),w('wheat','wheat','NN'),w('?','?','SENT')]).
o(8,'frase_Brown',[w('In','in','IN'),w('which','which','WDT'),w('chromosome','chromoso
me','NN'),w('are','be','VBPBE'),w('they','they','PPHS2'),w('located','locate','VBN'),w('?
','?','SENT')]).
o(9,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('NUEVA_SERIE','nueva
_serie','NP'),w('NUEVA_SERIE','nueva_serie','NP'),w('I','i','PPIS1'),w('want','want',
'VBP'),w('to','to','TO'),w('increase','increase','VB'),w('grains','grain','NNS'),w('per','per',
'IN'),w('spike','spike','NN'),w('in','in','IN'),w('barley','barley','NN'),w('!','!','SENT')]).
```

- o(10,'frase_Brown',[w('Which','which','WDT'),w('QTLs','qtls','NNS'),w('are','be','VBPBE'),w('related','relate','VBN'),w('to','to','TO'),w('this','this','DT'),w('?','?','SENT')]).
- o(11,'frase_Brown',[w('What','what','WP'),w('other','other','JJ'),w('traits','trait','NNS'),w('are','be','VBPBE'),w('affected','affect','VBN'),w('by','by','IN'),w('them','them','PPH O2'),w('?','?','SENT')]).
- o(12,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('',' ','SENT')]).
- o(13,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('',' ','SENT')]).
- o(14,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('',' ','SENT')]).
- o(15,'frase_Brown',[w('I','i','PPIS1'),w('would','would','MD'),w('like','like','VB'),w('to','to','TO'),w('know','know','VB'),w('the','the','DT'),w('QTLs','qtls','NP'),w('related','relate','VBN'),w('to','to','TO'),w('rust','rust','VB'),w('in','in','IN'),w('wheat','wheat','NN'),w('',' ','SENT')]).
- o(16,'frase_Brown',[w('Which','which','WDT'),w('genes','gene','NNS'),w('influence','influence','VBP'),w('the','the','DT'),w('resistance','resistance','NN'),w('to','to','TO'),w('plagues','plague','NNS'),w('in','in','IN'),w('wheat','wheat','NN'),w('?','?','SENT')]).
- o(17,'frase_Brown',[w('Are','be','VBPBE'),w('there','there','RB'),w('other','other','JJ'),w('plagues','plague','NNS'),w('in','in','IN'),w('wheat','wheat','NN'),w('?','?','SENT')]).
- o(18,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('',' ','SENT')]).
- o(19,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('',' ','SENT')]).
- o(20,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('',' ','SENT')]).
- o(21,'frase_Brown',[w('What','what','WP'),w('QTLs','qtls','NP'),w('are','be','VBPBE'),w('related','relate','VBN'),w('to','to','TO'),w('the','the','DT'),w('kernel','kernel','NN'),w('weight','weight','NN'),w('in','in','IN'),w('barley','barley','NN'),w('?','?','SENT')]).
- o(22,'frase_Brown',[w('Are','be','VBPBE'),w('there','there','RB'),w('any','any','DT'),w('biotechnology','biotechnology','NN'),w('companies','company','NNS'),w('that','that','CC'),w('havemade','havemade','JJ'),w('Genetic','genetic','NP'),w('Engineering','engineering','NP'),w('with','with','IN'),w('these','these','DT'),w('QTLs','qtls','NP'),w('in','in','IN'),w('barley','barley','NN'),w('?','?','SENT')]).
- o(23,'frase_Brown',[w('Are','be','VBPBE'),w('there','there','RB'),w('currently','currently','RB'),w('any','any','DT'),w('transgenic','transgenic','NP'),w('barley','barley','NN'),w('varieties','variety','NNS'),w('in','in','IN'),w('themarket','themarket','NN'),w('?','?','SENT')]).
- o(24,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('',' ','SENT')]).
- o(25,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('',' ','SENT')]).
- o(26,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('',' ','SENT')]).
- o(27,'frase_Brown',[w('What','what','WP'),w('QTLs','qtls','NP'),w('are','be','VBPBE'),w('related','relate','VBN'),w('to','to','TO'),w('frost','frost','VB'),w('tolerance','tolerance','NN'),w('and','and','CC'),w('resistance','resistance','NN'),w('to','to','TO'),w('Fusarium','fusarium','NP'),w('in','in','IN'),w('barley','barley','NN'),w('?','?','SENT')]).
- o(28,'frase_Brown',[w('Since','since','IN'),w('there','there','EX'),w('is','be','VBZBE'),w('no','no','DT'),w('common','common','JJ'),w('QTL','qt1','NP'),w('',' ',' '),w('are','be','VBPBE'),w('there','there','RB'),w('any','any','DT'),w('studies','study','NNS'),w('showing','show','VBG'),w('these','these','DT'),w('traits','trait','NNS'),w('are','be','VBPBE'),w('related','relate','VBN'),w('?','?','SENT')]).
- o(29,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('',' ','SENT')]).

o(30,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(31,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(32,'frase_Brown',[w('Are','be','VBPBE'),w('there','there','RB'),w('currently','currently','RB'),w('any','any','DT'),w('transgenic','transgenic','NP'),w('wheat','wheat','NN'),w('?',',','SENT')]).

o(33,'frase_Brown',[w('Is','be','VBZBE'),w('it','it','PPH1R2'),w('in','in','IN'),w('the','the','DT'),w('market','market','NN'),w('?',',','SENT')]).

o(34,'frase_Brown',[w('What','what','WP'),w('is','be','VBZBE'),w('its','its','POS3N'),w('price','price','NN'),w('?',',','SENT')]).

o(35,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(36,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(37,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(38,'frase_Brown',[w('What','what','WP'),w('QTLs','qtls','NP'),w('are','be','VBPBE'),w('related','relate','VBN'),w('to','to','TO'),w('awn','awn','NN'),w('type','type','NN'),w('in','in','IN'),w('barley','barley','NN'),w('?',',','SENT')]).

o(39,'frase_Brown',[w('And','and','CC'),w('to','to','TO'),w('lemma','lemma','NN'),w('color','colour','NN'),w('?',',','SENT')]).

o(40,'frase_Brown',[w('To','to','TO'),w('aleurone','aleurone','NN'),w('one','one','NN'),w('?',',','SENT')]).

o(41,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(42,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(43,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(44,'frase_Brown',[w('Does','do','VBZDO'),w('any','any','DT'),w('QTL','qtl','NP'),w('affect','affect','VB'),w('to','to','TO'),w('protein','protein','NN'),w('content','content','NN'),w('in','in','IN'),w('barley','barley','NN'),w('?',',','SENT')]).

o(45,'frase_Brown',[w('In','in','IN'),w('which','which','WDT'),w('chromosome','chromosome','NN'),w('are','be','VBPBE'),w('they','they','PPHS2'),w('located','locate','VBN'),w('?',',','SENT')]).

o(46,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(47,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(48,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(49,'frase_Brown',[w('I','i','PPIS1'),w('want','want','VBP'),w('to','to','TO'),w('increase','increase','VB'),w('resistance','resistance','NN'),w('to','to','TO'),w('common','common','JJ'),w('bunt','bunt','NN'),w('in','in','IN'),w('wheat','wheat','NN'),w('.',',','SENT')]).

o(50,'frase_Brown',[w('Which','which','WDT'),w('QTLs','qtls','NNS'),w('are','be','VBPBE'),w('related','relate','VBN'),w('to','to','TO'),w('this','this','DT'),w('?',',','SENT')]).

o(51,'frase_Brown',[w('What','what','WP'),w('other','other','JJ'),w('traits','trait','NNS'),w('are','be','VBPBE'),w('affected','affect','VBN'),w('by','by','IN'),w('them','them','PPH O2'),w('?',',','SENT')]).

o(52,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(53,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(54,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).

o(55,'frase_Brown',[w('I','i','PPIS1'),w('would','would','MD'),w('like','like','VB'),w('to','to','TO'),w('know','know','VB'),w('the','the','DT'),w('QTLs','qtls','NP'),w('related','relate','VBN'),w('to','to','TO'),w('biomolecules','biomolecules','NNS'),w('content','content','

JJ'),w('in','in','IN'),w('barley','barley','NN'),w('(','(','(','w('proteins/lipids/beta','protein
s/lipids/beta','NN'),w('glucan','glucan','NN'),w(')','(','(','w('','','','SENT')))).

o(56,'frase_Brown',[w('Which','which','WDT'),w('genes','gene','NNS'),w('influence','infl
uence','VBP'),w('the','the','DT'),w('resistance','resistance','NN'),w('to','to','TO'),w('Fus
arium','fusarium','NP'),w('in','in','IN'),w('barley','barley','NN'),w('?','?','SENT'))).

o(57,'frase_Brown',[w('Are','be','VBPBE'),w('there','there','RB'),w('other','other','JJ'),
w('plagues','plague','NNS'),w('in','in','IN'),w('barley','barley','NN'),w('?','?','SENT'))).

o(58,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('','','SENT'))).

o(59,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('','','SENT'))).

o(60,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('','','SENT'))).

o(61,'frase_Brown',[w('What','what','WP'),w('QTLs','qtls','NP'),w('are','be','VBPBE'),
w('related','relate','VBN'),w('to','to','TO'),w('the','the','DT'),w('grain','grain','NN'),w('y
ield','yield','NN'),w('in','in','IN'),w('wheat','wheat','NN'),w('?','?','SENT'))).

o(62,'frase_Brown',[w('Are','be','VBPBE'),w('there','there','RB'),w('any','any','DT'),w(''
biotechnology','biotechnology','NN'),w('companies','company','NNS'),w('that','that','WD
T'),w('have','have','VBPBE'),w('made','make','VBN'),w('Genetic','genetic','NP'),w('Eng
ineering','engineering','NP'),w('with','with','IN'),w('these','these','DT'),w('QTLs','qtls','
NP'),w('in','in','IN'),w('wheat','wheat','NN'),w('?','?','SENT'))).

o(63,'frase_Brown',[w('Are','be','VBPBE'),w('there','there','RB'),w('currently','currently
,','RB'),w('any','any','DT'),w('transgenic','transgenic','NP'),w('wheat','wheat','NN'),w('v
arieties','variety','NNS'),w('in','in','IN'),w('the','the','DT'),w('market','market','NN'),w(''
,','?','?','SENT'))).

o(64,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('','','SENT'))).

o(65,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('','','SENT'))).

o(66,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('','','SENT'))).

o(67,'frase_Brown',[w('What','what','WP'),w('QTLs','qtls','NP'),w('are','be','VBPBE'),
w('related','relate','VBN'),w('to','to','TO'),w('grain','grain','NN'),w('yield','yield','NN'),
w('and','and','CC'),w('plant','plant','NN'),w('height','height','NN'),w('in','in','IN'),w('wh
eat','wheat','NN'),w('?','?','SENT'))).

o(68,'frase_Brown',[w('Since','since','IN'),w('there','there','EX'),w('is','be','VBZBE'),w(''
no','no','DT'),w('common','common','JJ'),w('QTL','qtl','NP'),w('','','',''),w('are','be','VB
PBE'),w('there','there','RB'),w('any','any','DT'),w('studies','study','NNS'),w('showing','s
how','VBG'),w('these','these','DT'),w('traits','trait','NNS'),w('are','be','VBPBE'),w('rela
ted','relate','VBN'),w('?','?','SENT'))).

o(69,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('','','SENT'))).

o(70,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('','','SENT'))).

o(71,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('','','SENT'))).

o(72,'frase_Brown',[w('Are','be','VBPBE'),w('there','there','RB'),w('currently','currently
,','RB'),w('any','any','DT'),w('transgenic','transgenic','NP'),w('barley','barley','NN'),w(''
,','?','?','SENT'))).

o(73,'frase_Brown',[w('Is','be','VBZBE'),w('it','it','PPH1R2'),w('in','in','IN'),w('the','the
,','DT'),w('market','market','NN'),w('?','?','SENT'))).

o(74,'frase_Brown',[w('What','what','WP'),w('is','be','VBZBE'),w('its','its','POS3N'),w(''
price','price','NN'),w('?','?','SENT'))).

o(75,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('','','SENT'))).

- o(76,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).
- o(77,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).
- o(78,'frase_Brown',[w('Tellme','tellme','NP'),w('the','the','DT'),w('QTLs','qtls','NP'),w('related','relate','VBN'),w('to','to','TO'),w('the','the','DT'),w('number','number','NN'),w('of','of','OF'),w('kernels','kernel','NNS'),w('per','per','IN'),w('spike','spike','NN'),w('in','in','IN'),w('wheat','wheat','NN'),w('.',',','SENT')]).
- o(79,'frase_Brown',[w('And','and','CC'),w('to','to','TO'),w('resistance','resistance','NN'),w('to','to','TO'),w('BYDV','bydv','NP'),w('?', '?','SENT')]).
- o(80,'frase_Brown',[w('To','to','TO'),w('Septoria','septoria','JJ'),w('tritici','tritici','NNS'),w('?', '?','SENT')]).
- o(81,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).
- o(82,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).
- o(83,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).
- o(84,'frase_Brown',[w('Does','do','VBZDO'),w('any','any','DT'),w('QTL','qtl','NP'),w('affect','affect','VB'),w('on','on','IN'),w('deoxynivalenol','deoxynivalenol','JJ'),w('accumulation','accumulation','NN'),w('in','in','IN'),w('wheat','wheat','NN'),w('?', '?','SENT')]).
- o(85,'frase_Brown',[w('In','in','IN'),w('which','which','WDT'),w('chromosome','chromosome','NN'),w('are','be','VBPBE'),w('they','they','PPHS2'),w('located','locate','VBN'),w('?', '?','SENT')]).
- o(86,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).
- o(87,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).
- o(88,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).
- o(89,'frase_Brown',[w('I','i','PPIS1'),w('want','want','VBP'),w('to','to','TO'),w('increase','increase','VB'),w('hull','hull','NN'),w('cover','cover','NN'),w('in','in','IN'),w('barley.Which','barley.which','JJ'),w('QTLs','qtls','NNS'),w('are','be','VBPBE'),w('related','relate','VBN'),w('to','to','TO'),w('this','this','DT'),w('?', '?','SENT')]).
- o(90,'frase_Brown',[w('What','what','WP'),w('other','other','JJ'),w('traits','trait','NNS'),w('are','be','VBPBE'),w('affected','affect','VBN'),w('by','by','IN'),w('them','them','PPH O2'),w('?', '?','SENT')]).
- o(91,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).
- o(92,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).
- o(93,'frase_Brown',[w('NUEVA_SERIE','nueva_serie','NP'),w('.',',','SENT')]).
- o(94,'frase_Brown',[w('I','i','PPIS1'),w('would','would','MD'),w('like','like','VB'),w('to','to','TO'),w('know','know','VB'),w('the','the','DT'),w('QTLs','qtls','NP'),w('related','relate','VBN'),w('to','to','TO'),w('tan','tan','VB'),w('spots','spot','NNS'),w('in','in','IN'),w('wheat','wheat','NN'),w('.',',','SENT')]).
- o(95,'frase_Brown',[w('Which','which','WDT'),w('genes','gene','NNS'),w('influence','influence','VBP'),w('the','the','DT'),w('resistance','resistance','NN'),w('to','to','TO'),w('Russian','russian','JJ'),w('wheat','wheat','NN'),w('aphid','aphid','NN'),w('in','in','IN'),w('wheat','wheat','NN'),w('?', '?','SENT')]).
- o(96,'frase_Brown',[w('Are','be','VBPBE'),w('there','there','RB'),w('other','other','JJ'),w('plagues','plague','NNS'),w('in','in','IN'),w('wheat','wheat','NN'),w('?', '?','SENT')]).

6.2.2 Salida del análisis sintáctico de las series de la a.1 a la j.2

Sentence N§ 1 --> frase_Brown
 Tell me the QTLs related to awn color in wheat.

** PARTIALLY PARSED SENTENCE:
 conc(1,0,11)
 ** HEAD OF THE VERBAL PHRASE:
 conc(_280664,_280666,infinitivo,_281980,1,0,1)
 conc(_280664,_280666,infinitivo,_281980,1,0,1)
 conc(_280664,_280666,infinitivo,verboSolo,1,0,1)
 ** VERB:
 conc(_280664,_280666,infinitivo,resto,1,0,1)
 Tell

** NOUN PHRASE:
 conc(sing,_280764,prim,pronEnglish,compl,_282038,_282040,1,1,2)
 conc(sing,_280764,prim,pronEnglish,compl,_282038,_282040,1,1,2)
 conc(sing,_280764,prim,pronEnglish,compl,_282038,_282040,1,1,2)
 ** SIMPLE NOUN PHRASE:
 conc(sing,_280764,prim,pronEnglish,compl,_282038,_282040,1,1,2)
 ** PRONOUN:
 conc(sing,_280764,prim,persNoRefl,compl,_280772,_280774,1,1,2)
 me

** NOUN PHRASE:
 conc(sing,_313274,terc,propio,_302866,_302868,_302870,1,2,10)
 conc(sing,_313274,terc,propio,_302866,_302868,_302870,1,2,10)
 conc(sing,_313274,terc,propio,_302866,_302868,_302870,1,2,10)
 ** SIMPLE NOUN PHRASE:
 conc(sing,_313274,terc,propio,_302866,_302868,_302870,1,2,10)
 ** DETERMINER 1:
 conc(sing,_313274,det,1,2,3)
 ** DETERMINER:
 conc(sing,_313274,det,1,2,3)
 the

** HEAD OF THE NOUN PHRASE:
 conc(sing,_313274,propio,1,3,4)
 ** NOUN:
 conc(sing,_313274,propio,1,3,4)
 QTLs

** POST-MODIFIER:
 conc(sing,_313274,_313370,1,4,10)
 ** RELATIVE SENTENCE:
 conc(participio,1,4,10)
 ** HEAD OF THE VERBAL PHRASE:
 conc(_281058,_281060,participio,verboSolo,1,4,5)
 conc(_281058,_281060,participio,verboSolo,1,4,5)
 conc(_281058,_281060,participio,verboSolo,1,4,5)
 ** VERB:
 conc(_281058,_281060,participio,resto,1,4,5)
 related

** PREPOSITIONAL PHRASE:
 conc(_281156,of,1,5,10)
 ** SIMPLE PREPOSITIONAL PHRASE:

```

conc(_281156,of,1,5,10)
** PREPOSITION:
  conc(_281156,of,1,5,6)
  conc(_281156,of,1,5,6)
** SIMPLE PREPOSITION:
  conc(_281156,to,1,5,6)
  to
** NOUN PHRASE:
  conc(sing,_323482,terc,comun,_316902,_316904,_316906,1,6,10)
  conc(sing,_323482,terc,comun,_316902,_316904,_316906,1,6,10)
  conc(sing,_323482,terc,comun,_316902,_316904,_316906,1,6,10)
** SIMPLE NOUN PHRASE:
  conc(sing,_323482,terc,comun,_316902,_316904,_316906,1,6,10)
** HEAD OF THE NOUN PHRASE:
  conc(sing,_323482,comun,1,6,8)
  ** NOUN:
    conc(sing,_281252,comun,1,6,7)
    awn
  ** NOUN:
    conc(sing,_323482,comun,1,7,8)
    color
** PREPOSITIONAL PHRASE:
  conc(_281442,of,1,8,10)
** SIMPLE PREPOSITIONAL PHRASE:
  conc(_281442,of,1,8,10)
** PREPOSITION:
  conc(_281442,of,1,8,9)
  conc(_281442,of,1,8,9)
** SIMPLE PREPOSITION:
  conc(_281442,resto,1,8,9)
  in
** NOUN PHRASE:
  conc(sing,_336640,terc,comun,_330060,_330062,_330064,1,9,10)
  conc(sing,_336640,terc,comun,_330060,_330062,_330064,1,9,10)
  conc(sing,_336640,terc,comun,_330060,_330062,_330064,1,9,10)
** SIMPLE NOUN PHRASE:
  conc(sing,_336640,terc,comun,_330060,_330062,_330064,1,9,10)
** HEAD OF THE NOUN PHRASE:
  conc(sing,_336640,comun,1,9,10)
  ** NOUN:
    conc(sing,_336640,comun,1,9,10)
    wheat

```

** NON-PARSED WORD:

<----->

Sentence N§ 2 --> frase_Brown

And to glume color ?

** PARTIALLY PARSED SENTENCE:

conc(2,0,5)

** CONJUNCTION:

conc(2,0,1)

And

** PREPOSITIONAL PHRASE:

conc(_383110,of,2,1,4)

** SIMPLE PREPOSITIONAL PHRASE:

conc(_383110,of,2,1,4)

```

** PREPOSITION:
  conc(_383110,of,2,1,2)
  conc(_383110,of,2,1,2)
** SIMPLE PREPOSITION:
  conc(_383110,to,2,1,2)
  to
** NOUN PHRASE:
  conc(sing,_394168,terc.comun,_387588,_387590,_387592,2,2,4)
  conc(sing,_394168,terc.comun,_387588,_387590,_387592,2,2,4)
  conc(sing,_394168,terc.comun,_387588,_387590,_387592,2,2,4)
** SIMPLE NOUN PHRASE:
  conc(sing,_394168,terc.comun,_387588,_387590,_387592,2,2,4)
** HEAD OF THE NOUN PHRASE:
  conc(sing,_394168,comun,2,2,4)
  ** NOUN:
    conc(sing,_383206,comun,2,2,3)
    glume
  ** NOUN:
    conc(sing,_394168,comun,2,3,4)
    color
** NON-PARSED WORD:
  ?
<----->
  Sentence N§ 3 --> frase_Brown
To semolina one ?

** PARTIALLY PARSED SENTENCE:
  conc(3,0,4)
** PREPOSITIONAL PHRASE:
  conc(_467520,of,3,0,3)
** SIMPLE PREPOSITIONAL PHRASE:
  conc(_467520,of,3,0,3)
** PREPOSITION:
  conc(_467520,of,3,0,1)
  conc(_467520,of,3,0,1)
** SIMPLE PREPOSITION:
  conc(_467520,to,3,0,1)
  To
** NOUN PHRASE:
  conc(sing,_478926,terc.comun,_472346,_472348,_472350,3,1,3)
  conc(sing,_478926,terc.comun,_472346,_472348,_472350,3,1,3)
  conc(sing,_478926,terc.comun,_472346,_472348,_472350,3,1,3)
** SIMPLE NOUN PHRASE:
  conc(sing,_478926,terc.comun,_472346,_472348,_472350,3,1,3)
** HEAD OF THE NOUN PHRASE:
  conc(sing,_478926,comun,3,1,3)
  ** NOUN:
    conc(sing,_467616,comun,3,1,2)
    semolina
  ** NOUN:
    conc(sing,_478926,comun,3,2,3)
    one
** NON-PARSED WORD:
  ?
<----->

```

Sentence N§ 4 --> frase_Brown
NUEVA_SERIE.

** PARTIALLY PARSED SENTENCE:
conc(4,0,2)
** NOUN PHRASE:
conc(sing,_562472,terc,propio,_551994,_551996,_551998,4,0,1)
conc(sing,_562472,terc,propio,_551994,_551996,_551998,4,0,1)
conc(sing,_562472,terc,propio,_551994,_551996,_551998,4,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_562472,terc,propio,_551994,_551996,_551998,4,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_562472,propio,4,0,1)
** NOUN:
conc(sing,_562472,propio,4,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.

<----->

Sentence N§ 5 --> frase_Brown
NUEVA_SERIE.

** PARTIALLY PARSED SENTENCE:
conc(5,0,2)
** NOUN PHRASE:
conc(sing,_586212,terc,propio,_575734,_575736,_575738,5,0,1)
conc(sing,_586212,terc,propio,_575734,_575736,_575738,5,0,1)
conc(sing,_586212,terc,propio,_575734,_575736,_575738,5,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_586212,terc,propio,_575734,_575736,_575738,5,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_586212,propio,5,0,1)
** NOUN:
conc(sing,_586212,propio,5,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.

<----->

Sentence N§ 6 --> frase_Brown
NUEVA_SERIE.

** PARTIALLY PARSED SENTENCE:
conc(6,0,2)
** NOUN PHRASE:
conc(sing,_609952,terc,propio,_599474,_599476,_599478,6,0,1)
conc(sing,_609952,terc,propio,_599474,_599476,_599478,6,0,1)
conc(sing,_609952,terc,propio,_599474,_599476,_599478,6,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_609952,terc,propio,_599474,_599476,_599478,6,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_609952,propio,6,0,1)
** NOUN:
conc(sing,_609952,propio,6,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.

<----->

Sentence N§ 7 --> frase_Brown

Does any QTL affect on days to flowering in wheat ?

** PARTIALLY PARSED SENTENCE:

conc(7,0,11)

** HEAD OF THE VERBAL PHRASE:

conc(sing,terc,presente,_625768,7,0,1)

conc(sing,terc,presente,_625768,7,0,1)

conc(sing,terc,presente,verboSolo,7,0,1)

** VERB:

conc(sing,terc,presente,do,7,0,1)

Does

** NOUN PHRASE:

conc(sing,_636932,terc,propio,_625824,_625826,_625828,7,1,3)

conc(sing,_636932,terc,propio,_625824,_625826,_625828,7,1,3)

conc(sing,_636932,terc,propio,_625824,_625826,_625828,7,1,3)

** SIMPLE NOUN PHRASE:

conc(sing,_636932,terc,propio,_625824,_625826,_625828,7,1,3)

** DETERMINER 1:

conc(sing,_636932,det,7,1,2)

** DETERMINER:

conc(sing,_636932,det,7,1,2)

any

** HEAD OF THE NOUN PHRASE:

conc(sing,_636932,propio,7,2,3)

** NOUN:

conc(sing,_636932,propio,7,2,3)

QTL

** HEAD OF THE VERBAL PHRASE:

conc(_624750,_624752,infinitivo,_642166,7,3,4)

conc(_624750,_624752,infinitivo,_642166,7,3,4)

conc(_624750,_624752,infinitivo,verboSolo,7,3,4)

** VERB:

conc(_624750,_624752,infinitivo,resto,7,3,4)

affect

** PREPOSITIONAL PHRASE:

conc(_624848,of,7,4,10)

** SIMPLE PREPOSITIONAL PHRASE:

conc(_624848,of,7,4,10)

** PREPOSITION:

conc(_624848,of,7,4,5)

conc(_624848,of,7,4,5)

** SIMPLE PREPOSITION:

conc(_624848,resto,7,4,5)

on

** NOUN PHRASE:

conc(pl,_653082,terc,comun,_646502,_646504,_646506,7,5,10)

conc(pl,_653082,terc,comun,_646502,_646504,_646506,7,5,10)

conc(pl,_653082,terc,comun,_646502,_646504,_646506,7,5,10)

** SIMPLE NOUN PHRASE:

conc(pl,_653082,terc,comun,_646502,_646504,_646506,7,5,10)

** HEAD OF THE NOUN PHRASE:

conc(pl,_653082,comun,7,5,6)

** NOUN:

conc(pl,_653082,comun,7,5,6)


```

    days
** POST-MODIFIER:
conc(pl,_653082,_653178,7,6,10)
** RELATIVE SENTENCE:
conc(participio,7,6,10)
** SIMPLE PREPOSITION:
conc(_625038,to,7,6,7)
to
** HEAD OF THE VERBAL PHRASE:
conc(_625132,_625134,gerundio,verboSolo,7,7,8)
conc(_625132,_625134,gerundio,verboSolo,7,7,8)
conc(_625132,_625134,gerundio,verboSolo,7,7,8)
** VERB:
conc(_625132,_625134,gerundio,resto,7,7,8)
flowering
** PREPOSITIONAL PHRASE:
conc(_625230,of,7,8,10)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_625230,of,7,8,10)
** PREPOSITION:
conc(_625230,of,7,8,9)
conc(_625230,of,7,8,9)
** SIMPLE PREPOSITION:
conc(_625230,resto,7,8,9)
in
** NOUN PHRASE:
conc(sing,_665402,terc,comun,_658822,_658824,_658826,7,9,10)
conc(sing,_665402,terc,comun,_658822,_658824,_658826,7,9,10)
conc(sing,_665402,terc,comun,_658822,_658824,_658826,7,9,10)
** SIMPLE NOUN PHRASE:
conc(sing,_665402,terc,comun,_658822,_658824,_658826,7,9,10)
** HEAD OF THE NOUN PHRASE:
conc(sing,_665402,comun,7,9,10)
** NOUN:
conc(sing,_665402,comun,7,9,10)
wheat

** NON-PARSED WORD:
?
<----->
Sentence N§ 8 --> frase_Brown
In which chromosome are they located ?

** PARTIALLY PARSED SENTENCE:
conc(8,0,7)
** NON-PARSED WORD:
In
** RELATIVE SENTENCE:
conc(rel,8,1,5)
** PRONOUN:
conc(_768952,_768954,_768956,rel,_768960,_768962,_768964,8,1,2)
which
** NOUN PHRASE:
conc(sing,_813206,terc,comun,_806766,_806768,_806770,8,2,3)
conc(sing,_813206,terc,comun,_806766,_806768,_806770,8,2,3)
conc(sing,_813206,terc,comun,_806766,_806768,_806770,8,2,3)
** SIMPLE NOUN PHRASE:
conc(sing,_813206,terc,comun,_806766,_806768,_806770,8,2,3)

```

```

** HEAD OF THE NOUN PHRASE:
conc(sing,_813206,comun,8,2,3)
** NOUN:
conc(sing,_813206,comun,8,2,3)
chromosome
** HEAD OF THE VERBAL PHRASE:
conc(_769152,_769154,presente,_806788,8,3,4)
conc(_769152,_769154,presente,_806788,8,3,4)
conc(_769152,_769154,presente,verboSolo,8,3,4)
** VERB:
conc(_769152,_769154,presente,be,8,3,4)
are
** NOUN PHRASE:
conc(pl,_769252,terc,pronEnglish,suj,_806832,_806834,8,4,5)
conc(pl,_769252,terc,pronEnglish,suj,_806832,_806834,8,4,5)
conc(pl,_769252,terc,pronEnglish,suj,_806832,_806834,8,4,5)
** SIMPLE NOUN PHRASE:
conc(pl,_769252,terc,pronEnglish,suj,_806832,_806834,8,4,5)
** PRONOUN:
conc(pl,_769252,terc,persNoRefl,suj,_769260,_769262,8,4,5)
they
** HEAD OF THE VERBAL PHRASE:
conc(_769354,_769356,participio,_806558,8,5,6)
conc(_769354,_769356,participio,_806558,8,5,6)
conc(_769354,_769356,participio,verboSolo,8,5,6)
** VERB:
conc(_769354,_769356,participio,resto,8,5,6)
located
** NON-PARSED WORD:
?
<----->
Sentence N§ 9 --> frase_Brown
I want to increase grains per spike in barley.
** PARTIALLY PARSED SENTENCE:
conc(9,0,13)
** NOUN PHRASE:
conc(sing,_914398,prim,pronEnglish,suj,_933714,_933716,9,3,4)
conc(sing,_914398,prim,pronEnglish,suj,_933714,_933716,9,3,4)
conc(sing,_914398,prim,pronEnglish,suj,_933714,_933716,9,3,4)
** SIMPLE NOUN PHRASE:
conc(sing,_914398,prim,pronEnglish,suj,_933714,_933716,9,3,4)
** PRONOUN:
conc(sing,_914398,prim,persNoRefl,suj,_914406,_914408,9,3,4)
I
** HEAD OF THE VERBAL PHRASE:
conc(_914500,_914502,presente,_954064,9,4,7)
conc(_914500,_914502,presente,_954064,9,4,5)
conc(_914500,_914502,presente,verboSolo,9,4,5)
** VERB:
conc(_914500,_914502,presente,resto,9,4,5)
want
** VERBAL PERIPHRAISIS:
conc(9,5,7)
** NON-PARSED WORD:

```

```

to
** VERB:
  conc(_914692,_914694,infinitivo,resto,9,6,7)
  increase
** NOUN PHRASE:
  conc(_958870,_958872,terc,cantidad,_954120,_954122,_954124,9,7,12)
  conc(_958870,_958872,terc,cantidad,_954120,_954122,_954124,9,7,12)
  conc(_958870,_958872,terc,cantidad,_954120,_954122,_954124,9,7,12)
** SIMPLE NOUN PHRASE:
  conc(_958870,_958872,terc,cantidad,_954120,_954122,_954124,9,7,12)
** UNIT OF MEASURE:
  conc(9,7,8)
** NON-PARSED WORD:
  grains
** PREPOSITIONAL PHRASE:
  conc(_914886,of,9,8,12)
** SIMPLE PREPOSITIONAL PHRASE:
  conc(_914886,of,9,8,12)
** PREPOSITION:
  conc(_914886,of,9,8,9)
  conc(_914886,of,9,8,9)
** SIMPLE PREPOSITION:
  conc(_914886,resto,9,8,9)
  per
** NOUN PHRASE:
  conc(sing,_971198,terc,comun,_964618,_964620,_964622,9,9,12)
  conc(sing,_971198,terc,comun,_964618,_964620,_964622,9,9,12)
  conc(sing,_971198,terc,comun,_964618,_964620,_964622,9,9,12)
** SIMPLE NOUN PHRASE:
  conc(sing,_971198,terc,comun,_964618,_964620,_964622,9,9,12)
** HEAD OF THE NOUN PHRASE:
  conc(sing,_971198,comun,9,9,10)
** NOUN:
  conc(sing,_971198,comun,9,9,10)
  spike
** PREPOSITIONAL PHRASE:
  conc(_915076,of,9,10,12)
** SIMPLE PREPOSITIONAL PHRASE:
  conc(_915076,of,9,10,12)
** PREPOSITION:
  conc(_915076,of,9,10,11)
  conc(_915076,of,9,10,11)
** SIMPLE PREPOSITION:
  conc(_915076,resto,9,10,11)
  in
** NOUN PHRASE:
  conc(sing,_984174,terc,comun,_977594,_977596,_977598,9,11,12)
  conc(sing,_984174,terc,comun,_977594,_977596,_977598,9,11,12)
  conc(sing,_984174,terc,comun,_977594,_977596,_977598,9,11,12)
** SIMPLE NOUN PHRASE:
  conc(sing,_984174,terc,comun,_977594,_977596,_977598,9,11,12)
** HEAD OF THE NOUN PHRASE:
  conc(sing,_984174,comun,9,11,12)
** NOUN:
  conc(sing,_984174,comun,9,11,12)
  barley
** NON-PARSED WORD:

```

<----->

Sentence N§ 10 --> frase_Brown

Which QTLs are related to this ?

** PARTIALLY PARSED SENTENCE:

conc(10,0,7)

** RELATIVE SENTENCE:

conc(rel,10,0,4)

** PRONOUN:

conc(_1014640,_1014642,_1014644,rel,_1014648,_1014650,_1014652,10,0,1)

Which

** NOUN PHRASE:

conc(pl,_1022238,terc,comun,_1015798,_1015800,_1015802,10,1,2)

conc(pl,_1022238,terc,comun,_1015798,_1015800,_1015802,10,1,2)

conc(pl,_1022238,terc,comun,_1015798,_1015800,_1015802,10,1,2)

** SIMPLE NOUN PHRASE:

conc(pl,_1022238,terc,comun,_1015798,_1015800,_1015802,10,1,2)

** HEAD OF THE NOUN PHRASE:

conc(pl,_1022238,comun,10,1,2)

** NOUN:

conc(pl,_1022238,comun,10,1,2)

QTLs

** HEAD OF THE VERBAL PHRASE:

conc(_1014840,_1014842,pasiva,compBe,10,2,4)

conc(_1014840,_1014842,pasiva,compBe,10,2,4)

** COMPOUND VERB BE:

conc(_1014840,_1014842,pasiva,compBe,10,2,4)

** VERB:

conc(_1014840,_1014842,presente,be,10,2,3)

are

** VERB:

conc(_1014938,_1014940,participio,resto,10,3,4)

related

** NON-PARSED WORD:

to

** NON-PARSED WORD:

this

** NON-PARSED WORD:

?

<----->

Sentence N§ 11 --> frase_Brown

What other traits are affected by them ?

** PARTIALLY PARSED SENTENCE:

conc(11,0,8)

** RELATIVE SENTENCE:

conc(rel,11,0,7)

** PRONOUN:

conc(_1217676,_1217678,_1217680,rel,_1217684,_1217686,_1217688,11,0,1)

What

** NOUN PHRASE:

conc(pl,_1225432,terc,comun,_1218922,_1218924,_1218926,11,1,3)

conc(pl,_1225432,terc,comun,_1218922,_1218924,_1218926,11,1,3)

conc(pl,_1225432,terc,comun,_1218922,_1218924,_1218926,11,1,3)

** SIMPLE NOUN PHRASE:

```

conc(pl,_1225432,terc,comun,_1218922,_1218924,_1218926,11,1,3)
** PRE-MODIFIER:
conc(pl,_1225432,adj,11,1,2)
** ADJECTIVAL PHRASE:
conc(pl,_1225432,11,1,2)
** COORDINATED ADJECTIVE:
conc(pl,_1225432,_1228214,11,1,2)
** SIMPLE ADJECTIVE:
conc(pl,_1225432,cal,11,1,2)
other
** HEAD OF THE NOUN PHRASE:
conc(pl,_1225432,comun,11,2,3)
** NOUN:
conc(pl,_1225432,comun,11,2,3)
traits
** HEAD OF THE VERBAL PHRASE:
conc(_1217972,_1217974,pasiva,compBe,11,3,5)
conc(_1217972,_1217974,pasiva,compBe,11,3,5)
** COMPOUND VERB BE:
conc(_1217972,_1217974,pasiva,compBe,11,3,5)
** VERB:
conc(_1217972,_1217974,presente,be,11,3,4)
are
** VERB:
conc(_1218070,_1218072,participio,resto,11,4,5)
affected
** PREPOSITIONAL PHRASE:
conc(_1218168,resto,11,5,7)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_1218168,resto,11,5,7)
** PREPOSITION:
conc(_1218168,resto,11,5,6)
conc(_1218168,resto,11,5,6)
** SIMPLE PREPOSITION:
conc(_1218168,resto,11,5,6)
by
** NOUN PHRASE:
conc(pl,_1218264,terc,pronEnglish,compl,_1234834,_1234836,11,6,7)
conc(pl,_1218264,terc,pronEnglish,compl,_1234834,_1234836,11,6,7)
conc(pl,_1218264,terc,pronEnglish,compl,_1234834,_1234836,11,6,7)
** SIMPLE NOUN PHRASE:
conc(pl,_1218264,terc,pronEnglish,compl,_1234834,_1234836,11,6,7)
** PRONOUN:
conc(pl,_1218264,terc,persNoRefl,compl,_1218272,_1218274,11,6,7)
them
** NON-PARSED WORD:
?
<----->
Sentence N§ 12 --> frase_Brown
NUEVA_SERIE.

** PARTIALLY PARSED SENTENCE:
conc(12,0,2)
** NOUN PHRASE:
conc(sing,_1352640,terc,propio,_1342118,_1342120,_1342122,12,0,1)
conc(sing,_1352640,terc,propio,_1342118,_1342120,_1342122,12,0,1)
conc(sing,_1352640,terc,propio,_1342118,_1342120,_1342122,12,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_1352640,terc,propio,_1342118,_1342120,_1342122,12,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_1352640,propio,12,0,1)
** NOUN:

```

```

conc(sing,_1352640,propio,12,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.
<----->
Sentence N§ 13 --> frase_Brown
NUEVA_SERIE.

** PARTIALLY PARSED SENTENCE:
conc(13,0,2)
** NOUN PHRASE:
conc(sing,_1377626,terc,propio,_1366868,_1366870,_1366872,13,0,1)
conc(sing,_1377626,terc,propio,_1366868,_1366870,_1366872,13,0,1)
conc(sing,_1377626,terc,propio,_1366868,_1366870,_1366872,13,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_1377626,terc,propio,_1366868,_1366870,_1366872,13,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_1377626,propio,13,0,1)
** NOUN:
conc(sing,_1377626,propio,13,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.
<----->
Sentence N§ 14 --> frase_Brown
NUEVA_SERIE.

** PARTIALLY PARSED SENTENCE:
conc(14,0,2)
** NOUN PHRASE:
conc(sing,_1402322,terc,propio,_1391734,_1391736,_1391738,14,0,1)
conc(sing,_1402322,terc,propio,_1391734,_1391736,_1391738,14,0,1)
conc(sing,_1402322,terc,propio,_1391734,_1391736,_1391738,14,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_1402322,terc,propio,_1391734,_1391736,_1391738,14,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_1402322,propio,14,0,1)
** NOUN:
conc(sing,_1402322,propio,14,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.
<----->
Sentence N§ 15 --> frase_Brown
I would like to know the QTLs related to rust in wheat.

** PARTIALLY PARSED SENTENCE:
conc(15,0,13)
** NOUN PHRASE:
conc(sing,_1417198,prim,pronEnglish,suj,_1418766,_1418768,15,0,1)
conc(sing,_1417198,prim,pronEnglish,suj,_1418766,_1418768,15,0,1)
conc(sing,_1417198,prim,pronEnglish,suj,_1418766,_1418768,15,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_1417198,prim,pronEnglish,suj,_1418766,_1418768,15,0,1)
** PRONOUN:
conc(sing,_1417198,prim,persNoRefl,suj,_1417206,_1417208,15,0,1)
I
** HEAD OF THE VERBAL PHRASE:
conc(_1439162,_1439164,_1439166,compWill,15,1,5)
conc(_1439162,_1439164,_1439166,compWill,15,1,3)

```

```

** COMPOUND VERB WILL:
conc(_1439162,_1439164,_1439166,compWill,15,1,3)
** VERB:
conc(_1417300,_1417302,xx,auxiliary,15,1,2)
would
** VERB:
conc(_1417398,_1417400,infinitivo,resto,15,2,3)
like
** VERBAL PERIPHRAIS:
conc(15,3,5)
** NON-PARSED WORD:
to
** VERB:
conc(_1417590,_1417592,infinitivo,resto,15,4,5)
know
** NOUN PHRASE:
conc(sing,_1450796,terc,propio,_1439224,_1439226,_1439228,15,5,12)
conc(sing,_1450796,terc,propio,_1439224,_1439226,_1439228,15,5,12)
conc(sing,_1450796,terc,propio,_1439224,_1439226,_1439228,15,5,12)
** SIMPLE NOUN PHRASE:
conc(sing,_1450796,terc,propio,_1439224,_1439226,_1439228,15,5,12)
** DETERMINER 1:
conc(sing,_1450796,det,15,5,6)
** DETERMINER:
conc(sing,_1450796,det,15,5,6)
the
** HEAD OF THE NOUN PHRASE:
conc(sing,_1450796,propio,15,6,7)
** NOUN:
conc(sing,_1450796,propio,15,6,7)
QTLs
** POST-MODIFIER:
conc(sing,_1450796,_1450892,15,7,12)
** RELATIVE SENTENCE:
conc(participio,15,7,12)
** HEAD OF THE VERBAL PHRASE:
conc(_1417880,_1417882,participio,verboSolo,15,7,10)
conc(_1417880,_1417882,participio,verboSolo,15,7,8)
conc(_1417880,_1417882,participio,verboSolo,15,7,8)
** VERB:
conc(_1417880,_1417882,participio,resto,15,7,8)
related
** VERBAL PERIPHRAIS:
conc(15,8,10)
** NON-PARSED WORD:
to
** VERB:
conc(_1418072,_1418074,infinitivo,resto,15,9,10)
rust
** PREPOSITIONAL PHRASE:
conc(_1418170,of,15,10,12)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_1418170,of,15,10,12)
** PREPOSITION:
conc(_1418170,of,15,10,11)
conc(_1418170,of,15,10,11)
** SIMPLE PREPOSITION:
conc(_1418170,resto,15,10,11)
in
** NOUN PHRASE:
conc(sing,_1461088,terc,comun,_1454354,_1454356,_1454358,15,11,12)
conc(sing,_1461088,terc,comun,_1454354,_1454356,_1454358,15,11,12)

```

```

conc(sing,_1461088,terc,comun,_1454354,_1454356,_1454358,15,11,12)
** SIMPLE NOUN PHRASE:
conc(sing,_1461088,terc,comun,_1454354,_1454356,_1454358,15,11,12)
** HEAD OF THE NOUN PHRASE:
conc(sing,_1461088,comun,15,11,12)
** NOUN:
conc(sing,_1461088,comun,15,11,12)
wheat
** NON-PARSED WORD:

```

<----->

Sentence N§ 16 --> frase_Brown

Which genes influence the resistance to plagues in wheat ?

** PARTIALLY PARSED SENTENCE:

conc(16,0,10)

** RELATIVE SENTENCE:

conc(rel,16,0,9)

** PRONOUN:

conc(_38292,_38294,_38296,rel,_38300,_38302,_38304,16,0,1)

Which

** NOUN PHRASE:

conc(pl,_46158,terc,comun,_39718,_39720,_39722,16,1,2)

conc(pl,_46158,terc,comun,_39718,_39720,_39722,16,1,2)

conc(pl,_46158,terc,comun,_39718,_39720,_39722,16,1,2)

** SIMPLE NOUN PHRASE:

conc(pl,_46158,terc,comun,_39718,_39720,_39722,16,1,2)

** HEAD OF THE NOUN PHRASE:

conc(pl,_46158,comun,16,1,2)

** NOUN:

conc(pl,_46158,comun,16,1,2)

genes

** HEAD OF THE VERBAL PHRASE:

conc(_38492,_38494,presente,_39740,16,2,3)

conc(_38492,_38494,presente,_39740,16,2,3)

conc(_38492,_38494,presente,verboSolo,16,2,3)

** VERB:

conc(_38492,_38494,presente,resto,16,2,3)

influence

** NOUN PHRASE:

conc(sing,_62234,terc,comun,_39782,_39784,_39786,16,3,9)

conc(sing,_62234,terc,comun,_39782,_39784,_39786,16,3,9)

conc(sing,_62234,terc,comun,_39782,_39784,_39786,16,3,9)

** SIMPLE NOUN PHRASE:

conc(sing,_62234,terc,comun,_39782,_39784,_39786,16,3,9)

** DETERMINER 1:

conc(sing,_62234,det,16,3,4)

** DETERMINER:

conc(sing,_62234,det,16,3,4)

the

** HEAD OF THE NOUN PHRASE:

conc(sing,_62234,comun,16,4,5)

** NOUN:

conc(sing,_62234,comun,16,4,5)

resistance

** PREPOSITIONAL PHRASE:

conc(_38782,of,16,5,9)

** SIMPLE PREPOSITIONAL PHRASE:

conc(_38782,of,16,5,9)

** PREPOSITION:


```

conc(_38782,of,16,5,6)
conc(_38782,of,16,5,6)
** SIMPLE PREPOSITION:
  conc(_38782,to,16,5,6)
  to
** NOUN PHRASE:
conc(pl,_73648,terc,comun,_67068,_67070,_67072,16,6,9)
conc(pl,_73648,terc,comun,_67068,_67070,_67072,16,6,9)
conc(pl,_73648,terc,comun,_67068,_67070,_67072,16,6,9)
** SIMPLE NOUN PHRASE:
conc(pl,_73648,terc,comun,_67068,_67070,_67072,16,6,9)
** HEAD OF THE NOUN PHRASE:
  conc(pl,_73648,comun,16,6,7)
  ** NOUN:
    conc(pl,_73648,comun,16,6,7)
    plagues
** PREPOSITIONAL PHRASE:
conc(_38972,of,16,7,9)
** SIMPLE PREPOSITIONAL PHRASE:
  conc(_38972,of,16,7,9)
  ** PREPOSITION:
    conc(_38972,of,16,7,8)
    conc(_38972,of,16,7,8)
  ** SIMPLE PREPOSITION:
    conc(_38972,resto,16,7,8)
    in
  ** NOUN PHRASE:
    conc(sing,_86624,terc,comun,_80044,_80046,_80048,16,8,9)
    conc(sing,_86624,terc,comun,_80044,_80046,_80048,16,8,9)
    conc(sing,_86624,terc,comun,_80044,_80046,_80048,16,8,9)
  ** SIMPLE NOUN PHRASE:
    conc(sing,_86624,terc,comun,_80044,_80046,_80048,16,8,9)
  ** HEAD OF THE NOUN PHRASE:
    conc(sing,_86624,comun,16,8,9)
  ** NOUN:
    conc(sing,_86624,comun,16,8,9)
    wheat

** NON-PARSED WORD:
  ?
<----->
  Sentence N§ 17 --> frase_Brown
Are there other plagues in wheat ?

** PARTIALLY PARSED SENTENCE:
conc(17,0,7)
** HEAD OF THE VERBAL PHRASE:
  conc(_175326,_175328,presente,_176246,17,0,2)
  conc(_175326,_175328,presente,_176246,17,0,1)
  conc(_175326,_175328,presente,verboSolo,17,0,1)
** VERB:
  conc(_175326,_175328,presente,be,17,0,1)
  Are
** ADVERB:
  conc(general,17,1,2)
  there
** NOUN PHRASE:
conc(pl,_187704,terc,comun,_176302,_176304,_176306,17,2,6)
conc(pl,_187704,terc,comun,_176302,_176304,_176306,17,2,6)
conc(pl,_187704,terc,comun,_176302,_176304,_176306,17,2,6)
** SIMPLE NOUN PHRASE:
  conc(pl,_187704,terc,comun,_176302,_176304,_176306,17,2,6)
** PRE-MODIFIER:

```

```

conc(pl,_187704,adj,17,2,3)
** ADJECTIVAL PHRASE:
conc(pl,_187704,17,2,3)
** COORDINATED ADJECTIVE:
conc(pl,_187704,_190486,17,2,3)
** SIMPLE ADJECTIVE:
conc(pl,_187704,cal,17,2,3)
other
** HEAD OF THE NOUN PHRASE:
conc(pl,_187704,comun,17,3,4)
** NOUN:
conc(pl,_187704,comun,17,3,4)
plagues
** PREPOSITIONAL PHRASE:
conc(_175708,of,17,4,6)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_175708,of,17,4,6)
** PREPOSITION:
conc(_175708,of,17,4,5)
conc(_175708,of,17,4,5)
** SIMPLE PREPOSITION:
conc(_175708,resto,17,4,5)
in
** NOUN PHRASE:
conc(sing,_200978,terc,comun,_194398,_194400,_194402,17,5,6)
conc(sing,_200978,terc,comun,_194398,_194400,_194402,17,5,6)
conc(sing,_200978,terc,comun,_194398,_194400,_194402,17,5,6)
** SIMPLE NOUN PHRASE:
conc(sing,_200978,terc,comun,_194398,_194400,_194402,17,5,6)
** HEAD OF THE NOUN PHRASE:
conc(sing,_200978,comun,17,5,6)
** NOUN:
conc(sing,_200978,comun,17,5,6)
wheat
** NON-PARSED WORD:
?
```

<----->

Sentence N§ 18 --> frase_Brown
NUEVA_SERIE.

```

** PARTIALLY PARSED SENTENCE:
conc(18,0,2)
** NOUN PHRASE:
conc(sing,_273896,terc,propio,_263374,_263376,_263378,18,0,1)
conc(sing,_273896,terc,propio,_263374,_263376,_263378,18,0,1)
conc(sing,_273896,terc,propio,_263374,_263376,_263378,18,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_273896,terc,propio,_263374,_263376,_263378,18,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_273896,propio,18,0,1)
** NOUN:
conc(sing,_273896,propio,18,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
```

<----->

Sentence N§ 19 --> frase_Brown
NUEVA_SERIE.

```

** PARTIALLY PARSED SENTENCE:
```

```

conc(19,0,2)
** NOUN PHRASE:
conc(sing,_297640,terc,propio,_287162,_287164,_287166,19,0,1)
conc(sing,_297640,terc,propio,_287162,_287164,_287166,19,0,1)
conc(sing,_297640,terc,propio,_287162,_287164,_287166,19,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_297640,terc,propio,_287162,_287164,_287166,19,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_297640,propio,19,0,1)
** NOUN:
conc(sing,_297640,propio,19,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.
```

<----->

Sentence N§ 20 --> frase_Brown
NUEVA_SERIE.

```

** PARTIALLY PARSED SENTENCE:
conc(20,0,2)
** NOUN PHRASE:
conc(sing,_321380,terc,propio,_310902,_310904,_310906,20,0,1)
conc(sing,_321380,terc,propio,_310902,_310904,_310906,20,0,1)
conc(sing,_321380,terc,propio,_310902,_310904,_310906,20,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_321380,terc,propio,_310902,_310904,_310906,20,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_321380,propio,20,0,1)
** NOUN:
conc(sing,_321380,propio,20,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.
```

<----->

Sentence N§ 21 --> frase_Brown
What QTLs are related to the kernel weight in barley ?

```

** PARTIALLY PARSED SENTENCE:
conc(21,0,11)
** RELATIVE SENTENCE:
conc(rel,21,0,10)
** PRONOUN:
conc(_335888,_335890,_335892,rel,_335896,_335898,_335900,21,0,1)
What
** NOUN PHRASE:
conc(sing,_343852,terc,propio,_337412,_337414,_337416,21,1,2)
conc(sing,_343852,terc,propio,_337412,_337414,_337416,21,1,2)
conc(sing,_343852,terc,propio,_337412,_337414,_337416,21,1,2)
** SIMPLE NOUN PHRASE:
conc(sing,_343852,terc,propio,_337412,_337414,_337416,21,1,2)
** HEAD OF THE NOUN PHRASE:
conc(sing,_343852,propio,21,1,2)
** NOUN:
conc(sing,_343852,propio,21,1,2)
QTLs
** HEAD OF THE VERBAL PHRASE:
conc(_336088,_336090,pasiva,compBe,21,2,4)
conc(_336088,_336090,pasiva,compBe,21,2,4)
** COMPOUND VERB BE:
conc(_336088,_336090,pasiva,compBe,21,2,4)
** VERB:
conc(_336088,_336090,presente,be,21,2,3)
```

```

are
** VERB:
  conc(_336186,_336188,participio,resto,21,3,4)
  related
** PREPOSITIONAL PHRASE:
  conc(_336284,of,21,4,10)
** SIMPLE PREPOSITIONAL PHRASE:
  conc(_336284,of,21,4,10)
** PREPOSITION:
  conc(_336284,of,21,4,5)
  conc(_336284,of,21,4,5)
** SIMPLE PREPOSITION:
  conc(_336284,to,21,4,5)
  to
** NOUN PHRASE:
  conc(sing,_359542,terc,comun,_353032,_353034,_353036,21,5,10)
  conc(sing,_359542,terc,comun,_353032,_353034,_353036,21,5,10)
  conc(sing,_359542,terc,comun,_353032,_353034,_353036,21,5,10)
** SIMPLE NOUN PHRASE:
  conc(sing,_359542,terc,comun,_353032,_353034,_353036,21,5,10)
** DETERMINER 1:
  conc(sing,_359542,det,21,5,6)
** DETERMINER:
  conc(sing,_359542,det,21,5,6)
  the
** HEAD OF THE NOUN PHRASE:
  conc(sing,_359542,comun,21,6,8)
** NOUN:
  conc(sing,_336476,comun,21,6,7)
  kernel
** NOUN:
  conc(sing,_359542,comun,21,7,8)
  weight
** PREPOSITIONAL PHRASE:
  conc(_336666,of,21,8,10)
** SIMPLE PREPOSITIONAL PHRASE:
  conc(_336666,of,21,8,10)
** PREPOSITION:
  conc(_336666,of,21,8,9)
  conc(_336666,of,21,8,9)
** SIMPLE PREPOSITION:
  conc(_336666,resto,21,8,9)
  in
** NOUN PHRASE:
  conc(sing,_370790,terc,comun,_364210,_364212,_364214,21,9,10)
  conc(sing,_370790,terc,comun,_364210,_364212,_364214,21,9,10)
  conc(sing,_370790,terc,comun,_364210,_364212,_364214,21,9,10)
** SIMPLE NOUN PHRASE:
  conc(sing,_370790,terc,comun,_364210,_364212,_364214,21,9,10)
** HEAD OF THE NOUN PHRASE:
  conc(sing,_370790,comun,21,9,10)
** NOUN:
  conc(sing,_370790,comun,21,9,10)
  barley
** NON-PARSED WORD:
  ?
<----->
Sentence N§ 22 --> frase_Brown
Are there any biotechnology companies that havemade Genetic Engineering with these QTLs in barley ?

```

** PARTIALLY PARSED SENTENCE:

conc(22,0,15)

** HEAD OF THE VERBAL PHRASE:

conc(_475562,_475564,presente,_477256,22,0,2)

conc(_475562,_475564,presente,_477256,22,0,1)

conc(_475562,_475564,presente,verboSolo,22,0,1)

** VERB:

conc(_475562,_475564,presente,be,22,0,1)

Are

** ADVERB:

conc(general,22,1,2)

there

** NOUN PHRASE:

conc(pl,_488574,terc,comun,_477312,_477314,_477316,22,2,5)

conc(pl,_488574,terc,comun,_477312,_477314,_477316,22,2,5)

conc(pl,_488574,terc,comun,_477312,_477314,_477316,22,2,5)

** SIMPLE NOUN PHRASE:

conc(pl,_488574,terc,comun,_477312,_477314,_477316,22,2,5)

** DETERMINER 1:

conc(pl,_488574,det,22,2,3)

** DETERMINER:

conc(pl,_488574,det,22,2,3)

any

** HEAD OF THE NOUN PHRASE:

conc(pl,_488574,comun,22,3,5)

** NOUN:

conc(sing,_475850,comun,22,3,4)

biotechnology

** NOUN:

conc(pl,_488574,comun,22,4,5)

companies

** CONJUNCTION:

conc(22,5,6)

that

** NOUN PHRASE:

conc(sing,_504568,terc,propio,_494060,_494062,_494064,22,6,14)

conc(sing,_504568,terc,propio,_494060,_494062,_494064,22,6,14)

conc(sing,_504568,terc,propio,_494060,_494062,_494064,22,6,14)

** SIMPLE NOUN PHRASE:

conc(sing,_504568,terc,propio,_494060,_494062,_494064,22,6,14)

** PRE-MODIFIER:

conc(sing,_504568,adj,22,6,7)

** ADJECTIVAL PHRASE:

conc(sing,_504568,22,6,7)

** COORDINATED ADJECTIVE:

conc(sing,_504568,_507350,22,6,7)

** SIMPLE ADJECTIVE:

conc(sing,_504568,cal,22,6,7)

havemade

** HEAD OF THE NOUN PHRASE:

conc(sing,_504568,propio,22,7,9)

** NOUN:

conc(sing,_476226,propio,22,7,8)

Genetic

** NOUN:

conc(sing,_504568,propio,22,8,9)

Engineering

** PREPOSITIONAL PHRASE:

conc(_476416,of,22,9,14)

** SIMPLE PREPOSITIONAL PHRASE:

conc(_476416,of,22,9,14)

** PREPOSITION:

```

conc(_476416,of,22,9,10)
conc(_476416,of,22,9,10)
** SIMPLE PREPOSITION:
    conc(_476416,resto,22,9,10)
    with
** NOUN PHRASE:
conc(sing,_517954,terc,propio,_511444,_511446,_511448,22,10,14)
conc(sing,_517954,terc,propio,_511444,_511446,_511448,22,10,14)
conc(sing,_517954,terc,propio,_511444,_511446,_511448,22,10,14)
** SIMPLE NOUN PHRASE:
conc(sing,_517954,terc,propio,_511444,_511446,_511448,22,10,14)
** DETERMINER 1:
conc(sing,_517954,det,22,10,11)
** DETERMINER:
    conc(sing,_517954,det,22,10,11)
    these
** HEAD OF THE NOUN PHRASE:
conc(sing,_517954,propio,22,11,12)
** NOUN:
    conc(sing,_517954,propio,22,11,12)
    QTLs
** PREPOSITIONAL PHRASE:
conc(_476718,of,22,12,14)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_476718,of,22,12,14)
** PREPOSITION:
    conc(_476718,of,22,12,13)
    conc(_476718,of,22,12,13)
** SIMPLE PREPOSITION:
    conc(_476718,resto,22,12,13)
    in
** NOUN PHRASE:
conc(sing,_529020,terc,comun,_522440,_522442,_522444,22,13,14)
conc(sing,_529020,terc,comun,_522440,_522442,_522444,22,13,14)
conc(sing,_529020,terc,comun,_522440,_522442,_522444,22,13,14)
** SIMPLE NOUN PHRASE:
conc(sing,_529020,terc,comun,_522440,_522442,_522444,22,13,14)
** HEAD OF THE NOUN PHRASE:
conc(sing,_529020,comun,22,13,14)
** NOUN:
    conc(sing,_529020,comun,22,13,14)
    barley
** NON-PARSED WORD:
    ?
<----->
    Sentence N§ 23 --> frase_Brown
Are there currently any transgenic barley varieties in themarket ?

** PARTIALLY PARSED SENTENCE:
conc(23,0,10)
** HEAD OF THE VERBAL PHRASE:
conc(_602400,_602402,presente,_603604,23,0,2)
conc(_602400,_602402,presente,_603604,23,0,1)
conc(_602400,_602402,presente,verboSolo,23,0,1)
** VERB:
    conc(_602400,_602402,presente,be,23,0,1)
    Are
** ADVERB:
    conc(general,23,1,2)
    there

```

```

** NON-PARSED WORD:
currently
** NOUN PHRASE:
conc(pl,_655266,terc,comun,_644858,_644860,_644862,23,3,9)
conc(pl,_655266,terc,comun,_644858,_644860,_644862,23,3,9)
conc(pl,_655266,terc,comun,_644858,_644860,_644862,23,3,9)
** SIMPLE NOUN PHRASE:
conc(pl,_655266,terc,comun,_644858,_644860,_644862,23,3,9)
** DETERMINER 1:
conc(pl,_655266,det,23,3,4)
** DETERMINER:
conc(pl,_655266,det,23,3,4)
any
** HEAD OF THE NOUN PHRASE:
conc(pl,_655266,comun,23,4,7)
** NOUN:
conc(sing,_602780,propio,23,4,5)
transgenic
** NOUN:
conc(sing,_602876,comun,23,5,6)
barley
** NOUN:
conc(pl,_655266,comun,23,6,7)
varieties
** PREPOSITIONAL PHRASE:
conc(_603066,of,23,7,9)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_603066,of,23,7,9)
** PREPOSITION:
conc(_603066,of,23,7,8)
conc(_603066,of,23,7,8)
** SIMPLE PREPOSITION:
conc(_603066,resto,23,7,8)
in
** NOUN PHRASE:
conc(sing,_666696,terc,comun,_660116,_660118,_660120,23,8,9)
conc(sing,_666696,terc,comun,_660116,_660118,_660120,23,8,9)
conc(sing,_666696,terc,comun,_660116,_660118,_660120,23,8,9)
** SIMPLE NOUN PHRASE:
conc(sing,_666696,terc,comun,_660116,_660118,_660120,23,8,9)
** HEAD OF THE NOUN PHRASE:
conc(sing,_666696,comun,23,8,9)
** NOUN:
conc(sing,_666696,comun,23,8,9)
themarket
** NON-PARSED WORD:
?

```

```

<----->
Sentence N§ 24 --> frase_Brown
NUEVA_SERIE.

```

```

** PARTIALLY PARSED SENTENCE:
conc(24,0,2)
** NOUN PHRASE:
conc(sing,_740086,terc,propio,_729608,_729610,_729612,24,0,1)
conc(sing,_740086,terc,propio,_729608,_729610,_729612,24,0,1)
conc(sing,_740086,terc,propio,_729608,_729610,_729612,24,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_740086,terc,propio,_729608,_729610,_729612,24,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_740086,propio,24,0,1)
** NOUN:

```

```

conc(sing,_740086,propio,24,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.
<----->
Sentence N§ 25 --> frase_Brown
NUEVA_SERIE.

** PARTIALLY PARSED SENTENCE:
conc(25,0,2)
** NOUN PHRASE:
conc(sing,_763826,terc,propio,_753348,_753350,_753352,25,0,1)
conc(sing,_763826,terc,propio,_753348,_753350,_753352,25,0,1)
conc(sing,_763826,terc,propio,_753348,_753350,_753352,25,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_763826,terc,propio,_753348,_753350,_753352,25,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_763826,propio,25,0,1)
** NOUN:
conc(sing,_763826,propio,25,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.
<----->
Sentence N§ 26 --> frase_Brown
NUEVA_SERIE.

** PARTIALLY PARSED SENTENCE:
conc(26,0,2)
** NOUN PHRASE:
conc(sing,_787566,terc,propio,_777088,_777090,_777092,26,0,1)
conc(sing,_787566,terc,propio,_777088,_777090,_777092,26,0,1)
conc(sing,_787566,terc,propio,_777088,_777090,_777092,26,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_787566,terc,propio,_777088,_777090,_777092,26,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_787566,propio,26,0,1)
** NOUN:
conc(sing,_787566,propio,26,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.
<----->
Sentence N§ 27 --> frase_Brown
What QTLs are related to frost tolerance and resistance to Fusarium in barley ?

** PARTIALLY PARSED SENTENCE:
conc(27,0,14)
** RELATIVE SENTENCE:
conc(rel,27,0,13)
** PRONOUN:
conc(_802650,_802652,_802654,rel,_802658,_802660,_802662,27,0,1)
What
** NOUN PHRASE:
conc(sing,_810894,terc,propio,_804454,_804456,_804458,27,1,2)
conc(sing,_810894,terc,propio,_804454,_804456,_804458,27,1,2)
conc(sing,_810894,terc,propio,_804454,_804456,_804458,27,1,2)
** SIMPLE NOUN PHRASE:
conc(sing,_810894,terc,propio,_804454,_804456,_804458,27,1,2)
** HEAD OF THE NOUN PHRASE:

```



```

    conc(sing,_810894,propio,27,1,2)
    ** NOUN:
        conc(sing,_810894,propio,27,1,2)
        QTLs
** HEAD OF THE VERBAL PHRASE:
    conc(_802850,_802852,pasiva,compBe,27,2,6)
    conc(_802850,_802852,pasiva,compBe,27,2,4)
** COMPOUND VERB BE:
    conc(_802850,_802852,pasiva,compBe,27,2,4)
** VERB:
    conc(_802850,_802852,presente,be,27,2,3)
    are
** VERB:
    conc(_802948,_802950,participio,resto,27,3,4)
    related
** VERBAL PERIPHRAISIS:
    conc(27,4,6)
** NON-PARSED WORD:
    to
** VERB:
    conc(_803140,_803142,infinitivo,resto,27,5,6)
    frost
** NOUN PHRASE:
    conc(pl,masc,terc,comun,_874534,_874452,_874454,27,6,13)
    conc(sing,_874444,terc,comun,_874534,_874452,_874454,27,6,13)
    conc(sing,_874444,terc,comun,_874534,_874452,_874454,27,6,7)
** SIMPLE NOUN PHRASE:
    conc(sing,_874444,terc,comun,_874534,_874452,_874454,27,6,7)
** HEAD OF THE NOUN PHRASE:
    conc(sing,_874444,comun,27,6,7)
    ** NOUN:
        conc(sing,_874444,comun,27,6,7)
        tolerance
** CONJUNCTION:
    conc(27,7,8)
    and
    conc(_834072,_834018,_833964,_833910,_833856,_833802,_833748,27,8,13)
    conc(sing,_874528,terc,comun,_874534,_874536,_874538,27,8,13)
** SIMPLE NOUN PHRASE:
    conc(sing,_874528,terc,comun,_874534,_874536,_874538,27,8,13)
** HEAD OF THE NOUN PHRASE:
    conc(sing,_874528,comun,27,8,9)
    ** NOUN:
        conc(sing,_874528,comun,27,8,9)
        resistance
** PREPOSITIONAL PHRASE:
    conc(_803518,of,27,9,13)
** SIMPLE PREPOSITIONAL PHRASE:
    conc(_803518,of,27,9,13)
    ** PREPOSITION:
        conc(_803518,of,27,9,10)
        conc(_803518,of,27,9,10)
    ** SIMPLE PREPOSITION:
        conc(_803518,to,27,9,10)
        to
** NOUN PHRASE:
    conc(sing,_854138,terc,propio,_847558,_847560,_847562,27,10,13)
    conc(sing,_854138,terc,propio,_847558,_847560,_847562,27,10,13)
    conc(sing,_854138,terc,propio,_847558,_847560,_847562,27,10,13)
** SIMPLE NOUN PHRASE:
    conc(sing,_854138,terc,propio,_847558,_847560,_847562,27,10,13)
** HEAD OF THE NOUN PHRASE:

```

```

conc(sing,_854138,propio,27,10,11)
  ** NOUN:
    conc(sing,_854138,propio,27,10,11)
    Fusarium
** PREPOSITIONAL PHRASE:
conc(_803708,of,27,11,13)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_803708,of,27,11,13)
  ** PREPOSITION:
    conc(_803708,of,27,11,12)
    conc(_803708,of,27,11,12)
  ** SIMPLE PREPOSITION:
    conc(_803708,resto,27,11,12)
    in
  ** NOUN PHRASE:
    conc(sing,_867114,terc,comun,_860534,_860536,_860538,27,12,13)
    conc(sing,_867114,terc,comun,_860534,_860536,_860538,27,12,13)
    conc(sing,_867114,terc,comun,_860534,_860536,_860538,27,12,13)
  ** SIMPLE NOUN PHRASE:
    conc(sing,_867114,terc,comun,_860534,_860536,_860538,27,12,13)
  ** HEAD OF THE NOUN PHRASE:
    conc(sing,_867114,comun,27,12,13)
  ** NOUN:
    conc(sing,_867114,comun,27,12,13)
    barley

** NON-PARSED WORD:
?
<----->
Sentence N§ 28 --> frase_Brown
Since there is no common QTL, are there any studies showing these traits are related ?

** PARTIALLY PARSED SENTENCE:
conc(28,0,17)
** NON-PARSED WORD:
Since
** NOUN PHRASE:
conc(_1000022,_1000024,terc,there,_1000030,_1000032,_1000034,28,1,2)
** NON-PARSED WORD:
there
** HEAD OF THE VERBAL PHRASE:
conc(sing,terc,presente,_1004504,28,2,3)
conc(sing,terc,presente,_1004504,28,2,3)
conc(sing,terc,presente,verboSolo,28,2,3)
** VERB:
conc(sing,terc,presente,be,28,2,3)
is
** NOUN PHRASE:
conc(sing,_1015974,terc,propio,_1004560,_1004562,_1004564,28,3,6)
conc(sing,_1015974,terc,propio,_1004560,_1004562,_1004564,28,3,6)
conc(sing,_1015974,terc,propio,_1004560,_1004562,_1004564,28,3,6)
** SIMPLE NOUN PHRASE:
conc(sing,_1015974,terc,propio,_1004560,_1004562,_1004564,28,3,6)
** DETERMINER 1:
conc(sing,_1015974,det,28,3,4)
** DETERMINER:
conc(sing,_1015974,det,28,3,4)
no
** PRE-MODIFIER:
conc(sing,_1015974,adj,28,4,5)
** ADJECTIVAL PHRASE:

```

```

conc(sing,_1015974,28,4,5)
** COORDINATED ADJECTIVE:
conc(sing,_1015974,_1016846,28,4,5)
** SIMPLE ADJECTIVE:
conc(sing,_1015974,cal,28,4,5)
common
** HEAD OF THE NOUN PHRASE:
conc(sing,_1015974,propio,28,5,6)
** NOUN:
conc(sing,_1015974,propio,28,5,6)
QTL
** CONJUNCTION:
conc(28,6,7)
,
** HEAD OF THE VERBAL PHRASE:
conc(_961982,_961984,presente,_1075086,28,7,9)
conc(_961982,_961984,presente,_1075086,28,7,8)
conc(_961982,_961984,presente,verboSolo,28,7,8)
** VERB:
conc(_961982,_961984,presente,be,28,7,8)
are
** ADVERB:
conc(general,28,8,9)
there
** NOUN PHRASE:
conc(pl,_1086404,terc,comun,_1075142,_1075144,_1075146,28,9,14)
conc(pl,_1086404,terc,comun,_1075142,_1075144,_1075146,28,9,14)
conc(pl,_1086404,terc,comun,_1075142,_1075144,_1075146,28,9,14)
** SIMPLE NOUN PHRASE:
conc(pl,_1086404,terc,comun,_1075142,_1075144,_1075146,28,9,14)
** DETERMINER 1:
conc(pl,_1086404,det,28,9,10)
** DETERMINER:
conc(pl,_1086404,det,28,9,10)
any
** HEAD OF THE NOUN PHRASE:
conc(pl,_1086404,comun,28,10,11)
** NOUN:
conc(pl,_1086404,comun,28,10,11)
studies
** POST-MODIFIER:
conc(pl,_1086404,_1086500,28,11,14)
** RELATIVE SENTENCE:
conc(participio,28,11,14)
** HEAD OF THE VERBAL PHRASE:
conc(_962364,_962366,gerundio,verboSolo,28,11,12)
conc(_962364,_962366,gerundio,verboSolo,28,11,12)
conc(_962364,_962366,gerundio,verboSolo,28,11,12)
** VERB:
conc(_962364,_962366,gerundio,resto,28,11,12)
showing
** NOUN PHRASE:
conc(pl,_1096854,terc,comun,_1088826,_1088828,_1088830,28,12,14)
conc(pl,_1096854,terc,comun,_1088826,_1088828,_1088830,28,12,14)
conc(pl,_1096854,terc,comun,_1088826,_1088828,_1088830,28,12,14)
** SIMPLE NOUN PHRASE:
conc(pl,_1096854,terc,comun,_1088826,_1088828,_1088830,28,12,14)
** DETERMINER 1:
conc(pl,_1096854,det,28,12,13)
** DETERMINER:
conc(pl,_1096854,det,28,12,13)
these

```

```

** HEAD OF THE NOUN PHRASE:
  conc(pl,_1096854,comun,28,13,14)
** NOUN:
  conc(pl,_1096854,comun,28,13,14)
  traits
** HEAD OF THE VERBAL PHRASE:
  conc(_962670,_962672,pasiva,compBe,28,14,16)
  conc(_962670,_962672,pasiva,compBe,28,14,16)
** COMPOUND VERB BE:
  conc(_962670,_962672,pasiva,compBe,28,14,16)
** VERB:
  conc(_962670,_962672,presente,be,28,14,15)
  are
** VERB:
  conc(_962768,_962770,participio,resto,28,15,16)
  related
** NON-PARSED WORD:
  ?
<----->
  Sentence N§ 29 --> frase_Brown
NUEVA_SERIE.

** PARTIALLY PARSED SENTENCE:
  conc(29,0,2)
** NOUN PHRASE:
  conc(sing,_1199906,terc,propio,_1189428,_1189430,_1189432,29,0,1)
  conc(sing,_1199906,terc,propio,_1189428,_1189430,_1189432,29,0,1)
  conc(sing,_1199906,terc,propio,_1189428,_1189430,_1189432,29,0,1)
** SIMPLE NOUN PHRASE:
  conc(sing,_1199906,terc,propio,_1189428,_1189430,_1189432,29,0,1)
** HEAD OF THE NOUN PHRASE:
  conc(sing,_1199906,propio,29,0,1)
** NOUN:
  conc(sing,_1199906,propio,29,0,1)
  NUEVA_SERIE
** NON-PARSED WORD:
  .
<----->
  Sentence N§ 30 --> frase_Brown
NUEVA_SERIE.

** PARTIALLY PARSED SENTENCE:
  conc(30,0,2)
** NOUN PHRASE:
  conc(sing,_1223646,terc,propio,_1213168,_1213170,_1213172,30,0,1)
  conc(sing,_1223646,terc,propio,_1213168,_1213170,_1213172,30,0,1)
  conc(sing,_1223646,terc,propio,_1213168,_1213170,_1213172,30,0,1)
** SIMPLE NOUN PHRASE:
  conc(sing,_1223646,terc,propio,_1213168,_1213170,_1213172,30,0,1)
** HEAD OF THE NOUN PHRASE:
  conc(sing,_1223646,propio,30,0,1)
** NOUN:
  conc(sing,_1223646,propio,30,0,1)
  NUEVA_SERIE
** NON-PARSED WORD:
  .
<----->
  Sentence N§ 31 --> frase_Brown
NUEVA_SERIE.

```

** PARTIALLY PARSED SENTENCE:

conc(31,0,2)

** NOUN PHRASE:

conc(sing,_1247386,terc,propio,_1236908,_1236910,_1236912,31,0,1)

conc(sing,_1247386,terc,propio,_1236908,_1236910,_1236912,31,0,1)

conc(sing,_1247386,terc,propio,_1236908,_1236910,_1236912,31,0,1)

** SIMPLE NOUN PHRASE:

conc(sing,_1247386,terc,propio,_1236908,_1236910,_1236912,31,0,1)

** HEAD OF THE NOUN PHRASE:

conc(sing,_1247386,propio,31,0,1)

** NOUN:

conc(sing,_1247386,propio,31,0,1)

NUEVA_SERIE

** NON-PARSED WORD:

.

<----->

Sentence N§ 32 --> frase_Brown

Are there currently any transgenic wheat ?

** PARTIALLY PARSED SENTENCE:

conc(32,0,7)

** HEAD OF THE VERBAL PHRASE:

conc(_1261126,_1261128,presente,_1262044,32,0,2)

conc(_1261126,_1261128,presente,_1262044,32,0,1)

conc(_1261126,_1261128,presente,verboSolo,32,0,1)

** VERB:

conc(_1261126,_1261128,presente,be,32,0,1)

Are

** ADVERB:

conc(general,32,1,2)

there

** NON-PARSED WORD:

currently

** NOUN PHRASE:

conc(sing,_1313706,terc,comun,_1303298,_1303300,_1303302,32,3,6)

conc(sing,_1313706,terc,comun,_1303298,_1303300,_1303302,32,3,6)

conc(sing,_1313706,terc,comun,_1303298,_1303300,_1303302,32,3,6)

** SIMPLE NOUN PHRASE:

conc(sing,_1313706,terc,comun,_1303298,_1303300,_1303302,32,3,6)

** DETERMINER 1:

conc(sing,_1313706,det,32,3,4)

** DETERMINER:

conc(sing,_1313706,det,32,3,4)

any

** HEAD OF THE NOUN PHRASE:

conc(sing,_1313706,comun,32,4,6)

** NOUN:

conc(sing,_1261506,propio,32,4,5)

transgenic

** NOUN:

conc(sing,_1313706,comun,32,5,6)

wheat

** NON-PARSED WORD:

?

<----->

Sentence N§ 33 --> frase_Brown

Is it in the market ?

** PARTIALLY PARSED SENTENCE:

conc(33,0,6)

** HEAD OF THE VERBAL PHRASE:

conc(sing,terc,presente,_1370962,33,0,1)

```

conc(sing,terc,presente,_1370962,33,0,1)
conc(sing,terc,presente,verboSolo,33,0,1)
** VERB:
  conc(sing,terc,presente,be,33,0,1)
  Is
** NOUN PHRASE:
conc(sing,_1370226,terc,pronEnglish,_1370232,_1371020,_1371022,33,1,2)
conc(sing,_1370226,terc,pronEnglish,_1370232,_1371020,_1371022,33,1,2)
conc(sing,_1370226,terc,pronEnglish,_1370232,_1371020,_1371022,33,1,2)
** SIMPLE NOUN PHRASE:
  conc(sing,_1370226,terc,pronEnglish,_1370232,_1371020,_1371022,33,1,2)
** PRONOUN:
  conc(sing,_1370226,terc,persIt,_1370232,_1370234,_1370236,33,1,2)
  it
** PREPOSITIONAL PHRASE:
conc(_1370328,of,33,2,5)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_1370328,of,33,2,5)
** PREPOSITION:
  conc(_1370328,of,33,2,3)
  conc(_1370328,of,33,2,3)
** SIMPLE PREPOSITION:
  conc(_1370328,resto,33,2,3)
  in
** NOUN PHRASE:
conc(sing,_1402652,terc,comun,_1396142,_1396144,_1396146,33,3,5)
conc(sing,_1402652,terc,comun,_1396142,_1396144,_1396146,33,3,5)
conc(sing,_1402652,terc,comun,_1396142,_1396144,_1396146,33,3,5)
** SIMPLE NOUN PHRASE:
  conc(sing,_1402652,terc,comun,_1396142,_1396144,_1396146,33,3,5)
** DETERMINER 1:
  conc(sing,_1402652,det,33,3,4)
** DETERMINER:
  conc(sing,_1402652,det,33,3,4)
  the
** HEAD OF THE NOUN PHRASE:
  conc(sing,_1402652,comun,33,4,5)
** NOUN:
  conc(sing,_1402652,comun,33,4,5)
  market
** NON-PARSED WORD:
  ?
<----->
Sentence N§ 34 --> frase_Brown
What is its price ?

** PARTIALLY PARSED SENTENCE:
conc(34,0,5)
** RELATIVE SENTENCE:
conc(rel,34,0,4)
** PRONOUN:
  conc(_12270,_12272,_12274,rel,_12278,_12280,_12282,34,0,1)
  What
** HEAD OF THE VERBAL PHRASE:
  conc(sing,terc,presente,_13268,34,1,2)
  conc(sing,terc,presente,_13268,34,1,2)
  conc(sing,terc,presente,verboSolo,34,1,2)
** VERB:
  conc(sing,terc,presente,be,34,1,2)
  is

```

```

** NOUN PHRASE:
conc(sing,_39400,terc,comun,_13310,_13312,_13314,34,2,4)
conc(sing,_39400,terc,comun,_13310,_13312,_13314,34,2,4)
conc(sing,_39400,terc,comun,_13310,_13312,_13314,34,2,4)
** SIMPLE NOUN PHRASE:
conc(sing,_39400,terc,comun,_13310,_13312,_13314,34,2,4)
** DETERMINER 1:
conc(sing,_39400,pos,34,2,3)
** PRONOUN:
conc(sing,_39400,terc,pos,_12480,_12482,_12484,34,2,3)
its
** HEAD OF THE NOUN PHRASE:
conc(sing,_39400,comun,34,3,4)
** NOUN:
conc(sing,_39400,comun,34,3,4)
price
** NON-PARSED WORD:
?

```

<----->

Sentence N§ 35 --> frase_Brown
NUEVA_SERIE.

```

** PARTIALLY PARSED SENTENCE:
conc(35,0,2)
** NOUN PHRASE:
conc(sing,_124852,terc,propio,_114374,_114376,_114378,35,0,1)
conc(sing,_124852,terc,propio,_114374,_114376,_114378,35,0,1)
conc(sing,_124852,terc,propio,_114374,_114376,_114378,35,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_124852,terc,propio,_114374,_114376,_114378,35,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_124852,propio,35,0,1)
** NOUN:
conc(sing,_124852,propio,35,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.

```

<----->

Sentence N§ 36 --> frase_Brown
NUEVA_SERIE.

```

** PARTIALLY PARSED SENTENCE:
conc(36,0,2)
** NOUN PHRASE:
conc(sing,_148592,terc,propio,_138114,_138116,_138118,36,0,1)
conc(sing,_148592,terc,propio,_138114,_138116,_138118,36,0,1)
conc(sing,_148592,terc,propio,_138114,_138116,_138118,36,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_148592,terc,propio,_138114,_138116,_138118,36,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_148592,propio,36,0,1)
** NOUN:
conc(sing,_148592,propio,36,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.

```

<----->

Sentence N§ 37 --> frase_Brown
NUEVA_SERIE.

```

** PARTIALLY PARSED SENTENCE:
conc(37,0,2)

```

```

** NOUN PHRASE:
conc(sing,_172332,terc,propio,_161854,_161856,_161858,37,0,1)
conc(sing,_172332,terc,propio,_161854,_161856,_161858,37,0,1)
conc(sing,_172332,terc,propio,_161854,_161856,_161858,37,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_172332,terc,propio,_161854,_161856,_161858,37,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_172332,propio,37,0,1)
** NOUN:
conc(sing,_172332,propio,37,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.
<----->
Sentence N§ 38 --> frase_Brown
What QTLs are related to awn type in barley ?

** PARTIALLY PARSED SENTENCE:
conc(38,0,10)
** RELATIVE SENTENCE:
conc(rel,38,0,9)
** PRONOUN:
conc(_186648,_186650,_186652,rel,_186656,_186658,_186660,38,0,1)
What
** NOUN PHRASE:
conc(sing,_194516,terc,propio,_188076,_188078,_188080,38,1,2)
conc(sing,_194516,terc,propio,_188076,_188078,_188080,38,1,2)
conc(sing,_194516,terc,propio,_188076,_188078,_188080,38,1,2)
** SIMPLE NOUN PHRASE:
conc(sing,_194516,terc,propio,_188076,_188078,_188080,38,1,2)
** HEAD OF THE NOUN PHRASE:
conc(sing,_194516,propio,38,1,2)
** NOUN:
conc(sing,_194516,propio,38,1,2)
QTLs
** HEAD OF THE VERBAL PHRASE:
conc(_186848,_186850,pasiva,compBe,38,2,4)
conc(_186848,_186850,pasiva,compBe,38,2,4)
** COMPOUND VERB BE:
conc(_186848,_186850,pasiva,compBe,38,2,4)
** VERB:
conc(_186848,_186850,presente,be,38,2,3)
are
** VERB:
conc(_186946,_186948,participio,resto,38,3,4)
related
** PREPOSITIONAL PHRASE:
conc(_187044,of,38,4,9)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_187044,of,38,4,9)
** PREPOSITION:
conc(_187044,of,38,4,5)
conc(_187044,of,38,4,5)
** SIMPLE PREPOSITION:
conc(_187044,to,38,4,5)
to
** NOUN PHRASE:
conc(sing,_210276,terc,comun,_203696,_203698,_203700,38,5,9)
conc(sing,_210276,terc,comun,_203696,_203698,_203700,38,5,9)
conc(sing,_210276,terc,comun,_203696,_203698,_203700,38,5,9)

```



```

** SIMPLE NOUN PHRASE:
conc(sing,_210276,terc,comun,_203696,_203698,_203700,38,5,9)
** HEAD OF THE NOUN PHRASE:
conc(sing,_210276,comun,38,5,7)
** NOUN:
conc(sing,_187140,comun,38,5,6)
awn
** NOUN:
conc(sing,_210276,comun,38,6,7)
type
** PREPOSITIONAL PHRASE:
conc(_187330,of,38,7,9)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_187330,of,38,7,9)
** PREPOSITION:
conc(_187330,of,38,7,8)
conc(_187330,of,38,7,8)
** SIMPLE PREPOSITION:
conc(_187330,resto,38,7,8)
in
** NOUN PHRASE:
conc(sing,_223434,terc,comun,_216854,_216856,_216858,38,8,9)
conc(sing,_223434,terc,comun,_216854,_216856,_216858,38,8,9)
conc(sing,_223434,terc,comun,_216854,_216856,_216858,38,8,9)
** SIMPLE NOUN PHRASE:
conc(sing,_223434,terc,comun,_216854,_216856,_216858,38,8,9)
** HEAD OF THE NOUN PHRASE:
conc(sing,_223434,comun,38,8,9)
** NOUN:
conc(sing,_223434,comun,38,8,9)
barley

** NON-PARSED WORD:
?
<----->
Sentence N§ 39 --> frase_Brown
And to lemma color ?

** PARTIALLY PARSED SENTENCE:
conc(39,0,5)
** CONJUNCTION:
conc(39,0,1)
And
** PREPOSITIONAL PHRASE:
conc(_325790,of,39,1,4)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_325790,of,39,1,4)
** PREPOSITION:
conc(_325790,of,39,1,2)
conc(_325790,of,39,1,2)
** SIMPLE PREPOSITION:
conc(_325790,to,39,1,2)
to
** NOUN PHRASE:
conc(sing,_336848,terc,comun,_330268,_330270,_330272,39,2,4)
conc(sing,_336848,terc,comun,_330268,_330270,_330272,39,2,4)
conc(sing,_336848,terc,comun,_330268,_330270,_330272,39,2,4)
** SIMPLE NOUN PHRASE:
conc(sing,_336848,terc,comun,_330268,_330270,_330272,39,2,4)
** HEAD OF THE NOUN PHRASE:
conc(sing,_336848,comun,39,2,4)
** NOUN:
conc(sing,_325886,comun,39,2,3)

```

```

lemma
** NOUN:
  conc(sing,_336848,comun,39,3,4)
  color
** NON-PARSED WORD:
?
<----->
Sentence N§ 40 --> frase_Brown
To aleurone one ?

```

```

** PARTIALLY PARSED SENTENCE:
conc(40,0,4)
** PREPOSITIONAL PHRASE:
conc(_410200,of,40,0,3)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_410200,of,40,0,3)
** PREPOSITION:
conc(_410200,of,40,0,1)
conc(_410200,of,40,0,1)
** SIMPLE PREPOSITION:
conc(_410200,to,40,0,1)
To
** NOUN PHRASE:
conc(sing,_421606,terc,comun,_415026,_415028,_415030,40,1,3)
conc(sing,_421606,terc,comun,_415026,_415028,_415030,40,1,3)
conc(sing,_421606,terc,comun,_415026,_415028,_415030,40,1,3)
** SIMPLE NOUN PHRASE:
conc(sing,_421606,terc,comun,_415026,_415028,_415030,40,1,3)
** HEAD OF THE NOUN PHRASE:
conc(sing,_421606,comun,40,1,3)
** NOUN:
conc(sing,_410296,comun,40,1,2)
aleurone
** NOUN:
conc(sing,_421606,comun,40,2,3)
one
** NON-PARSED WORD:
?
<----->
Sentence N§ 41 --> frase_Brown
NUEVA_SERIE.

```

```

** PARTIALLY PARSED SENTENCE:
conc(41,0,2)
** NOUN PHRASE:
conc(sing,_505152,terc,propio,_494674,_494676,_494678,41,0,1)
conc(sing,_505152,terc,propio,_494674,_494676,_494678,41,0,1)
conc(sing,_505152,terc,propio,_494674,_494676,_494678,41,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_505152,terc,propio,_494674,_494676,_494678,41,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_505152,propio,41,0,1)
** NOUN:
conc(sing,_505152,propio,41,0,1)
NUEVA_SERIE
** NON-PARSED WORD:
.
<----->
Sentence N§ 42 --> frase_Brown
NUEVA_SERIE.

```

```

** PARTIALLY PARSED SENTENCE:
conc(42,0,2)
** NOUN PHRASE:
conc(sing,_528892,terc,propio,_518414,_518416,_518418,42,0,1)
conc(sing,_528892,terc,propio,_518414,_518416,_518418,42,0,1)
conc(sing,_528892,terc,propio,_518414,_518416,_518418,42,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_528892,terc,propio,_518414,_518416,_518418,42,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_528892,propio,42,0,1)
** NOUN:
conc(sing,_528892,propio,42,0,1)
NUEVA_SERIE
** NON-PARSED WORD:

```

```

<----->
Sentence N§ 43 --> frase_Brown
NUEVA_SERIE.

```

```

** PARTIALLY PARSED SENTENCE:
conc(43,0,2)
** NOUN PHRASE:
conc(sing,_552632,terc,propio,_542154,_542156,_542158,43,0,1)
conc(sing,_552632,terc,propio,_542154,_542156,_542158,43,0,1)
conc(sing,_552632,terc,propio,_542154,_542156,_542158,43,0,1)
** SIMPLE NOUN PHRASE:
conc(sing,_552632,terc,propio,_542154,_542156,_542158,43,0,1)
** HEAD OF THE NOUN PHRASE:
conc(sing,_552632,propio,43,0,1)
** NOUN:
conc(sing,_552632,propio,43,0,1)
NUEVA_SERIE
** NON-PARSED WORD:

```

```

<----->
Sentence N§ 44 --> frase_Brown
Does any QTL affect to protein content in barley ?

```

```

** PARTIALLY PARSED SENTENCE:
conc(44,0,10)
** HEAD OF THE VERBAL PHRASE:
conc(sing,terc,presente,_568160,44,0,1)
conc(sing,terc,presente,_568160,44,0,1)
conc(sing,terc,presente,verboSolo,44,0,1)
** VERB:
conc(sing,terc,presente,do,44,0,1)
Does
** NOUN PHRASE:
conc(sing,_579324,terc,propio,_568216,_568218,_568220,44,1,3)
conc(sing,_579324,terc,propio,_568216,_568218,_568220,44,1,3)
conc(sing,_579324,terc,propio,_568216,_568218,_568220,44,1,3)
** SIMPLE NOUN PHRASE:
conc(sing,_579324,terc,propio,_568216,_568218,_568220,44,1,3)
** DETERMINER 1:
conc(sing,_579324,det,44,1,2)
** DETERMINER:
conc(sing,_579324,det,44,1,2)
any
** HEAD OF THE NOUN PHRASE:
conc(sing,_579324,propio,44,2,3)
** NOUN:

```

```

conc(sing,_579324,propio,44,2,3)
QTL
** HEAD OF THE VERBAL PHRASE:
conc(_567238,_567240,infinitivo,_584558,44,3,4)
conc(_567238,_567240,infinitivo,_584558,44,3,4)
conc(_567238,_567240,infinitivo,verboSolo,44,3,4)
** VERB:
conc(_567238,_567240,infinitivo,resto,44,3,4)
affect
** PREPOSITIONAL PHRASE:
conc(_567336,of,44,4,9)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_567336,of,44,4,9)
** PREPOSITION:
conc(_567336,of,44,4,5)
conc(_567336,of,44,4,5)
** SIMPLE PREPOSITION:
conc(_567336,to,44,4,5)
to
** NOUN PHRASE:
conc(sing,_595552,terc,comun,_588972,_588974,_588976,44,5,9)
conc(sing,_595552,terc,comun,_588972,_588974,_588976,44,5,9)
conc(sing,_595552,terc,comun,_588972,_588974,_588976,44,5,9)
** SIMPLE NOUN PHRASE:
conc(sing,_595552,terc,comun,_588972,_588974,_588976,44,5,9)
** HEAD OF THE NOUN PHRASE:
conc(sing,_595552,comun,44,5,7)
** NOUN:
conc(sing,_567432,comun,44,5,6)
protein
** NOUN:
conc(sing,_595552,comun,44,6,7)
content
** PREPOSITIONAL PHRASE:
conc(_567622,of,44,7,9)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_567622,of,44,7,9)
** PREPOSITION:
conc(_567622,of,44,7,8)
conc(_567622,of,44,7,8)
** SIMPLE PREPOSITION:
conc(_567622,resto,44,7,8)
in
** NOUN PHRASE:
conc(sing,_608710,terc,comun,_602130,_602132,_602134,44,8,9)
conc(sing,_608710,terc,comun,_602130,_602132,_602134,44,8,9)
conc(sing,_608710,terc,comun,_602130,_602132,_602134,44,8,9)
** SIMPLE NOUN PHRASE:
conc(sing,_608710,terc,comun,_602130,_602132,_602134,44,8,9)
** HEAD OF THE NOUN PHRASE:
conc(sing,_608710,comun,44,8,9)
** NOUN:
conc(sing,_608710,comun,44,8,9)
barley
** NON-PARSED WORD:
?
<----->
Sentence N§ 45 --> frase_Brown
In which chromosome are they located ?

```

** PARTIALLY PARSED SENTENCE:

conc(45,0,7)

** NON-PARSED WORD:

In

** RELATIVE SENTENCE:

conc(rel,45,1,5)

** PRONOUN:

conc(_693056,_693058,_693060,rel,_693064,_693066,_693068,45,1,2)
which

** NOUN PHRASE:

conc(sing,_737310,terc,comun,_730870,_730872,_730874,45,2,3)

conc(sing,_737310,terc,comun,_730870,_730872,_730874,45,2,3)

conc(sing,_737310,terc,comun,_730870,_730872,_730874,45,2,3)

** SIMPLE NOUN PHRASE:

conc(sing,_737310,terc,comun,_730870,_730872,_730874,45,2,3)

** HEAD OF THE NOUN PHRASE:

conc(sing,_737310,comun,45,2,3)

** NOUN:

conc(sing,_737310,comun,45,2,3)

chromosome

** HEAD OF THE VERBAL PHRASE:

conc(_693256,_693258,presente,_730892,45,3,4)

conc(_693256,_693258,presente,_730892,45,3,4)

conc(_693256,_693258,presente,verboSolo,45,3,4)

** VERB:

conc(_693256,_693258,presente,be,45,3,4)

are

** NOUN PHRASE:

conc(pl,_693356,terc,pronEnglish,suj,_730936,_730938,45,4,5)

conc(pl,_693356,terc,pronEnglish,suj,_730936,_730938,45,4,5)

conc(pl,_693356,terc,pronEnglish,suj,_730936,_730938,45,4,5)

** SIMPLE NOUN PHRASE:

conc(pl,_693356,terc,pronEnglish,suj,_730936,_730938,45,4,5)

** PRONOUN:

conc(pl,_693356,terc,persNoRefl,suj,_693364,_693366,45,4,5)

they

** HEAD OF THE VERBAL PHRASE:

conc(_693458,_693460,participio,_730662,45,5,6)

conc(_693458,_693460,participio,_730662,45,5,6)

conc(_693458,_693460,participio,verboSolo,45,5,6)

** VERB:

conc(_693458,_693460,participio,resto,45,5,6)

located

** NON-PARSED WORD:

?

<----->

Sentence N§ 46 --> frase_Brown

NUEVA_SERIE.

** PARTIALLY PARSED SENTENCE:

conc(46,0,2)

** NOUN PHRASE:

conc(sing,_847078,terc,propio,_836600,_836602,_836604,46,0,1)

conc(sing,_847078,terc,propio,_836600,_836602,_836604,46,0,1)

conc(sing,_847078,terc,propio,_836600,_836602,_836604,46,0,1)

** SIMPLE NOUN PHRASE:

conc(sing,_847078,terc,propio,_836600,_836602,_836604,46,0,1)

** HEAD OF THE NOUN PHRASE:

conc(sing,_847078,propio,46,0,1)

** NOUN:

conc(sing,_847078,propio,46,0,1)

NUEVA_SERIE

**** NON-PARSED WORD:**
 .
 <----->
 Sentence N§ 47 --> frase_Brown
 NUEVA_SERIE.

**** PARTIALLY PARSED SENTENCE:**
 conc(47,0,2)
**** NOUN PHRASE:**
 conc(sing,_870818,terc,propio,_860340,_860342,_860344,47,0,1)
 conc(sing,_870818,terc,propio,_860340,_860342,_860344,47,0,1)
 conc(sing,_870818,terc,propio,_860340,_860342,_860344,47,0,1)
**** SIMPLE NOUN PHRASE:**
 conc(sing,_870818,terc,propio,_860340,_860342,_860344,47,0,1)
**** HEAD OF THE NOUN PHRASE:**
 conc(sing,_870818,propio,47,0,1)
**** NOUN:**
 conc(sing,_870818,propio,47,0,1)
 NUEVA_SERIE

**** NON-PARSED WORD:**
 .

<----->
 Sentence N§ 48 --> frase_Brown
 NUEVA_SERIE.

**** PARTIALLY PARSED SENTENCE:**
 conc(48,0,2)
**** NOUN PHRASE:**
 conc(sing,_894558,terc,propio,_884080,_884082,_884084,48,0,1)
 conc(sing,_894558,terc,propio,_884080,_884082,_884084,48,0,1)
 conc(sing,_894558,terc,propio,_884080,_884082,_884084,48,0,1)
**** SIMPLE NOUN PHRASE:**
 conc(sing,_894558,terc,propio,_884080,_884082,_884084,48,0,1)
**** HEAD OF THE NOUN PHRASE:**
 conc(sing,_894558,propio,48,0,1)
**** NOUN:**
 conc(sing,_894558,propio,48,0,1)
 NUEVA_SERIE

**** NON-PARSED WORD:**
 .

<----->
 Sentence N§ 49 --> frase_Brown
 I want to increase resistance to common bunt in wheat.

**** PARTIALLY PARSED SENTENCE:**
 conc(49,0,11)
**** NOUN PHRASE:**
 conc(sing,_909050,prim,pronEnglish,suj,_910420,_910422,49,0,1)
 conc(sing,_909050,prim,pronEnglish,suj,_910420,_910422,49,0,1)
 conc(sing,_909050,prim,pronEnglish,suj,_910420,_910422,49,0,1)
**** SIMPLE NOUN PHRASE:**
 conc(sing,_909050,prim,pronEnglish,suj,_910420,_910422,49,0,1)
**** PRONOUN:**
 conc(sing,_909050,prim,persNoRefl,suj,_909058,_909060,49,0,1)
 I

**** HEAD OF THE VERBAL PHRASE:**
 conc(_909152,_909154,presente,_930770,49,1,4)
 conc(_909152,_909154,presente,_930770,49,1,2)
 conc(_909152,_909154,presente,verboSolo,49,1,2)
**** VERB:**

```

conc(_909152,_909154,presente,resto,49,1,2)
want
** VERBAL PERIPHRAIS:
conc(49,2,4)
** NON-PARSED WORD:
to
** VERB:
conc(_909344,_909346,infinitivo,resto,49,3,4)
increase
** NOUN PHRASE:
conc(sing,_941734,terc,comun,_930826,_930828,_930830,49,4,10)
conc(sing,_941734,terc,comun,_930826,_930828,_930830,49,4,10)
conc(sing,_941734,terc,comun,_930826,_930828,_930830,49,4,10)
** SIMPLE NOUN PHRASE:
conc(sing,_941734,terc,comun,_930826,_930828,_930830,49,4,10)
** HEAD OF THE NOUN PHRASE:
conc(sing,_941734,comun,49,4,5)
** NOUN:
conc(sing,_941734,comun,49,4,5)
resistance
** PREPOSITIONAL PHRASE:
conc(_909538,of,49,5,10)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_909538,of,49,5,10)
** PREPOSITION:
conc(_909538,of,49,5,6)
conc(_909538,of,49,5,6)
** SIMPLE PREPOSITION:
conc(_909538,to,49,5,6)
to
** NOUN PHRASE:
conc(sing,_955138,terc,comun,_948478,_948480,_948482,49,6,10)
conc(sing,_955138,terc,comun,_948478,_948480,_948482,49,6,10)
conc(sing,_955138,terc,comun,_948478,_948480,_948482,49,6,10)
** SIMPLE NOUN PHRASE:
conc(sing,_955138,terc,comun,_948478,_948480,_948482,49,6,10)
** PRE-MODIFIER:
conc(sing,_955138,adj,49,6,7)
** ADJECTIVAL PHRASE:
conc(sing,_955138,49,6,7)
** COORDINATED ADJECTIVE:
conc(sing,_955138,_957920,49,6,7)
** SIMPLE ADJECTIVE:
conc(sing,_955138,cal,49,6,7)
common
** HEAD OF THE NOUN PHRASE:
conc(sing,_955138,comun,49,7,8)
** NOUN:
conc(sing,_955138,comun,49,7,8)
bunt
** PREPOSITIONAL PHRASE:
conc(_909824,of,49,8,10)
** SIMPLE PREPOSITIONAL PHRASE:
conc(_909824,of,49,8,10)
** PREPOSITION:
conc(_909824,of,49,8,9)
conc(_909824,of,49,8,9)
** SIMPLE PREPOSITION:
conc(_909824,resto,49,8,9)
in
** NOUN PHRASE:
conc(sing,_968412,terc,comun,_961832,_961834,_961836,49,9,10)

```

```

conc(sing,_968412,terc,comun,_961832,_961834,_961836,49,9,10)
conc(sing,_968412,terc,comun,_961832,_961834,_961836,49,9,10)
** SIMPLE NOUN PHRASE:
conc(sing,_968412,terc,comun,_961832,_961834,_961836,49,9,10)
** HEAD OF THE NOUN PHRASE:
conc(sing,_968412,comun,49,9,10)
** NOUN:
conc(sing,_968412,comun,49,9,10)
wheat
** NON-PARSED WORD:
.
```

<----->

Sentence N§ 50 --> frase_Brown
Which QTLs are related to this ?

```

** PARTIALLY PARSED SENTENCE:
conc(50,0,7)
** RELATIVE SENTENCE:
conc(rel,50,0,4)
** PRONOUN:
conc(_997334,_997336,_997338,rel,_997342,_997344,_997346,50,0,1)
Which
** NOUN PHRASE:
conc(pl,_1004932,terc,comun,_998492,_998494,_998496,50,1,2)
conc(pl,_1004932,terc,comun,_998492,_998494,_998496,50,1,2)
conc(pl,_1004932,terc,comun,_998492,_998494,_998496,50,1,2)
** SIMPLE NOUN PHRASE:
conc(pl,_1004932,terc,comun,_998492,_998494,_998496,50,1,2)
** HEAD OF THE NOUN PHRASE:
conc(pl,_1004932,comun,50,1,2)
** NOUN:
conc(pl,_1004932,comun,50,1,2)
QTLs
** HEAD OF THE VERBAL PHRASE:
conc(_997534,_997536,pasiva,compBe,50,2,4)
conc(_997534,_997536,pasiva,compBe,50,2,4)
** COMPOUND VERB BE:
conc(_997534,_997536,pasiva,compBe,50,2,4)
** VERB:
conc(_997534,_997536,presente,be,50,2,3)
are
** VERB:
conc(_997632,_997634,participio,resto,50,3,4)
related
** NON-PARSED WORD:
to
** NON-PARSED WORD:
this
** NON-PARSED WORD:
?
```

<----->

Sentence N§ 51 --> frase_Brown
What other traits are affected by them ?

```

** PARTIALLY PARSED SENTENCE:
conc(51,0,8)
** RELATIVE SENTENCE:
conc(rel,51,0,7)
** PRONOUN:
conc(_1200370,_1200372,_1200374,rel,_1200378,_1200380,_1200382,51,0,1)
```


What

** NOUN PHRASE:

conc(pl,_1208126,terc,comun,_1201616,_1201618,_1201620,51,1,3)

conc(pl,_1208126,terc,comun,_1201616,_1201618,_1201620,51,1,3)

conc(pl,_1208126,terc,comun,_1201616,_1201618,_1201620,51,1,3)

** SIMPLE NOUN PHRASE:

conc(pl,_1208126,terc,comun,_1201616,_1201618,_1201620,51,1,3)

** PRE-MODIFIER:

conc(pl,_1208126,adj,51,1,2)

** ADJECTIVAL PHRASE:

conc(pl,_1208126,51,1,2)

** COORDINATED ADJECTIVE:

conc(pl,_1208126,_1210908,51,1,2)

** SIMPLE ADJECTIVE:

conc(pl,_1208126,cal,51,1,2)

other

** HEAD OF THE NOUN PHRASE:

conc(pl,_1208126,comun,51,2,3)

** NOUN:

conc(pl,_1208126,comun,51,2,3)

traits

** HEAD OF THE VERBAL PHRASE:

conc(_1200666,_1200668,pasiva,compBe,51,3,5)

conc(_1200666,_1200668,pasiva,compBe,51,3,5)

** COMPOUND VERB BE:

conc(_1200666,_1200668,pasiva,compBe,51,3,5)

** VERB:

conc(_1200666,_1200668,presente,be,51,3,4)

are

** VERB:

conc(_1200764,_1200766,participio,resto,51,4,5)

affected

** PREPOSITIONAL PHRASE:

conc(_1200862,resto,51,5,7)

** SIMPLE PREPOSITIONAL PHRASE:

conc(_1200862,resto,51,5,7)

** PREPOSITION:

conc(_1200862,resto,51,5,6)

conc(_1200862,resto,51,5,6)

** SIMPLE PREPOSITION:

conc(_1200862,resto,51,5,6)

by

** NOUN PHRASE:

conc(pl,_1200958,terc,pronEnglish,compl,_1217528,_1217530,51,6,7)

conc(pl,_1200958,terc,pronEnglish,compl,_1217528,_1217530,51,6,7)

conc(pl,_1200958,terc,pronEnglish,compl,_1217528,_1217530,51,6,7)

** SIMPLE NOUN PHRASE:

conc(pl,_1200958,terc,pronEnglish,compl,_1217528,_1217530,51,6,7)

** PRONOUN:

conc(pl,_1200958,terc,persNoRefl,compl,_1200966,_1200968,51,6,7)

them

** NON-PARSED WORD:

?

6.2.3 Salida del sistema de resolución de la anáfora

oracTag(conc(1,0,11),_903066,nucleoVerbal(conc(_822434,_822436,infinitivo,_823750,1,0,1),_926388,nucleoVerbalAux(conc(_822434,_822436,infinitivo,_823750,1,0,1),_926414,verboSolo(conc(_822434,_822436,infinitivo,verboSolo,1,0,1),_926434,verbo(conc(_822434,_822436,infinitivo,resto,1,0,1),_827286,(1,0,1,'Tell',tell,_822386))))),_926372,_926374),sn(conc(sing,_914972,prim.pronEnglish,compl,_913890,_913892,1,1,2),_909980,snCoordinado(conc(sing,_914972,prim.pronEnglish,compl,_913890,_913892,1,1,2),_911282,snSimplePodado(conc(sing,_914972,prim.pronEnglish,compl,_913890,_913892,1,1,2),_914996,pronEnglish(conc(sing,_914972,prim.persNoRefl,compl,_914980,_914982,1,1,2),_844194,(1,1,2,me,me,_822484))))),_828614,_828616)),sn(conc(sing,_926672,terc.propio,compl,_926680,_926682,1,2,10),_915962,snCoordinado(conc(sing,_926672,terc.propio,compl,_926680,_926682,1,2,10),_849018,snSimple(conc(sing,_926672,terc.propio,compl,_926680,_926682,1,2,10),_917078,snSimplePodado(conc(sing,_926672,terc.propio,compl,_926680,_926682,1,2,10),_927600,det(conc(sing,_926672,det,1,2,3),_855182,art(conc(sing,_926672,det,1,2,3),_855218,(1,2,3,the,the,_822588))))),_927554,nucleoSN(conc(sing,_926672,propio,1,3,4),_927626,[sust(conc(sing,_926672,propio,1,3,4),_856544,(1,3,4,'QTLs',qtls,_822684))]),_927558,postModificadorSN(conc(sing,_926672,_855140,1,4,10),_857076,oracRelat(conc(participio,1,4,10),_857362,_857364,nucleoVerbal(conc(_822828,_822830,participio,verboSolo,1,4,5),_951378,nucleoVerbalAux(conc(_822828,_822830,participio,verboSolo,1,4,5),_951404,verboSolo(conc(_822828,_822830,participio,verboSolo,1,4,5),_951424,verbo(conc(_822828,_822830,participio,resto,1,4,5),_857926,(1,4,5,related,relate,_822780))))),_951362,_951364),sp(conc(_822926,of,1,5,10),_858568,spSimple(conc(_822926,of,1,5,10),_931634,prep(conc(_822926,of,1,5,6),_858714,prepSimpleG(conc(_822926,of,1,5,6),_931662,prepSimple(conc(_822926,to,1,5,6),_858836,(1,5,6,to,to,_822878))))),_858718,_858720),sn(conc(sing,_939758,terc.comun,_939764,_939766,_939768,1,6,10),_931904,snCoordinado(conc(sing,_939758,terc.comun,_939764,_939766,_939768,1,6,10),_859156,snSimple(conc(sing,_939758,terc.comun,_939764,_939766,_939768,1,6,10),_933052,snSimplePodado(conc(sing,_939758,terc.comun,_939764,_939766,_939768,1,6,10),_940686,_940638,_940640,nucleoSN(conc(sing,_939758,comun,1,6,8),_940712,[sust(conc(sing,_823022,comun,1,6,7),_868662,(1,6,7,awn,awn,_822972)),sust(conc(sing,_939758,comun,1,7,8),_868844,(1,7,8,color,colour,_823068))]),_940644,_940646,sp(conc(_823212,of,1,8,10),_871726,spSimple(conc(_823212,of,1,8,10),_943476,prep(conc(_823212,of,1,8,9),_871872,prepSimpleG(conc(_823212,of,1,8,9),_943504,prepSimple(conc(_823212,resto,1,8,9),_871994,(1,8,9,in,in,_823164))))),_871876,_871878),sn(conc(sing,_951662,terc.comun,_951668,_951670,_951672,1,9,10),_943770,snCoordinado(conc(sing,_951662,terc.comun,_951668,_951670,_951672,1,9,10),_872314,snSimple(conc(sing,_951662,terc.comun,_951668,_951670,_951672,1,9,10),_944942,snSimplePodado(conc(sing,_951662,terc.comun,_951668,_951670,_951672,1,9,10),_952590,_952542,_952544,nucleoSN(conc(sing,_951662,comun,1,9,10),_952616,[sust(conc(sing,_951662,comun,1,9,10),_881820,(1,9,10,wheat,wheat,_823258))]),_952548,_952550,_952552)),_872318,_872320)),_871730,_871732)),_859160,_859162)),_858572,_858574),_857370,_857372)),_927562)),_849022,_849024),w((1,10,11,'!','!',_908800))).

oracTag(conc(2,0,5),_1032808,conj(conc(2,0,1),_961974,(2,0,1,'And',and,_957528)),sp(conc(_957664,of,2,1,4),_962038,spSimple(conc(_957664,of,2,1,4),_1036670,prep(conc(_957664,of,2,1,2),_962184,prepSimpleG(conc(_957664,of,2,1,2),_1036698,prepSimple(conc(_957664,to,2,1,2),_962306,(2,1,2,to,to,_957616))))),_962188,_962190),sn(conc(sing,_103922,terc.comun,_1039228,_1039218,_1039232,2,2,4),_1036916,snCoordinado(conc(sing,_1039222,terc.comun,_1039228,_1039218,_1039232,2,2,4),_962626,snSimple(conc(sing,_1039222,terc.comun,_1039228,_1039218,_1039232,2,2,4),_1038040,snSimplePodado(conc(sing,_1039222,terc.comun,_1039228,_1039218,_1039232,2,2,4),_1040188,_1040136,_1040138,nucleoSN(conc(sing,_1039222,comun,2,2,4),_1040214,[sust(conc(sing,_957760,comun,2,2,3),_972132,(2,2,3,glume,glume,_957710)),sust(conc(sing,_1039222,comun,2,3,4),_972314,(2,3,4,color,colour,_957806))]),_1040142,_1040144,_1040146)),_962630,_962632)),_962042,_962044),w((2,4,5,?,?,_1034736))).

oracTag(conc(3,0,4),_1121932,sp(conc(_1046274,of,3,0,3),_1051008,spSimple(conc(_1046274,of,3,0,3),_1125362,prep(conc(_1046274,of,3,0,1),_1051154,prepSimpleG(conc(_1046274,of,3,0,1),_1125390,prepSimple(conc(_1046274,to,3,0,1),_1051276,(3,0,1,'To',to,_1046226))))),_1051158,_1051160),sn(conc(sing,_1127958,terc.comun,_1127964,cantidad,_1127968,3,1,3),_1125608,snCoordinado(conc(sing,_1127958,terc.comun,_1127964,cantidad,_1127968,3,1,3),_1051596,snSimple(conc(sing,_1127958,terc.comun,_1127964,cantidad,_1127968,3,1,3),_1126762,snSimplePodado(conc(sing,_1127958,terc.comun,_1127964,cantidad,_1127968,3,1,3),_1129018,_1128966,_1128968,nucleoSN(conc(sing,_1127958,comun,3,1,3),_1129044,[sust(conc(sing,_1046370,comun,3,1,2),_1061256,(3,1,2,semolina,semolina,_1046320)),sust(conc(sing,_1127958,comun,3,2,3),_1061438,(3,2,3,one,one,_1046416))]),_1128972,_1128974,_1128976)),_1051600,_1051602)),_1051012,_1051014),w((3,3,4,?,?,_1123944))).

oracTag(conc(4,0,2),_1154656,sn(conc(sing,_1159922,terc.propio,suj,_1159918,_1159932,4,0,1),_1156584,snCoordinado(conc(sing,_1159922,terc.propio,suj,_1159918,_1159932,4,0,1),_1141346,snSimple(conc(sing,_1159922,terc.propio,suj,_1159918,_1159932,4,0,1),_1157808,snSimplePodado(conc(sing,_1159922,terc.propio,suj,_1159918,_1159932,4,0,1),_1160888,_1160836,_1160838,nucleoSN(conc(sing,_1159922,propio,4,0,1),_1160914,[sust(conc(sing,_1159922,propio,4,0,

1),_1150856,(4,0,1,'NUEVA_SERIE',nueva_serie,_1136416))),_1160842,_1160844,_1160846)),_1141350,_1141352)),
w((4,1,2,'','_1155588))).

oracTag(conc(5,0,2),_1187386,sn(conc(sing,_1192544,terc,propio,suj,_1192540,_1192554,5,0,1),_1189314,snCoordinad
o(conc(sing,_1192544,terc,propio,suj,_1192540,_1192554,5,0,1),_1174126,snSimple(conc(sing,_1192544,terc,propio,suj,
_1192540,_1192554,5,0,1),_1190430,snSimplePodado(conc(sing,_1192544,terc,propio,suj,_1192540,_1192554,5,0,1),_1
193510,_1193458,_1193460,nucleoSN(conc(sing,_1192544,propio,5,0,1),_1193536,[sust(conc(sing,_1192544,propio,5,0,
1),_1183632,(5,0,1,'NUEVA_SERIE',nueva_serie,_1169196))),_1193464,_1193466,_1193468)),_1174130,_1174132))
),w((5,1,2,'','_1188318))).

oracTag(conc(6,0,2),_1218990,sn(conc(sing,_1224206,terc,propio,suj,_1224202,_1224216,6,0,1),_1220964,snCoordinad
o(conc(sing,_1224206,terc,propio,suj,_1224202,_1224216,6,0,1),_1205392,snSimple(conc(sing,_1224206,terc,propio,suj,
_1224202,_1224216,6,0,1),_1222080,snSimplePodado(conc(sing,_1224206,terc,propio,suj,_1224202,_1224216,6,0,1),_1
225172,_1225120,_1225122,nucleoSN(conc(sing,_1224206,propio,6,0,1),_1225198,[sust(conc(sing,_1224206,propio,6,0,
1),_1215226,(6,0,1,'NUEVA_SERIE',nueva_serie,_1200460))),_1225126,_1225128,_1225130)),_1205396,_1205398))
),w((6,1,2,'','_1219952))).

oracTag(conc(7,0,11),_1357936,nucleoVerbal(conc(sing,terc,presente,_1233708,7,0,1),_1375904,nucleoVerbalAux(conc(
sing,terc,presente,_1233708,7,0,1),_1375930,verboSolo(conc(sing,terc,presente,verboSolo,7,0,1),_1375950,verbo(conc(si
ng,terc,presente,do,7,0,1),_1237586,(7,0,1,'Does',do,_1232306))),_1375888,_1375890),sn(conc(sing,_1376188,terc,pro
pio,compl,_1376196,_1376198,7,1,3),_1365362,snCoordinado(conc(sing,_1376188,terc,propio,compl,_1376196,_137619
8,7,1,3),_1238910,snSimple(conc(sing,_1376188,terc,propio,compl,_1376196,_1376198,7,1,3),_1366478,snSimplePodad
o(conc(sing,_1376188,terc,propio,compl,_1376196,_1376198,7,1,3),_1377116,det(conc(sing,_1376188,det,7,1,2),_12452
82,art(conc(sing,_1376188,det,7,1,2),_1245318,(7,1,2,any,any,_1232404))),_1377070,nucleoSN(conc(sing,_1376188,pro
pio,7,2,3),_1377142,[sust(conc(sing,_1376188,propio,7,2,3),_1246660,(7,2,3,'QTL',qtl,_1232500))),_1377074,_137707
6,_1377078)),_1238914,_1238916)),nucleoVerbal(conc(_1232644,_1232646,infinitivo,_1250500,7,3,4),avisoCambioCla
usula,nucleoVerbalAux(conc(_1232644,_1232646,infinitivo,_1250500,7,3,4),_1388724,verboSolo(conc(_1232644,_1232
646,infinitivo,verboSolo,7,3,4),_1388744,verbo(conc(_1232644,_1232646,infinitivo,resto,7,3,4),_1254084,(7,3,4,affect,af
fect,_1232596))),_1388682,_1388684),sp(conc(_1232742,of,7,4,10),_1254784,spSimple(conc(_1232742,of,7,4,10),_138
0858,prep(conc(_1232742,of,7,4,5),_1254938,prepSimpleG(conc(_1232742,of,7,4,5),_1380886,prepSimple(conc(_12327
42,resto,7,4,5),_1255060,(7,4,5,on,on,_1232694))),_1254942,_1254944),sn(conc(pl,_1388982,terc,comun,_1388988,ti
mpo,_1388992,7,5,10),_1381104,snCoordinado(conc(pl,_1388982,terc,comun,_1388988,tiempo,_1388992,7,5,10),_1255
380,snSimple(conc(pl,_1388982,terc,comun,_1388988,tiempo,_1388992,7,5,10),_1382228,snSimplePodado(conc(pl,_13
88982,terc,comun,_1388988,tiempo,_1388992,7,5,10),_1389910,_1389862,_1389864,nucleoSN(conc(pl,_1388982,comu
n,7,5,6),_1389936,[sust(conc(pl,_1388982,comun,7,5,6),_1264890,(7,5,6,days,day,_1232788))),_1389868,postModifica
dorSN(conc(pl,_1388982,_1261576,7,6,10),_1265462,oracRelat(conc(participio,7,6,10),_1265754,prepSimple(conc(_12
32932,to,7,6,7),_1265902,(7,6,7,to,to,_1232884))),nucleoVerbal(conc(_1233026,_1233028,gerundio,verboSolo,7,7,8),_14
00960,nucleoVerbalAux(conc(_1233026,_1233028,gerundio,verboSolo,7,7,8),_1400986,verboSolo(conc(_1233026,_1233
028,gerundio,verboSolo,7,7,8),_1401006,verbo(conc(_1233026,_1233028,gerundio,resto,7,7,8),_1266598,(7,7,8,flowering
,flowering,_1232978))),_1400944,_1400946),sp(conc(_1233124,of,7,8,10),_1267162,spSimple(conc(_1233124,of,7,8,10)
,_1393086,prep(conc(_1233124,of,7,8,9),_1267308,prepSimpleG(conc(_1233124,of,7,8,9),_1393114,prepSimple(conc(_
1233124,resto,7,8,9),_1267430,(7,8,9,in,in,_1233076))),_1267312,_1267314),sn(conc(sing,_1401244,terc,comun,_14012
50,_1401252,_1401254,7,9,10),_1393364,snCoordinado(conc(sing,_1401244,terc,comun,_1401250,_1401252,_1401254,
7,9,10),_1267750,snSimple(conc(sing,_1401244,terc,comun,_1401250,_1401252,_1401254,7,9,10),_1394520,snSimpleP
odado(conc(sing,_1401244,terc,comun,_1401250,_1401252,_1401254,7,9,10),_1402172,_1402124,_1402126,nucleoSN(c
onc(sing,_1401244,comun,7,9,10),_1402198,[sust(conc(sing,_1401244,comun,7,9,10),_1277528,(7,9,10,wheat,wheat,_12
33170))),_1402130,_1402132,_1402134)),_1267754,_1267756)),_1267166,_1267168),_1265762,_1265764)),_138987
2)),_1255384,_1255386)),_1254788,_1254790),w((7,10,11,?,?,_1363870))).

oracTag(conc(8,0,7),_72212,w((8,0,1,'In',in,_72420)),oracRelat(conc(rel,8,1,5),_1444400,pronEnglish(conc(_77552,_77
554,_77556,rel,_77560,_77562,_77564,8,1,2),_1444572,(8,1,2,which,which,_1406078)),sn(conc(sing,_87528,terc,comun
,suj,_87536,_87538,8,2,3),_77766,snCoordinado(conc(sing,_87528,terc,comun,suj,_87536,_87538,8,2,3),_1444798,snSi
mple(conc(sing,_87528,terc,comun,suj,_87536,_87538,8,2,3),_78886,snSimplePodado(conc(sing,_87528,terc,comun,suj
,_87536,_87538,8,2,3),_88456,_88408,_88410,nucleoSN(conc(sing,_87528,comun,8,2,3),_88482,[sust(conc(sing,_8752
8,comun,8,2,3),_1455372,(8,2,3,chromosome,chromosome,_1406182))),_88414,_88416,_88418)),_1444802,_1444804))
,nucleoVerbal(conc(_79948,_79950,presente,_79954,8,3,4),_91252,nucleoVerbalAux(conc(_79948,_79950,presente,_7
9954,8,3,4),_91278,verboSolo(conc(_79948,_79950,presente,verboSolo,8,3,4),_91298,verbo(conc(_79948,_79950,presen
te,be,8,3,4),_1461912,(8,3,4,are,be,_1406278))),_91236,_91238),_1444408,sn(conc(pl,_112848,terc,pronEnglish,suj,_
91648,_91650,8,4,5),_91604,snCoordinado(conc(pl,_112848,terc,pronEnglish,suj,_91648,_91650,8,4,5),_1463460,snSi
mple(conc(pl,_112848,terc,pronEnglish,suj,_91648,_91650,8,4,5),_92724,snSimplePodado(conc(pl,_112848,terc,pronE
nglish,suj,_91648,_91650,8,4,5),_11524,pronEnglish(conc(pl,_112848,terc,persNoRefl,suj,_112856,_112858,8,4,5),snSi
mplePodado(conc(pl,_112928,terc,comun,_112934,tiempo,_112938,7,5,10),_1389910,_1389862,_1389864,nucleoSN(co

nc(pl,_112928,comun,7,5,6),_1389936,[sust(conc(pl,_112928,comun,7,5,6),_1264890,(7,5,6,days,day,_1232788))],_1389868,postModificadorSN(conc(pl,_112928,_1261576,7,6,10),_1265462,oracRelat(conc(participio,7,6,10),_1265754,prepSimple(conc(_1232932,to,7,6,7),_1265902,(7,6,7,to,to,_1232884)),nucleoVerbal(conc(_1233026,_1233028,gerundio,verboSolo,7,7,8),_1400960,nucleoVerbalAux(conc(_1233026,_1233028,gerundio,verboSolo,7,7,8),_1400986,verboSolo(conc(_1233026,_1233028,gerundio,verboSolo,7,7,8),_1401006,verbo(conc(_1233026,_1233028,gerundio,resto,7,7,8),_1266598,(7,7,8,flowering,flowering,_1232978)))]),_1400944,_1400946),sp(conc(_1233124,of,7,8,10),_1267162,spSimple(conc(_1233124,of,7,8,10),_1393086,prep(conc(_1233124,of,7,8,9),_1267308,prepSimpleG(conc(_1233124,of,7,8,9),_1393114,prepSimple(conc(_1233124,resto,7,8,9),_1267430,(7,8,9,in,in,_1233076)))]),_1267312,_1267314),sn(conc(sing,_1401244,terc,comun,_1401250,_1401252,_1401254,7,9,10),_1393364,snCoordinado(conc(sing,_1401244,terc,comun,_1401250,_1401252,_1401254,7,9,10),_1267750,snSimple(conc(sing,_1401244,terc,comun,_1401250,_1401252,_1401254,7,9,10),_1394520,snSimplePodado(conc(sing,_1401244,terc,comun,_1401250,_1401252,_1401254,7,9,10),_1402172,_1402124,_1402126,nucleoSN(conc(sing,_1401244,comun,7,9,10),_1402198,[sust(conc(sing,_1401244,comun,7,9,10),_1277528,(7,9,10,wheat,wheat,_1233170)))]),_1402130,_1402132,_1402134),_1267754,_1267756)]),_1267166,_1267168),_1265762,_1265764),_1389872),(8,4,5,they,they,_1406376)))]),_1463464,_1463466),_1444412,nucleoVerbal(conc(_1406528,_1406530,participio,_1444246,8,5,6),_125814,nucleoVerbalAux(conc(_1406528,_1406530,participio,_1444246,8,5,6),_125840,verboSolo(conc(_1406528,_1406530,participio,verboSolo,8,5,6),_125860,verbo(conc(_1406528,_1406530,participio,resto,8,5,6),_14182,(8,5,6,located,locate,_1406480)))]),_125798,_125800),w((8,6,7,?,_75302))).

oracTag(conc(9,0,13),_205956,sn(conc(sing,_221644,terc,propio,suj,_221652,_221654,9,0,3),_213200,snCoordinado(conc(sing,_221644,terc,propio,suj,_221652,_221654,9,0,3),_134242,snSimple(conc(sing,_221644,terc,propio,suj,_221652,_221654,9,0,3),_214316,snSimplePodado(conc(sing,_221644,terc,propio,suj,_221652,_221654,9,0,3),_222572,_222524,_222526,nucleoSN(conc(sing,_221644,propio,9,0,3),_222598,[sust(conc(sing,_128300,propio,9,0,1),_143748,(9,0,1,'NUEVA_SERIE',nueva_serie,_128250)),sust(conc(sing,_128396,propio,9,1,2),_143930,(9,1,2,'NUEVA_SERIE',nueva_serie,_128346)),sust(conc(sing,_221644,propio,9,2,3),_144112,(9,2,3,'NUEVA_SERIE',nueva_serie,_128442)))]),_222530,_222532,_222534),_134246,_134248),sn(conc(sing,_229536,prim,pronEnglish,suj,_228454,_228456,9,3,4),_225732,snCoordinado(conc(sing,_229536,prim,pronEnglish,suj,_228454,_228456,9,3,4),_152284,snSimple(conc(sing,_229536,prim,pronEnglish,suj,_228454,_228456,9,3,4),_227034,snSimplePodado(conc(sing,_229536,prim,pronEnglish,suj,_228454,_228456,9,3,4),_229560,pronEnglish(conc(sing,_229536,prim,persNoRefI,suj,_229544,_229546,9,3,4),_167868,(9,3,4,'I',_128538)))]),_152288,_152290),nucleoVerbal(conc(_228280,_228282,presente,_228286,9,4,7),_252626,nucleoVerbalAux(conc(_228280,_228282,presente,_228286,9,4,5),_170006,verboSolo(conc(_228280,_228282,presente,verboSolo,9,4,5),_171742,verbo(conc(_228280,_228282,presente,resto,9,4,5),_171790,(9,4,5,want,want,_128642)))]),_252610,perif_rasis(conc(9,5,7),_252696,w((9,5,6,to,to,_252683)),verbo(conc(_128882,_128884,infinitivo,resto,9,6,7),_172288,(9,6,7,increase,increase,_128834)))]),sn(conc(_231944,_231946,terc,cantidad,_231952,_231954,_231956,9,7,12),_230856,snCoordinado(conc(_231944,_231946,terc,cantidad,_231952,_231954,_231956,9,7,12),_173122,snSimple(conc(_231944,_231946,terc,cantidad,_231952,_231954,_231956,9,7,12),_231786,snSimplePodado(conc(_231944,_231946,terc,cantidad,_231952,_231954,_231956,9,7,12),_178076,cantidadIngles(conc(9,7,8),_178420,w((9,7,8,grains,grain,_209040)))]),_178080,_178082,sp(conc(_129076,of,9,8,12),_178704,spSimple(conc(_129076,of,9,8,12),_233828,prep(conc(_129076,of,9,8,9),_178850,prepSimpleG(conc(_129076,of,9,8,9),_233856,prepSimple(conc(_129076,resto,9,8,9),_178972,(9,8,9,per,per,_129028)))]),_178854,_178856),sn(conc(sing,_241986,terc,comun,_241992,_241994,_241996,9,9,12),_234090,snCoordinado(conc(sing,_241986,terc,comun,_241992,_241994,_241996,9,9,12),_179292,snSimple(conc(sing,_241986,terc,comun,_241992,_241994,_241996,9,9,12),_235230,snSimplePodado(conc(sing,_241986,terc,comun,_241992,_241994,_241996,9,9,12),_242914,_242866,_242868,nucleoSN(conc(sing,_241986,comun,9,9,10),_242940,[sust(conc(sing,_241986,comun,9,9,10),_188798,(9,9,10,spike,spike,_129122)))]),_242872,_242874,sp(conc(_129266,of,9,10,12),_191680,spSimple(conc(_129266,of,9,10,12),_244730,prep(conc(_129266,of,9,10,11),_191826,prepSimpleG(conc(_129266,of,9,10,11),_244758,prepSimple(conc(_129266,resto,9,10,11),_191948,(9,10,11,in,in,_129218)))]),_191830,_191832),sn(conc(sing,_252950,terc,comun,_252956,_252958,_252960,9,11,12),_245016,snCoordinado(conc(sing,_252950,terc,comun,_252956,_252958,_252960,9,11,12),_192268,snSimple(conc(sing,_252950,terc,comun,_252956,_252958,_252960,9,11,12),_246180,snSimplePodado(conc(sing,_252950,terc,comun,_252956,_252958,_252960,9,11,12),_253878,_253830,_253832,nucleoSN(conc(sing,_252950,comun,9,11,12),_253904,[sust(conc(sing,_252950,comun,9,11,12),_201774,(9,11,12,barley,barley,_129312)))]),_253836,_253838,_253840),_192272,_192274)))]),_191684,_191686),_179296,_179298),_178708,_178710),_173126,_173128),w((9,12,13,';',_211994))).

oracTag(conc(10,0,7),_450646,oracRelat(conc(rel,10,0,4),_259890,pronEnglish(conc(_455470,_455472,_455474,rel,_455478,_455480,_455482,10,0,1),_260062,(10,0,1,'Which',which,_258738)),sn(conc(pl,_465468,terc,comun,compl,_465476,_465478,10,1,2),_455684,snCoordinado(conc(pl,_465468,terc,comun,compl,_465476,_465478,10,1,2),_260288,snSimple(conc(pl,_465468,terc,comun,compl,_465476,_465478,10,1,2),_456804,snSimplePodado(conc(pl,_465468,terc,comun,compl,_465476,_465478,10,1,2),_466396,_466348,_466350,nucleoSN(conc(pl,_465468,comun,10,1,2),_269794,(10,1,2,'QTLs',qtls,_258842)))]),_466354,_466356,_466358),_260292,_260294),nucleoVerbal(conc(_457866,_457868,pasiva,compBe,10,2,4),_469192,nucleoVerbalAux(conc(_457866,_457868,pasiva,compBe,10,2,4),_469218,verbCompBe(conc(_457866,_457868,pasiva,compBe,10,2,4),_469246,verbo(conc(_457866,_4

57868, presente, be, 10, 2, 3), _274314, (10, 2, 3, are, be, _258938)), _469228, _469230, _469232, verbo(conc(_259084, _259086, p, articipio, resto, 10, 3, 4), _274818, (10, 3, 4, related, relate, _259036))), _469176, _469178), _259898, _259900, _259902), w((10, 4, 5, to, to, _452686)), w((10, 5, 6, this, this, _452942)), w((10, 6, 7, ?, ?, _453200))).

oracTag(conc(11, 0, 8), _581608, oracRelat(conc(rel, 11, 0, 7), _473210, pronEnglish(conc(_587196, _587198, _587200, rel, _587204, _587206, _587208, 11, 0, 1), _473382, (11, 0, 1, 'What', what, _471970)), sn(conc(pl, _632706, terc, comun, compl, _632714, _632716, 11, 1, 3), _587410, snCoordinado(conc(pl, _632706, terc, comun, compl, _632714, _632716, 11, 1, 3), _473608, snSimple(conc(pl, _632706, terc, comun, compl, _632714, _632716, 11, 1, 3), _588530, snSimplePodado(conc(pl, _632706, terc, comun, compl, _632714, _632716, 11, 1, 3), _598122, _598074, preModificadorSN(conc(pl, _632706, adj, 11, 1, 2), _482492, sadjetival(conc(pl, _632706, 11, 1, 2), _482524, _482526, adjetivo(conc(pl, _632706, _482556, 11, 1, 2), _482736, [adjSimple(conc(pl, _632706, cal, 11, 1, 2), _482882, (11, 1, 2, other, other, _472074))), _482740, _482742), _482530)), nucleoSN(conc(pl, _632706, comun, 11, 2, 3), _598148, [sust(conc(pl, _632706, comun, 11, 2, 3), _483482, (11, 2, 3, traits, trait, _472170))), _598080, _598082, _598084), _473612, _473614)), nucleoVerbal(conc(_589592, _589594, pasiva, compBe, 11, 3, 5), _601712, nucleoVerbalAux(conc(_589592, _589594, pasiva, compBe, 11, 3, 5), _601738, verbCompBe(conc(_589592, _589594, pasiva, compBe, 11, 3, 5), _601766, verbo(conc(_589592, _589594, presente, be, 11, 3, 4), _488002, (11, 3, 4, are, be, _472266))), _601748, _601750, _601752, verbo(conc(_472412, _472414, participio, resto, 11, 4, 5), _488506, (11, 4, 5, affected, affect, _472364))), _601696, _601698), sp(conc(_472510, resto, 11, 5, 7), _489070, spSimple(conc(_472510, resto, 11, 5, 7), _602146, prep(conc(_472510, resto, 11, 5, 6), _489216, prepSimpleG(conc(_472510, resto, 11, 5, 6), _602174, prepSimple(conc(_472510, resto, 11, 5, 6), _489338, (11, 5, 6, by, by, _472462))), _489220, _489222), sn(conc(pl, _624064, terc, pronEnglish, compl, _602440, _602442, 11, 6, 7), _602396, snCoordinado(conc(pl, _624064, terc, pronEnglish, compl, _602440, _602442, 11, 6, 7), _489658, snSimple(conc(pl, _624064, terc, pronEnglish, compl, _602440, _602442, 11, 6, 7), _603524, snSimplePodado(conc(pl, _624064, terc, pronEnglish, compl, _602440, _602442, 11, 6, 7), _505166, pronEnglish(conc(pl, _624064, terc, persNoRefl, compl, _624072, _624074, 11, 6, 7), _505242, (11, 6, 7, them, them, _472556))), _489662, _489664))), _489074, _489076), _473220, _473222), w((11, 7, 8, ?, ?, _585226))).

oracTag(conc(12, 0, 2), _660972, sn(conc(sing, _666138, terc, propio, suj, _666134, _666148, 12, 0, 1), _662908, snCoordinado(conc(sing, _666138, terc, propio, suj, _666134, _666148, 12, 0, 1), _647712, snSimple(conc(sing, _666138, terc, propio, suj, _666134, _666148, 12, 0, 1), _664024, snSimplePodado(conc(sing, _666138, terc, propio, suj, _666134, _666148, 12, 0, 1), _667104, _667052, _667054, nucleoSN(conc(sing, _666138, propio, 12, 0, 1), _667130, [sust(conc(sing, _666138, propio, 12, 0, 1), _657218, (12, 0, 1, 'NUEVA_SERIE', nueva_serie, _642782))), _667058, _667060, _667062)), _647716, _647718)), w((12, 1, 2, '!', '!', _661904))).

oracTag(conc(13, 0, 2), _692254, sn(conc(sing, _697420, terc, propio, suj, _697416, _697430, 13, 0, 1), _694190, snCoordinado(conc(sing, _697420, terc, propio, suj, _697416, _697430, 13, 0, 1), _678994, snSimple(conc(sing, _697420, terc, propio, suj, _697416, _697430, 13, 0, 1), _695306, snSimplePodado(conc(sing, _697420, terc, propio, suj, _697416, _697430, 13, 0, 1), _698386, _698334, _698336, nucleoSN(conc(sing, _697420, propio, 13, 0, 1), _698412, [sust(conc(sing, _697420, propio, 13, 0, 1), _688500, (13, 0, 1, 'NUEVA_SERIE', nueva_serie, _674064))), _698340, _698342, _698344)), _678998, _679000)), w((13, 1, 2, '!', '!', _693186))).

oracTag(conc(14, 0, 2), _723526, sn(conc(sing, _728692, terc, propio, suj, _728688, _728702, 14, 0, 1), _725462, snCoordinado(conc(sing, _728692, terc, propio, suj, _728688, _728702, 14, 0, 1), _710266, snSimple(conc(sing, _728692, terc, propio, suj, _728688, _728702, 14, 0, 1), _726578, snSimplePodado(conc(sing, _728692, terc, propio, suj, _728688, _728702, 14, 0, 1), _729658, _729606, _729608, nucleoSN(conc(sing, _728692, propio, 14, 0, 1), _729684, [sust(conc(sing, _728692, propio, 14, 0, 1), _719772, (14, 0, 1, 'NUEVA_SERIE', nueva_serie, _705336))), _729612, _729614, _729616)), _710270, _710272)), w((14, 1, 2, '!', '!', _724458))).

oracTag(conc(15, 0, 13), _805004, sn(conc(sing, _814920, prim, pronEnglish, suj, _813798, _813800, 15, 0, 1), _811076, snCoordinado(conc(sing, _814920, prim, pronEnglish, suj, _813798, _813800, 15, 0, 1), _742826, snSimple(conc(sing, _814920, prim, pronEnglish, suj, _813798, _813800, 15, 0, 1), _812378, snSimplePodado(conc(sing, _814920, prim, pronEnglish, suj, _813798, _813800, 15, 0, 1), _814944, pronEnglish(conc(sing, _814920, prim, persNoRefl, suj, _814928, _814930, 15, 0, 1), _758410, (15, 0, 1, 'I', i, _736828))), _742830, _742832)), nucleoVerbal(conc(_813624, _813626, _813628, compWill, 15, 1, 5), _826724, nucleoVerbalAux(conc(_813624, _813626, _813628, compWill, 15, 1, 3), _760548, verbCompWill(conc(_813624, _813626, _813628, compWill, 15, 1, 3), _761894, verbo(conc(_736980, _736982, xx, auxiliary, 15, 1, 2), _762206, (15, 1, 2, would, would, _736932))), _761898, _761900, verbo(conc(_737078, _737080, infinitivo, resto, 15, 2, 3), _762484, (15, 2, 3, like, like, _737030))), _761904, _761906, _761908, _761910)), _826708, perifrasis(conc(15, 3, 5), _826794, w((15, 3, 4, to, to, _826781)), verbo(conc(_737270, _737272, i, infinitivo, resto, 15, 4, 5), _763564, (15, 4, 5, know, know, _737222))), sn(conc(sing, _827048, terc, propio, compl, _827056, _827058, 15, 5, 12), _816244, snCoordinado(conc(sing, _827048, terc, propio, compl, _827056, _827058, 15, 5, 12), _764398, snSimple(conc(sing, _827048, terc, propio, compl, _827056, _827058, 15, 5, 12), _827976, det(conc(sing, _827048, det, 15, 5, 6), _770562, art(conc(sing, _827048, det, 15, 5, 6), _770598, (15, 5, 6, the, the, _737320))), _827930, nucleoSN(conc(sing, _827048, propio, 15, 6, 7), _828002, [sust(conc(sing, _827048, propio, 15, 6, 7), _771924, (15, 6, 7, 'QTLS', qtls, _737416))), _827934, postModificadorSN(conc(sing, _827048, _770520, 15, 7, 12), _772456, oracRelat(conc(participio, 15, 7, 12), _772742, _772744, nucleoVerbal(conc(_737560, _737562, participio, verboSolo, 15, 7, 10), _839940, nucleoVerbalAux(conc(_737560, _737562, participio, verboSolo, 15, 7, 8), _773130, verboSolo(conc(

c(_737560,_737562,participio,verboSolo,15,7,8),_773258,verbo(conc(_737560,_737562,participio,resto,15,7,8),_773306,(15,7,8,related,relate,_737512))))),_839924,perifrasis(conc(15,8,10),_840010,w((15,8,9,to,to,_839997)),verbo(conc(_737752,_737754,infinitivo,resto,15,9,10),_773804,(15,9,10,rust,rust,_737704))))),sp(conc(_737850,of,15,10,12),_773878,spSimple(conc(_737850,of,15,10,12),_832090,prep(conc(_737850,of,15,10,11),_774024,prepSimpleG(conc(_737850,of,15,10,11),_832118,prepSimple(conc(_737850,resto,15,10,11),_774146,(15,10,11,in,in,_737802))))),_774028,_774030),sn(conc(sing,_840264,terc,comun,_840270,_840272,_840274,15,11,12),_832360,snCoordinado(conc(sing,_840264,terc,comun,_840270,_840272,_840274,15,11,12),_774466,snSimple(conc(sing,_840264,terc,comun,_840270,_840272,_840274,15,11,12),_833508,snSimplePodado(conc(sing,_840264,terc,comun,_840270,_840272,_840274,15,11,12),_841192,_841144,_841146,nucleoSN(conc(sing,_840264,comun,15,11,12),_841218,[sust(conc(sing,_840264,comun,15,11,12),_783972,(15,11,12,wheat,wheat,_737896))))),_841150,_841152,_841154)),_774470,_774472))))),_773882,_773884),_772750,_772752)),_827938)),_764402,_764404)),w((15,12,13,','',',',_810020)))).

oracTag(conc(16,0,10),_960750,oracRelat(conc(rel,16,0,9),_846444,pronEnglish(conc(_968162,_968164,_968166,rel,_968170,_968172,_968174,16,0,1),_846616,(16,0,1,'Which',which,_845024)),sn(conc(pl,_978136,terc,comun,suj,_978144,_978146,16,1,2),_968376,snCoordinado(conc(pl,_978136,terc,comun,suj,_978144,_978146,16,1,2),_846842,snSimple(conc(pl,_978136,terc,comun,suj,_978144,_978146,16,1,2),_969496,snSimplePodado(conc(pl,_978136,terc,comun,suj,_978144,_978146,16,1,2),_979064,_979016,_979018,nucleoSN(conc(pl,_978136,comun,16,1,2),_979090,[sust(conc(pl,_978136,comun,16,1,2),_856348,(16,1,2,genes,gene,_845128))))),_979022,_979024,_979026)),_846846,_846848)),nucleoVerbal(conc(_970558,_970560,presente,_970564,16,2,3),_1014354,nucleoVerbalAux(conc(_970558,_970560,presente,_970564,16,2,3),_1014380,verboSolo(conc(_970558,_970560,presente,verboSolo,16,2,3),_1014400,verbo(conc(_970558,_970560,presente,resto,16,2,3),_861796,(16,2,3,influence,influence,_845224))))),_1014338,_1014340),_846452,sn(conc(sing,_992942,terc,comun,compl,_992950,_992952,16,3,9),_982212,snCoordinado(conc(sing,_992942,terc,comun,compl,_992950,_992952,16,3,9),_862988,snSimple(conc(sing,_992942,terc,comun,compl,_992950,_992952,16,3,9),_983332,snSimplePodado(conc(sing,_992942,terc,comun,compl,_992950,_992952,16,3,9),_993870,det(conc(sing,_992942,det,16,3,4),_869152,art(conc(sing,_992942,det,16,3,4),_869188,(16,3,4,the,the,_845322))))),_993824,nucleoSN(conc(sing,_992942,comun,16,4,5),_993896,[sust(conc(sing,_992942,comun,16,4,5),_870514,(16,4,5,resistance,resistance,_845418))))),_993828,_993830,sp(conc(_845562,of,16,5,9),_873744,spSimple(conc(_845562,of,16,5,9),_996932,prep(conc(_845562,of,16,5,6),_873890,prepSimpleG(conc(_845562,of,16,5,6),_996960,prepSimple(conc(_845562,to,16,5,6),_874012,(16,5,6,to,to,_845514))))),_873894,_873896),sn(conc(pl,_1005058,terc,comun,_1005064,_1005066,_1005068,16,6,9),_997198,snCoordinado(conc(pl,_1005058,terc,comun,_1005064,_1005066,_1005068,16,6,9),_874332,snSimple(conc(pl,_1005058,terc,comun,_1005064,_1005066,_1005068,16,6,9),_998342,snSimplePodado(conc(pl,_1005058,terc,comun,_1005064,_1005066,_1005068,16,6,9),_1005986,_1005938,_1005940,nucleoSN(conc(pl,_1005058,comun,16,6,7),_1006012,[sust(conc(pl,_1005058,comun,16,6,7),_883838,(16,6,7,plagues,plague,_845608))))),_1005944,_1005946,sp(conc(_845752,of,16,7,9),_886720,spSimple(conc(_845752,of,16,7,9),_1007808,prep(conc(_845752,of,16,7,8),_886866,prepSimpleG(conc(_845752,of,16,7,8),_1007836,prepSimple(conc(_845752,resto,16,7,8),_886988,(16,7,8,in,in,_845704))))),_886870,_886872),sn(conc(sing,_1014638,terc,comun,_1014644,_1014646,_1014648,16,8,9),_1008098,snCoordinado(conc(sing,_1014638,terc,comun,_1014644,_1014646,_1014648,16,8,9),_887308,snSimple(conc(sing,_1014638,terc,comun,_1014644,_1014646,_1014648,16,8,9),_1009266,snSimplePodado(conc(sing,_1014638,terc,comun,_1014644,_1014646,_1014648,16,8,9),_1015566,_1015518,_1015520,nucleoSN(conc(sing,_1014638,comun,16,8,9),_1015592,[sust(conc(sing,_1014638,comun,16,8,9),_896814,(16,8,9,wheat,wheat,_845798))))),_1015524,_1015526,_1015528)),_887312,_887314)),_886724,_886726)),_874336,_874338)),_873748,_873750)),_862992,_862994)),_846456),w((16,9,10,?,?,_966208)))).

Reunido el Tribunal que suscribe en el día de la fecha acordó otorgar, por _____ a la Tesis Doctoral de Yenory Rojas Hernández la calificación de

_____, de _____ de

El Presidente,

El Secretario,

**UNIVERSIDAD DE ALICANTE
CEDIP**

La presente Tesis de Yenory Rojas Hernández ha sido registrada con el no. _____ del registro de entrada correspondiente.

_____, de _____ de

El encargado del registro,

Universitat d'Alacant
Universidad de Alicante