Validación de Respuestas: Metodología de Evaluación y Desarrollo de Sistemas

Álvaro Rodrigo Yuste

Universidad Nacional de Educación a Distancia

Programa de Posgrado en Inteligencia Artificial y Sistemas Informáticos

Validación de Respuestas: Metodología de Evaluación y Desarrollo de Sistemas

Tesis de Máster

Álvaro Rodrigo Yuste

Ingeniero en Informática

Director: Dr. Anselmo Peñas Padilla

Departamento de Lenguajes y Sistemas Informáticos

Escuela Técnica Superior de Ingeniería Informática

Universidad Nacional de Educación a Distancia

a mis padres y a mi hermano

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

En los últimos años la cantidad de información digital disponible ha aumentado enormemente y ha habido una explosión de las comunicaciones entre ordenadores como vía de transmisión de información entre usuarios. Esta gran cantidad de información disponible ha impulsado la investigación en sistemas de información textual para facilitar la localización, acceso y tratamiento de toda esta ingente cantidad de datos.

Los motores de búsqueda de documentos en Internet son los sistemas más conocidos de este tipo. Sin embargo, una vez recuperados los documentos, el usuario tiene que seleccionar aquellos con contenido relacionado con la información solicitada y posteriormente buscar en ellos la información deseada, lo que a veces supone una gran cantidad de tiempo y esfuerzo.

Este problema ha provocado una creciente necesidad de sistemas que permitan a un usuario formular sus necesidades de información empleando un lenguaje cotidiano y recibir una respuesta rápida, precisa y escueta. Éste es el objetivo de los sistemas de Búsqueda de Respuestas (Vicedo, 2003).

En este trabajo se propone una metodología para evaluar sistemas de Validación de Respuestas, los cuáles pueden mejorar el rendimiento de los sistemas de Búsqueda de Respuestas. Además, en este trabajo también se describen el planteamiento, desarrollo y prueba de un sistema de este tipo.

1.1. Motivación

Detrás de un sistema de Búsqueda de Respuestas tiene desarrollo un complejo procesamiento que incluye la clasificación de la pregunta y del tipo de la respuesta, la recuperación de textos, extracción de la respuesta, validación de la respuesta, etc. Este complejo procesamiento requiere no solo algún tipo de tratamiento de textos, sino también el uso de grandes recursos como bases de datos léxicas, diccionarios, colecciones de paráfrasis, etc. La adquisición, tratamiento y representación efectivas de este conocimiento para ser usado por un sistema de Búsqueda de Respuestas es todavía

un desafío abierto. Por este motivo, el trabajo en semántica, útil para la Búsqueda de Respuestas, se vería enriquecido con la colaboración entre las comunidades de Representación del Conocimiento y de Procesamiento del Lenguaje Natural. De hecho, los sistemas con mejor rendimiento ya han incorporado algún tipo de inferencia o razonamiento para identificar las respuestas candidatas a una pregunta (Harabagiu et al., 2003), (Moldovan et al., 2003).

Uno de los elementos de un sistema de Búsqueda de Respuestas donde introducir más procesamiento lingüístico, conocimiento y técnicas de aprendizaje automático es el módulo de Validación de Respuestas. Un sistema de Validación de Respuestas recibe una tripleta formada por una Pregunta, una Respuesta Candidata y un Texto Soporte, y devuelve un valor booleano que indica si la Respuesta a la Pregunta es o no es correcta según el Texto Soporte. El primer Ejercicio de Validación de Respuestas, Answer Validation Exercise (AVE 2006)[1](Peñas et al., 2007), se lanzó dentro del Cross Language Evaluation Forum (CLEF)[2] con el fin de promover el desarrollo y evaluación de este tipo de sistemas ideados para validar la veracidad de las respuestas devueltas por los sistemas de Búsqueda de Respuestas.

La Validación de Respuestas es una tarea que introduce otra oportunidad para el razonamiento y, por tanto, de colaboración entre las comunidades de Representación del Conocimiento y de Procesamiento del Lenguaje Natural: el texto que soporta la respuesta debe implicar de algún modo a la respuesta dada. De este modo la idea de Validación de Respuestas se puede reformular como un problema de Reconocimiento de Implicación Textual, en inglés Recognising Textual Entailment (RTE) (Dagan et al., 2006). En este trabajo se estudian y desarrollan las metodologías necesarias para reformular el problema de Validación de Respuestas en términos de Implicación Textual. Además se aportan una serie de experimentos realizados en foros internacionales de evaluación competitiva.

1.2. Objetivos del trabajo

Los objetivos principales de este trabajo de investigación son lo que a continuación se enumeran:

- Realizar una propuesta metodológica de desarrollo y evaluación de sistemas de Validación de Respuestas. Las respuestas serán las devueltas por los sistemas de Búsquedas de Respuestas y la validación se fundamentará en Implicación Textual.
- Hacer un estudio sobre los trabajos actuales en reconocimiento de Implicación Textual.

[1] http://nlp.uned.es/QA/ave/
[2] http://www.clef-campaign.org/

- Estudiar el estado actual de la investigación sobre el reconocimiento de entidades, estudiando una posible aplicación a Implicación Textual y Validación de respuestas.
- Desarrollar un prototipo para la tarea de reconocimiento de Implicación Textual que se base en el reconocimiento de entidades.
- Proponer un sistema basado en entidades e Implicación Textual para la tarea de Validación de Respuestas.
- Participar en campañas internacionales de evaluación tanto de Implicación Textual como de Validación de Respuestas.
- Plantear líneas de trabajo futuro consecuentes con los resultados obtenidos.

1.3. Estructura de la memoria

En el capítulo 2 (página 15) se exponen los conceptos básicos manejados a lo largo del trabajo. Para ello, en este capítulo se estudian los problemas de reconocimiento de entidades y reconocimiento de Implicación Textual. Además se muestran parte de los enfoques existentes actualmente para tratar de resolver dichos problemas.

En el capítulo 3 (página 29) se expone el problema de la Validación de Respuestas y su contexto dentro de la Búsqueda de Respuestas. Además, en este capítulo se propone una metodología para evaluar los sistemas de Validación de Respuestas reformulando el problema como uno de Implicación Texual, mostrando también cómo implementar dicha metodología. Finalmente, en este capítulo se describe cómo se puso en marcha dicha metodología dentro de un foro de evaluación internacional.

Basándose en el estudio realizado sobre entidades e Implicación Textual, en el capítulo 4 (página 41) se describe la construcción de un sistema de Implicación Textual basado en el reconocimiento de entidades. También se muestran los resultados y las conclusiones obtenidas por este sistema en su participación en un foro de evaluación internacional.

Habiendo sido redefinida la Validación de Respuestas como un problema de Implicación Textual, en el capítulo 5 (página 51) se describe un sistema para validar respuestas usando Implicación Textual y entidades. También se muestran los resultados obtenidos por este sistema al participar en la tarea de Validación de Respuestas descrita.

En el capítulo 6 (página 57) se recogen las conclusiones obtenidas a lo largo del trabajo realizado.

Finalmente, las líneas futuras sobre las que continuar la investigación del presente trabajo se exponen en el capítulo 7 (página 59).

Capítulo 2

Preliminares

Los experimentos que se han realizado y que se van a relatar a lo largo de este trabajo parten del uso de reconocimiento de entidades en Implicación Textual y Validación de Respuestas. Por ello, a lo largo de este capítulo se definirán los conceptos de reconocimiento de entidades y de Implicación Textual. También se mostrarán los principales foros de evaluación centrados en estos dos temas y algunos trabajos existentes de interés para alcanzar los objetivos propuestos.

2.1. Reconocimiento de entidades

2.1.1. Definición de entidad

El término entidad nombrada, en inglés Named Entity (NE), ha sido utilizado de forma más o menos semejante en distintas fuentes. En las conferencias MUC[1] (Message Understanding Conference) (Chinchor, 1998) y CoNLL[2] (Conferences on Computacional Natural Language Learning) (Sang, 2002) y en (Palmer and Day, 1997) se define a las entidades nombradas como las frases que son identificadores únicos de entidades (organizaciones, personas y localidades), las expresiones temporales (fechas o expresiones de tiempo como puede ser *mediodía*) y las expresiones numéricas (porcentajes o cantidades monetarias). La importancia que tiene realizar esta detección reside en que según (Chinchor, 1998) los nombres, fechas y números son importantes a la hora de tratar textos que sirven de fuente a bases de datos.

En Wikipedia [3] se indica que en la expresión entidades nombradas la palabra *nombrada* restringe las entidades a aquellas que solamente usan uno o varios identificadores concretos para identificar al referente. Así, en el programa ACE (Automatic Content Extraction)[4](Doddington et al., 2004),

[1] http://www.cs.nyu.edu/cs/faculty/grishman/muc6.html
[2] http://www.cnts.ua.ac.be/conll2002/
[3] http://wikipedia.org/
[4] http://www.nist.gov/speech/tests/ace/index.htm

no se habla de entidades nombradas sino solamente de entidades. Según el ACE, una entidad se define simplemente como un objeto o conjunto de objetos en el mundo.

Para el programa ACE no solo es importante detectar las entidades sino también lo que ellos denominan *menciones* de entidades y que son cualquier referencia que se haga a una entidad. Se puede hacer referencia a una entidad haciendo uso de un sustantivo común, de un sintagma nominal o usando un pronombre. Por ejemplo, en el texto de la figura 2.1, la expresión *el científico más conocido e importante del siglo XX* es una mención a Albert Einstein.

```
Albert Einstein, nacido en Alemania y nacionalizado en
Estados Unidos en 1940, es el científico más conocido e
importante del siglo XX
```

Figura 2.1: Texto de ejemplo de menciones a entidades.

Usando las fuentes citadas anteriormente como referencia, nosotros distinguimos tres tipos de entidades: las nombradas, las expresiones numéricas y las expresiones temporales.

2.1.2. Tipos de entidades

A lo largo del tiempo se han ido dando distintas jerarquías sobre qué se consideraba una entidad nombrada y su tipo. Una de las primeras y que sirvió como referencia fue la de la conferencia MUC-6 (Grishman and Sundheim, 1996). Esta jerarquía constaba principalmente de tres categorías:

- Nombres propios, acrónimos y otros identificadores únicos que se subdividían en los siguientes tipos:
  - Organizaciones: nombres de corporaciones, organizaciones gubernamentales o de otro tipo.
  - Personas: nombres de personas.
  - Localidades: nombres de localidades políticas o geográficas como ciudades, provincias, países, masas de agua, montañas, etc.
- Expresiones temporales que se subdividen en los siguientes tipos:
  - Fecha: fechas completas o parciales.
  - Tiempo: una expresión temporal completa o parcial de un tiempo del día (como por ejemplo 5 p.m.).
- Expresiones numéricas que pueden estar expresadas por números o letras, con los siguientes subtipos:
  - Dinero: expresiones monetarias.

- Porcentajes.

En 2003 Satoshi Sekine creó una jerarquía extendida de entidades nombradas[5] que fue diseñada para reunir las necesidades crecientes de un amplio rango de tipos de entidades nombradas. Esta jerarquía fue creada a partir de la jerarquía anteriormente descrita del MUC, el conjunto de entidades nombradas desarrollado para el proyecto IREX (Sekine and Isahara, 2000) y la jerarquía extendida de entidades nombradas que contiene aproximadamente 150 tipos (Sekine et al., 2002). Esta jerarquía está dividida en tres grandes clases que, al igual que en el MUC y el proyecto IREX, son: nombre, expresiones numéricas y expresiones temporales.

2.1.3. Definición reconocimiento de entidades

La tarea de reconocimiento de entidades nombradas, en inglés Named Entity Recognition (NER), consiste en dadas unas categorías predefinidas de entidades nombradas de interés, localizar de forma automática en un texto dado todas las palabras que sean instancias de dichas categorías. Por otro lado, una vez detectadas las entidades nombradas se entiende por clasificación de las mismas el proceso de otorgar a cada entidad nombrada una determinada categoría semántica (ej: *George Bush* es de tipo persona).

Si por ejemplo se consideran las categorías persona, organización, localidad y expresión temporal, el reconocimiento y clasificación de entidades nombradas sobre el texto de la figura 2.2 daría lugar a una anotación similar a la de la figura 2.3.

```
William Henry Gates III (Seattle, Washington, Estados Unidos,
28 de octubre de 1955) más conocido como Bill Gates, es un
empresario y filántropo estadounidense, cofundador de la
empresa de software Microsoft, productora del sistema
operativo para computadoras personales más utilizado en el
mundo. (...) En 1976, abandonó la universidad y se trasladó
a Albuquerque.
```

Figura 2.2: Ejemplo de texto donde anotar entidades nombradas.

El reconocimiento y clasificación de entidades nombradas puede servir tanto para poblar ontologías de determinados dominios (Tanev and Magnini, 2006), como para aportar información a sistemas de pregunta-respuesta o sistemas de extracción de información (Cucerzan and Yarowsky, 1999). En el caso de los sistemas de extracción de información se ha convertido en una tarea de suma importancia por su capacidad de proveer de información útil para correferencia y rellenado de plantillas (Palmer and Day, 1997).

[5] http://nlp.cs.nyu.edu/ene/

[William Henry Gates III]$_{persona}$ ([Seattle]$_{localidad}$, [Washington]$_{localidad}$, [Estados Unidos]$_{localidad}$, [28 de octubre de 1955]$_{exp.temporal}$)más conocido como [Bill Gates]$_{persona}$, es un empresario y filántropo estadounidense, cofundador de la empresa de software [Microsoft]$_{organizacion}$, productora del sistema operativo para computadoras personales más utilizado en el mundo. (...) En [1976]$_{exp.temporal}$, abandonó la universidad y se trasladó a [Albuquerque]$_{localidad}$.

Figura 2.3: Ejemplo de texto con las entidades nombradas marcadas.

Las técnicas de reconocimiento y clasificación de entidades nombradas pueden agruparse en métodos supervisados, métodos no supervisados y métodos ligeramente supervisados según el grado de supervisión que se necesite en el proceso de entrenamiento.

2.1.4. Evaluación

El rendimiento de los sistemas de reconocimiento de entidades nombradas se suele medir en términos de precisión y cobertura (recall en inglés) tal y como se define en (Grishman and Sundheim, 1996). La precisión es la proporción de entidades propuestas correctamente por el sistema (fórmula 2.1) y la cobertura es la proporción de entidades existentes en el texto que el sistema ha propuesto correctamente (ver fórmula 2.2). Estas dos medidas se suelen combinar en una única llamada medida F y que se define como la media armónica entre la precisión y la cobertura como se indica en 2.3.

$$precision = \frac{|entidades\ propuestas\ correctamente|}{|entidades\ propuestas|} \tag{2.1}$$

$$cobertura = \frac{|entidades\ propuestas\ correctamente|}{|entidades\ en\ el\ texto|} \tag{2.2}$$

$$F = \frac{2 * cobertura * precision}{cobertura + precision} \tag{2.3}$$

Supongamos, por ejemplo, un sistema creado para detectar nombres de localidades que detecta en el texto de la figura 2.3 Washington, Estados Unidos y Microsoft. El sistema ha detectado de forma correcta dos localidades (Washington y Estados Unidos), por lo que obtiene una precisión de 0.67 (67 %). En cuanto a la cobertura, de las cuatro localidades existentes en el texto (Seattle, Washington, Estados Unidos y Albuquerque), el sistema ha detectado dos, por lo que el valor de cobertura es de 0.5 (50 %). La medida F de este sistema para identificar localidades sería de 0.57 (57 %).

2.1.5. Métodos supervisados

En los métodos supervisados (Collins and Singer, 1999) el proceso de entrenamiento no es totalmente automático ya que se necesita la intervención humana. Esta intervención consiste en la anotación manual de los tipos de entidades nombradas de una colección de documentos para posteriormente aprender las reglas de clasificación a partir de los ejemplos. Al no ser totalmente automático, el proceso de entrenamiento es más duradero ya que se tarda mucho tiempo en tareas de anotación de documentos, además de ser costoso. Otro inconveniente adicional es que cada vez que se quiera realizar el entrenamiento para un nuevo idioma se tiene que repetir el proceso de anotación con el coste tanto de tiempo como económico que ello supone. Este tipo de métodos ha sido aplicado con éxito a la tarea conjunta de reconocimiento y clasificación de entidades nombradas obteniendo resultados superiores al 90 % de precisión (Bikel et al., 1997).

Un método supervisado que trata de ser lo más independiente posible del idioma es el descrito en (McNamee and Mayfield, 2002). Esta independencia se refiere sobre todo a no usar herramientas específicas como por ejemplo etiquetadores morfosintácticos. Lo que se usa de un idioma es una lista de las 1000 palabras más frecuentes de dicho idioma. El método construye un clasificador para cada posible tipo de salida y aunque no obtiene muy buenos resultados (una F del 60 %), hay que tener en cuenta que usa pocos recursos lingüísticos. Con este método se consigue mejorar un poco los resultados usando dos etapas: una de detección de las NE y otra de clasificación.

Otro método supervisado es el utilizado en (Cucerzan and Yarowsky, 2002), donde se usa el algoritmo descrito en (Cucerzan and Yarowsky, 1999). Este método varía respecto del original en que la lista inicial para el entrenamiento se obtiene de un corpus anotado. El algoritmo usa un chunker y puede hacer uso de etiquetadores morfosintácticos para obtener mejores resultados (se obtiene una F alrededor del 75 % dependiendo del idioma).

2.1.6. Métodos no supervisados

En los métodos no supervisados (Collins and Singer, 1999), la anotación de la colección de documentos se hace de forma totalmente automática. Estos métodos necesitan de menor tiempo y son menos costosos ya que el coste asociado a crear colecciones anotadas a mano es alto (Cucerzan and Yarowsky, 1999). Como principal ventaja presentan el que son más fáciles de adaptar a cada nuevo idioma entrenando al clasificador con una colección de documentos del idioma objetivo. La utilidad de estos métodos se puede ver en por ejemplo los sistemas de Búsqueda de Respuestas, los cuáles requieren sistemas de reconocimiento de entidades nombradas que sean fáciles de portar a distintos dominios (Kwak and Cha, 2005).

2.1.7. Métodos ligeramente supervisados

En los últimos años se está explorando una posibilidad intermedia reduciendo al máximo la información anotada. En (Collins and Singer, 1999) se describen tres algoritmos no supervisados de clasificación de entidades nombradas. Estos algoritmos parten de la idea de que para muchas entidades nombradas basta con su ortografía (una entidad nombrada que por ejemplo contenga *Mr.* hace referencia a una persona) y contexto (una entidad nombrada afectada por una aposición cuyo núcleo es por ejemplo *presidente*, hace referencia a una persona como en el caso de *George Bush, el presidente estadounidense, ha...*) para clasificarlas. En las tres alternativas se comienza con un conjunto muy reducido de siete reglas (como que si la entidad nombrada contiene *Mr.* se refiere a persona) que se aplican a la colección de documentos que va a servir de entrenamiento y las reglas que se aprenden se aplican de nuevo recursivamente. Los tres algoritmos fueron aplicados por los autores y se obtuvo en los tres casos unos resultados en torno al 90 % de precisión. La dificultad de adaptar este método a otros idiomas es que requiere el uso de un analizador sintáctico. Mientras que encontrar una herramienta de este tipo para inglés no es difícil, los demás idiomas no siempre disponen de analizadores que obtengan buenos resultados.

En (Kwak and Cha, 2005) se describe un método para la clasificación de NE basado en el DL-CoTrain expuesto en (Collins and Singer, 1999). La diferencia es que solo se usa un etiquetador morfosintáctico y un chunker para las características contextuales en lugar de un analizador sintáctico. Este hecho hace que el método sea más fácil de adaptar a otros idiomas. A diferencia de (Collins and Singer, 1999) (donde se establecían las reglas iniciales a mano), las reglas iniciales se extraen de un pequeño corpus anotado de entrenamiento. Luego se aplican estas reglas a un corpus no anotado donde solo están marcadas las NEs. Como resultado final se obtiene una lista de decisión para clasificar las NEs. Además, los autores muestran que usando una cuarta parte del corpus anotado se obtienen resultados comparables a los de los métodos supervisados (82 % de F del método no supervisado por 83 % del supervisado).

Otro método intermedio aparece descrito en Cucerzan and Yarowsky (1999), donde se propone un método para reconocimiento y clasificación de entidades nombradas que trata de ser lo más independiente posible del idioma. El algoritmo propuesto usa para su entrenamiento textos sin anotar y parte de una lista de NE formada por entre 40 y 100 ejemplares por cada clase que se pretende manejar.

Básicamente el algoritmo se basa en cómo están escritas las palabras (hace uso de la idea de que en ciertas clases de entidades algunos prefijos y sufijos son útiles para la detección y clasificación) y en su contexto (ej: uso de *Mr.* o de *mayor of*).

En los experimentos que se realizaron para varios idiomas se obtuvieron

resultados superiores al 70 % de medida F. Para estos experimentos se usaron textos de entrenamiento sin anotar. Además no se usó ni información específica del idioma, ni tokenizadores, ni otras herramientas y se necesitó de como mucho 15 minutos de esfuerzo humano para el entrenamiento (tiempo empleado en la creación de las listas de palabras que se utilizan al principio).

2.1.8. Foros de evaluación

El reconocimiento y clasificación de entidades nombradas ha sido el eje de varios foros de evaluación. Uno de los primeros en introducirlo como tarea fue la serie de conferencias MUC en 1995 en su sexta edición (MUC-6) (Grishman and Sundheim, 1996). La intención era la de investigar sobre sistemas de reconocimiento de entidades nombradas que serían de gran utilidad en diversas tareas de procesamiento del lenguaje natural como la extracción de información. En 1998 se siguió realizando la tarea de reconocimiento de entidades nombradas en el MUC-7 (Chinchor, 1998), extendiéndola a chino y japonés (antes solo se realizaba en inglés).

```
La O
petición O
del O
Abogado B-PER
General I-PER
tiene O
lugar O
después O
de O
que O
un O
juez O
del O
Tribunal B-ORG
Supremo I-ORG
del O
estado O
de O
Victoria B-LOC
```

Figura 2.4: Ejemplo de anotación del CoNLL.

Durante 2002 y 2003 se realizaron dos tareas de reconocimiento y clasificación de entidades nombradas dentro del ciclo de conferencias CoNLL (Sang, 2002),(Sang and Meulder, 2003), orientadas a que la tarea se realizase lo más independiente posible del lenguaje. Por este motivo cada año

se realizó la tarea sobre dos idiomas distintos (español y holandés en 2002 y alemán e inglés en 2003).

En estas tareas la organización suministraba los datos de entrenamiento, donde cada palabra estaba marcada con su etiqueta POS y las entidades nombradas se marcaban usando el formato IOB (B indica el comienzo de la entidad, I las demás palabras de la entidad y O las palabras que no son de una entidad nombrada), seguido de la clase de entidad nombrada que es (por ejemplo PER para persona y ORG para organización). Se puede ver un ejemplo en la figura 2.4.

Actualmente, el National Institute of Standards and Technology (NIST)[6] organiza una serie de competiciones similares bajo el nombre Automatic Content Extraction Evaluation (ACE)[7] que se celebran normalmente cada año desde 2001. En esta competición hay tareas de reconocimiento de entidades, menciones a dichas entidades, valores, así como una tarea específica de reconocimiento y normalización de expresiones temporales.

2.2. Implicación Textual (RTE)

2.2.1. Definición

La tarea de reconocimiento de Implicación Textual, en inglés Recognising Textual Entailment (RTE), consiste en decidir, dados dos textos en lenguaje natural, si el significado de un texto implica el significado del otro texto (llamado hipótesis)(Dagan et al., 2006). Dicho de otro modo, se trata de averiguar si el significado de la hipótesis se puede obtener a partir del otro texto. Sean por ejemplos los dos textos siguientes:

1. Patricia y Eugenio se casaron en Madrid el 28 de junio.

2. Eugenio es el marido de Patricia.

Es evidente que el significado del segundo texto se puede deducir a partir del primero. En este caso diríamos que el primer texto implica al segundo.

Un fenómeno fundamental del lenguaje natural es la variabilidad de las expresiones semánticas, donde el mismo significado se puede expresar de diversas formas. La tarea RTE es un intento de unificar los esfuerzos para capturar el mayor número de inferencias semánticas necesarias por distintas aplicaciones de procesamiento del lenguaje natural. El deseo es que la investigación en este campo lleve al desarrollo de motores de implicación que puedan ser usados como un módulo independiente en muchas aplicaciones (de forma similar al uso actual de los analizadores o tokenizadores en las aplicaciones actuales de lenguaje natural), de tal modo que la noción de implicación textual sea independiente del tipo de tarea al que se aplique.

[6] http://www.nist.gov/

[7] http://www.nist.gov/speech/tests/ace/

2.2.2. Aplicaciones

Son muchas las aplicaciones de procesamiento del lenguaje natural que se beneficiarían al tener un módulo de Implicación Textual. A continuación se muestran algunos ejemplos de dichas tareas:

- Sistemas de Pregunta Respuesta(QA, Question Answering): la respuesta dada por los sistemas de QA tiene que ser implicada por el texto que soporta dicha respuesta. En esta aplicación, los sistemas de RTE podrían ayudar a validar las respuestas candidatas y así realizar una mejor selección de la respuesta final. Algunos trabajos recientes (Harabagiu and Hickl., 2006) han mostrado cómo el uso de sistemas de RTE puede mejorar el rendimiento de sistemas de Búsquedas de Respuestas.

- Resumen automático (SUM, Automatic Summarization): en un resumen, un fragmento con información redundante es candidato a ser eliminado si hay otro fragmento que lo implica. Por tanto, los sistemas de RTE servirían para eliminar fragmentos repetidos del resumen.

- Recuperación de información (IR, Information Retrieval): en este caso los documentos recuperados deberían implicar a la consulta que se ha lanzado para obtener dichos documentos. Los sistemas de RTE ayudarían en este caso a los de IR a verificar que los documentos recuperados son relevantes dada la consulta del usuario.

- Extracción de Información (IE, Information Extraction): las relaciones que se extraen a partir de un determinado documento deben de ser implicadas por dicho documento, de tal modo que los sistemas de RTE servirían para comprobar que las relaciones extraídas son correctas.

2.2.3. Foro de evaluación

A pesar del interés que tiene el campo de la implicación textual, no ha sido hasta fechas recientes cuando se han empezado a desarrollar foros de evaluación competitivos para poner a prueba los distintos enfoques de distintos grupos.

PASCAL RTE Challenge

En el año 2005 se celebró la primera edición del PASCAL RTE Challengue, que tenía como meta proporcionar una primera oportunidad para presentar y comparar diferentes aproximaciones para modelar y reconocer la implicación textual. A esta primera edición se presentaron 16 sistemas participantes de todo el mundo y ha sido tal el éxito, que se han celebrado nuevas ediciones en 2006 y 2007 con 23 y 26 participantes respectivamente.

Cuadro 2.1: Número de pares del PASCAL RTE en cada edición.

Año	**Cjto. Desarrollo**	**Cjto. Test**
2005	567	800
2006	800	800
2007	799	800

Los sistemas participantes reciben una serie de pares de fragmentos de texto llamados Texto-Hipótesis (T-H). Generalmente el Texto está formado por una o dos frases y la Hipótesis por una frase corta como se puede ver en el par mostrado en la figura 2.5:

```
Texto: Between March and June, scientific observers say, up
to 300,000 seals are killed. In Canada, seal-hunting means
jobs, but opponents say it is vicious and endangers the
species, also threatened by global warming.

Hipótesis: Hunting endangers seal species.
```

Figura 2.5: Par Texto-Hipótesis de ejemplo.

Los pares de los conjuntos de datos son etiquetados a mano por varios evaluadores humanos eligiendo solamente aquellos pares en los cuales coincide el juicio. La anotación consiste en decidir si el texto implica a la hipótesis (y entonces se le asigna a dicho par el valor YES) o no (y entonces se le asigna a dicho par el valor NO). Una vez creados y anotados todos los pares, estos se dividen en un conjunto de desarrollo que se distribuye a los participantes para el desarrollo de sus sistemas y en un conjunto de test para evaluar la respuesta de los sistemas. En el cuadro 2.1 se puede ver el número de pares que hubo disponibles en cada edición para el desarrollo y la evaluación de los sistemas participantes.

Los conjuntos de datos están subdivididos en 4 subconjuntos (excepto los de la edición 2005 que son 7 subconjuntos) que se corresponden con ejemplos de éxito o fallo de diversas aplicaciones de lenguaje natural. En cada subconjunto se trató de balancear el número de ejemplos positivos y negativos. En algunos casos, los ejemplos se tomaron de recursos externos, como conjuntos de datos disponibles públicamente o la salida de sistemas reales, y en otros casos se obtuvieron de la Web. Los distintos tipos de aplicaciones que se tuvieron en cuenta son (entre paréntesis se indica el acrónimo que se utiliza para cada tipo de tarea):

- Recuperación de Información (IR): las hipótesis se generaron a partir de consultas a sistemas de recuperación de información que expresaban determinadas relaciones semánticas, mientras que los textos se

extrajeron de los documentos recuperados por un motor de búsqueda.

- Búsqueda de Respuestas (QA): dado un sistema real de QA, elegían primero como texto un fragmento que el sistema indicase que contenía la respuesta correcta. Para crear la hipótesis, cambiaban la pregunta a forma afirmativa insertando la respuesta dada por el sistema.
- Extracción de Información (IE): a partir de relaciones interesantes en extracción de información, los anotadores elegían como textos frases de noticias candidatas a contener la relación. Como hipótesis eligieron formulaciones sencillas de la relación.
- Resumen Automático (SUM): los anotadores recibieron clusters de documentos sobre el mismo tema y el resumen obtenido por un sistema para cada cluster. Los anotadores escogieron frases con un gran solapamiento léxico. Para los ejemplos positivos la hipótesis fue simplificada eliminando partes de la frase hasta que fuese totalmente implicada por el texto. Los ejemplos negativos fueron simplificados del mismo modo.

En la evaluación se compara la clasificación de los pares realizada por los sistemas entre los que tienen y no tienen implicación con la dada para el conjunto de test por los anotadores. Como medida de evaluación se utiliza la medida accuracy que se define como la proporción de aciertos, es decir, la fracción de respuestas correctas. Debido a que los conjuntos de test están balanceados en términos de ejemplos positivos y negativos, un sistema que diese siempre como respuesta YES (o un sistema que diese siempre NO) lograría un accuracy del 50 %, lo cuál constituye un baseline natural.

2.2.4. Sistemas participantes en el primer PASCAL RTE Challenge

En esta primera edición (Dagan et al., 2006), como era de esperar, los sistemas participantes usaron una amplia gama de inferencias que trataban el fenómeno de la Implicación Textual desde varios niveles. El tipo más básico de inferencia medía el grado de solapamiento de palabras entre el texto y la hipótesis incluyendo, posiblemente, stemming, lematización y etiquetado morfosintáctico. Los tratamientos a alto nivel de inferencia léxica consideraban relaciones entre palabras que podían reflejar implicación, basándose en métodos estadísticos o WordNet.

Otros sistemas medían el grado de solapamiento entre las estructuras sintácticas del texto y la hipótesis basándose en algún criterio de distancia. Finalmente, fueron pocos los sistemas que incorporaron alguna forma de conocimiento del mundo, y fueron unos pocos sistemas más los que usaron un verificador lógico para hacer la inferencia de implicación, normalmente sobre representaciones enriquecidas semánticamente.

Sobre los distintos tipos de conocimiento usado, se aplicaron varios métodos para tomar la decisión final de implicación entre los cuáles están modelos probabilísticos, modelos probabilísticos de traducción automática, métodos supervisados de aprendizaje, inferencia lógica y varios mecanismos específicos de puntuación.

Los resultados obtenidos por los participantes se pueden ver como los típicos para una tarea nueva de relativa dificultad. En general, los resultados de los sistemas estuvieron entre el 50 y el 60 % de accuracy, con pequeñas diferencias entre los participantes.

Un dato sorprendente fue el hecho de que la complejidad y la sofistificación de la inferencia utilizada no se correspondían plenamente con el rendimiento obtenido, dándose el caso en el que algunos de los mejores resultados se obtuvieron por sistemas que usaban un simple tratamiento léxico.

2.2.5. Sistemas participantes en el segundo PASCAL RTE Challenge

En esta edición (Bar-Haim et al., 2006), algunos de los métodos utilizados fueron: cálculo de solapamiento léxico basado en lexicones tales como WordNet y recursos adquiridos automáticamente basados en medidas estadísticas, encaje de ngramas y solapamiento de secuencias entre texto e hipótesis, cálculo de solapamiento sintáctico haciendo uso de por ejemplo algoritmos de distancia de edición de árboles, anotación semántica usando recursos como FrameNet, inferencia lógica usando demostradores lógicos, uso de conocimiento del mundo, reglas de inferencia, plantillas de paráfrasis, y adquisición (automática y manual) de corpora adicional de implicación.

Muchos de los sistemas obtuvieron medidas de similitud a diferentes niveles de análisis (léxico, sintáctico y lógico), y luego usaron dichas medidas como atributos para un clasificador que tomaba la decisión final. En general, el criterio para tomar la decisión de implicación fue la similitud entre el texto y la hipótesis, o la cobertura de la hipótesis por parte del texto (en métodos léxicos y sintácticos), además de la posibilidad de inferir la hipótesis a partir del texto (en el enfoque lógico). Hubo incluso participantes que trataron de detectar la no implicación buscando varios casos de falta de concordancia entre el texto y la hipótesis, siguiendo las observaciones de (Vanderwende and Dolan, 2005), donde se sugiere que a veces es más fácil detectar la no implicación.

En esta edición los resultados estuvieron entre el 53 y el 75 % de accuracy, encontrándose la mayoría de los sistemas en el rango de 55-61 %. Hubo dos sistemas, (Hickl et al., 2006) con 75.4 % y (Tatu et al., 2006b) con 73.8 %, que obtuvieron un valor alrededor del 10 % más que el resto de los participantes. Los resultados mostraron que los sistemas que hacían uso de técnicas de análisis profundo como solapamiento sintáctico o inferencia lógica, podían superar considerablemente a los sistemas léxicos, que conseguían

un resultado alrededor del 60 %.

La mayoría de los participantes señalaron como razones principales para el bajo rendimiento de los sistemas el tamaño de las colecciones disponibles para entrenamiento y la falta de conocimiento lingüístico y del mundo. Los sistemas que mejor trataron estos problemas fueron los que mejores resultados obtuvieron. Para ello, (Hickl et al., 2006) utilizó un corpus de implicación de gran tamaño obtenido automáticamente a partir de la Web y que contribuyó en un 10 % a los resultados obtenidos. Por otro lado, (Tatu et al., 2006b) desarrolló un sistema basado en inferencia lógica que hacía uso de conocimiento lingüístico y del mundo obtenido de varias fuentes.

2.2.6. Sistemas participantes en el tercer PASCAL RTE Challenge

En la última edición celebrada hasta el momento (Giampiccolo et al., 2007), se ha notado un cierto movimiento hacia los métodos en profundidad con una consolidación general de los métodos basados en la estructura sintáctica del texto y la hipótesis. Además, ha habido un incremento evidente del número de sistemas que hacen uso de alguna forma de inferencia lógica.

También se ha observado cómo se va haciendo un uso más cuidadoso de determinados fenómenos semánticos. Por ejemplo, en (Tatu and Moldovan, 2007) se realizo un análisis sofisticado de las entidades nombradas, distinguiendo para los nombres de personas entre el nombre y el apellido. También se han empezado a usar este año algunas formas de extracción de relaciones.

En cuanto al uso de recursos, las bases de datos léxicas (en su mayoría WordNet y DIRT) han sido usadas ampliamente. Otro tipo de recursos como Framenet, Verbnet y Propbank también han estado bastante presentes en varios sistemas.

Respecto a los resultados obtenidos por los sistemas, los valores han estado entre el 49 y el 80 % de accuracy, encontrándose la mayoría de los sistemas entre el 59 y el 66 %.

Capítulo 3

Propuesta metodológica de evaluación de sistemas de Validación de Respuestas

La experiencia obtenida a lo largo de la tarea de Búsqueda de Respuestas del CLEF muestra una falta de trabajo en la tarea de Validación de Respuestas. Tomando como ejemplo la evaluación de los sistemas de Búsqueda de Respuestas en español en el año 2005, el 73 % de las preguntas fueron contestadas correctamente por al menos un participante (Vallin et al., 2005), mientras que el mejor sistema solo contestó correctamente al 42 % de las preguntas (ver figura 3.1). Por tanto, los sistemas de Búsqueda de Respuestas necesitan de criterios para decidir si sus respuestas son o no correctas e incluso, asignarles un valor de confianza preciso. Es aquí donde surge la necesidad de tecnologías de Validación de Respuestas, ya que los sistemas de Búsqueda de Respuestas deberían ser capaces de juzgar si están devolviendo una respuesta correcta.

En este capítulo proponemos una metodología para evaluar sistemas de Validación de Respuestas. Además se muestra cómo implementamos dicha metodología y la pusimos en marcha como tarea dentro de un foro de evaluación internacional.

3.1. Validación de Respuestas

La Validación de Respuestas, en inglés Answer Validation (AV), consiste en decidir si las respuestas de un sistema de Búsqueda de Respuestas son o no correctas. Un sistema de Validación de Respuestas recibe una tripleta formada por una pregunta, una respuesta candidata y un texto soporte y devuelve un valor booleano indicando si la respuesta a la pregunta es o no correcta de acuerdo con el texto soporte.

Se espera que los sistemas de Validación de Respuestas sean útiles para

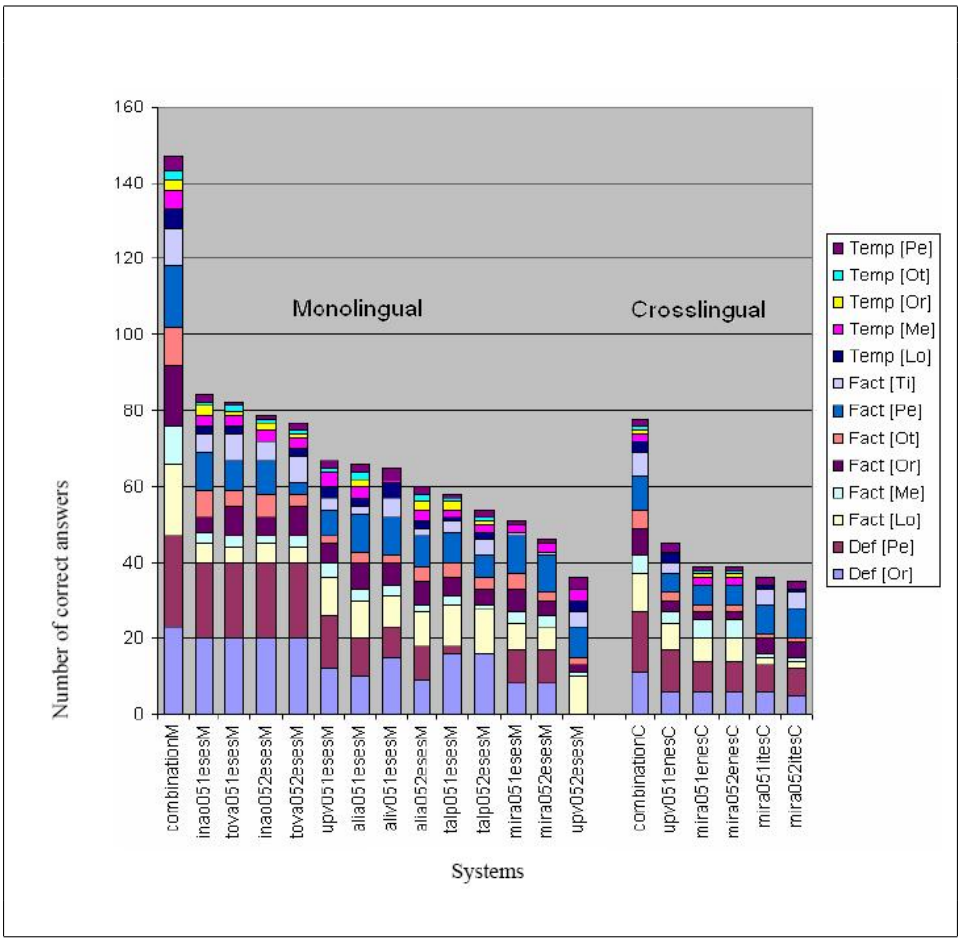

Figura 3.1: Respuestas correctas en español en la tarea de Búsqueda de Respuestas del CLEF 2005. Análisis realizado por tipo de pregunta.

mejorar el rendimiento de los sistemas de Búsqueda de Respuestas, además de mejorar la precisión de los valores de confianza de éstos. También se espera que sirvan para ayudar en la evaluación manual de las respuestas y en el desarrollo de criterios para la colaboración entre distintos sistemas.

3.2. Reformulación como un problema de Implicación Textual

El primer paso en nuestra propuesta de evaluación fue reformular la Validación de Respuestas como un problema de Implicación Textual. Para ello nos basamos en (Dagan et al., 2006), donde se indica que cuando un sistema de Búsqueda de Respuestas devuelve una respuesta y un texto que soporta la veracidad de la respuesta, entonces un sistema de RTE podría ser

aplicado para validar las respuestas después de la siguiente reformulación:

1. Construir una hipótesis combinando la pregunta más la respuesta y pasarla a forma afirmativa. Por ejemplo, para la pregunta "¿Quién es Vicente Fox?" y la respuesta "Presidente de México", una posible hipótesis sería "Vicente Fox es el presidente de México".

2. Evaluar la implicación: si el texto soporte implica a esta hipótesis, entonces cabe esperar que la respuesta sea correcta (ver ejemplos de la figura 3.2).

```
<pair id=‘‘286’’ entailment=‘‘YES’’ task=‘‘QA’’>
  <t> President Vicente Fox heads into his final year of
  office Thursday, promising a more democratic, less corrupt
  and economically stable Mexico.</t>
  <h>Vicente Fox is the President of Mexico.</h>
</pair>
<pair id=‘‘296’’ entailment=‘‘NO’’ task=‘‘QA’’>
  <t>FTAA is a huge extension of the principles behind
  NAFTA, aimed at increasing international government, not
  strengthening the United States. </t>
  <h>NAFTA is a huge extension of FTAA.</h>
</pair>
```

Figura 3.2: Ejemplos de pares texto-hipótesis obtenidos del PASCAL RTE-2.

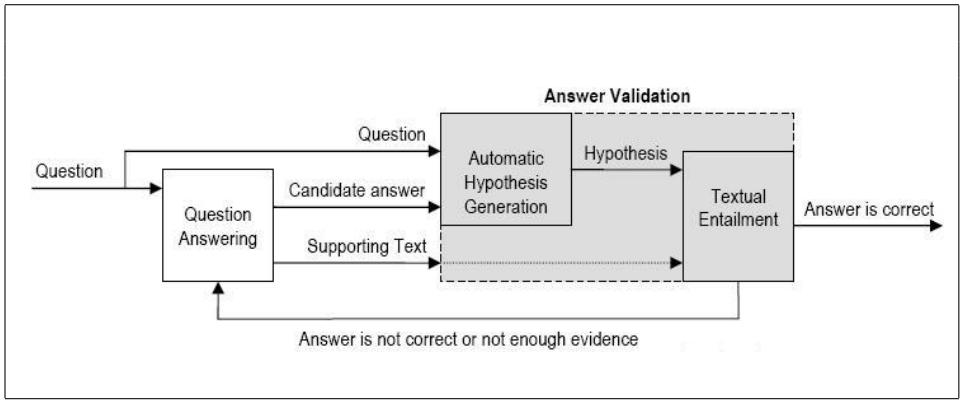

Figura 3.3: Contexto de un sistema de Validación de Respuestas dentro de un sistema de Búsqueda de Respuestas.

Con este enfoque, los sistemas de RTE deben tener también en cuenta el problema de la generación automática de hipótesis a partir de las preguntas

y las respuestas candidatas (ver figura 3.3). Con el fin de que nuestra propuesta se asemejase a la tarea del PASCAL RTE Challenge, decidimos omitir este problema, de tal modo que los sistemas de Validación de Respuestas recibirían una hipótesis ya construida.

3.3. Desarrollo de colecciones de evaluación

El siguiente paso consistió en desarrollar el método con el que generar colecciones que sirviesen como entrada a los sistemas de Validación de Respuestas basados en implicación textual, a partir de la salida de los sistemas reales de Búsqueda de Respuestas.

Consideramos que estas colecciones debían estar formadas por un conjunto de pares texto-hipótesis similares a los del PASCAL RTE Challenge, de tal modo que cada par está asociado a la respuesta de un sistema de Búsqueda de Respuestas. El texto del par sería el texto soporte dado por el sistema y la hipótesis se generaría a partir de la pregunta y la respuesta candidata basándose en la propuesta del PASCAL RTE Challenge. Los posibles juicios para cada par, siguiendo la metodología del PASCAL RTE Challenge, serían YES si se considera que la respuesta es correcta y soportada por el texto dado, o NO en caso contrario.

A continuación se muestra la metodología que propusimos para crear las hipótesis, junto con el método para otorgar un juicio a cada par a partir de las evaluaciones manuales realizadas a los sistemas de Búsqueda de Respuestas.

3.3.1. Construcción de la hipótesis

Como la Implicación Textual está definida entre frases afirmativas, el primer paso consiste en transformar la pregunta a forma afirmativa. Por ejemplo, la pregunta 1 sería transformada manualmente al patrón 2.

1. ¿Cuál es la capital de Croacia?
2. <**respuesta/**> es la capital de Croacia

donde la marca <**respuesta/**> es instanciada por cualquier respuesta dada a esa pregunta por cualquier sistema. A través de estos patrones se generarían todas las hipótesis de forma automática sustituyendo la marca <**respuesta/**> por la correspondiente respuesta dada por el sistema de Búsqueda de Respuestas. Por ejemplo, para la respuesta *Zagreb*, la instanciación del patrón daría lugar a la hipótesis *Zagreb es la capital de Croacia.*

Nos dimos cuenta de que usando este método algunas respuestas incorrectas o inexactas podrían darle a la hipótesis no solo una semántica incorrecta, sino también una estructura sintáctica incorrecta. Por ejemplo, para la respuesta *Zagreb fue entonces vista como el centro político*, la instanciación del patrón daría lugar a la hipótesis *Zagreb fue entonces vista como*

el centro político es la capital de Croacia. Consideramos que como el objetivo final es el desarrollo de sistemas reales de Validación de Respuestas, ésta es una característica deseable para las colecciones, permitiendo el desarrollo de criterios sintácticos para detectar respuestas incorrectas y promoviendo también el desarrollo de sistemas robustos a errores.

Por último, decidimos que las respuestas NIL debían ser eliminadas ya que estas respuestas pueden ser o no ser correctas, pero en cualquier caso, NIL indica que no hay respuesta que validar. También tomamos la decisión de que se debían eliminar los casos donde la respuesta y el texto soporte están repetidos con el fin de que las colecciones no contengan pares repetidos.

3.3.2. Determinando el valor de implicación

Debido a que los juicios que se emiten para los sistemas de Búsqueda de Respuestas no son binarios, es necesario convertir dichos juicios en valores de implicación (YES o NO). Nosotros propusimos la siguiente conversión:

- Respuestas correctas: en este caso la respuesta a la pregunta es correcta y el texto soporte apoya su veracidad. Por tanto el texto implica la hipótesis y el valor de implicación es YES.

- Respuestas no soportadas: en este caso aunque la respuesta podría ser considerada como correcta, el texto soporte no apoya su veracidad. Por ello, el texto no puede implicar a la hipótesis y el valor de implicación es NO.

- Respuestas inexactas: este es un caso difícil incluso para los evaluadores humanos de búsqueda de respuestas. En este caso no se tienen criterios claros para determinar el valor de implicación sin la intervención humana. Como generalmente son pocos los casos, se optó por ignorarlos.

- Respuestas incorrectas: en este caso la respuesta a la pregunta no es válida. Aunque la respuesta puede estar extraída directamente del texto, la reformulación de la pregunta y la respuesta como una frase (la hipótesis), evita el que haya implicación entre el texto y la hipótesis. Por tanto, se le da el valor de implicación NO.

3.4. Medidas de evaluación

En lugar de usar una medida global de precisión como medida de evaluación, consideramos centrarnos solamente en las respuestas correctas, es decir, en los pares con implicación. Tuvimos dos razones para tomar esta decisión:

- Una respuesta será validada si se tiene suficiente evidencia para afirmar que es cierta. En los casos donde no se tenga dicha evidencia, el sistema de validación pedirá otra respuesta candidata. De este modo, la Validación de Respuestas debería enfocarse en la detección de que hay suficiente evidencia de que las respuestas son correctas.
- En un entorno real no hay un equilibrio entre las respuestas candidatas correctas e incorrectas, o lo que es lo mismo, un sistema de validación no recibe respuestas correctas e incorrectas en la misma proporción. De hecho, las experiencias en el CLEF durante los años anteriores mostraron que solamente el 23 % de las respuestas dadas por los sistemas (la media de todos los sistemas) eran correctas. Aunque la proporción puede cambiar, lo importante es que la evaluación de módulos de Validación de Respuestas debe considerar la salida real de los sistemas de Búsqueda de Respuestas, la cuál no está balanceada. Si se considerase la precisión sobre todos los pares, entonces un sistema baseline que siempre devolviese NO obtendría una precisión de 0.7, lo que parece ser un valor demasiado alto para los propósitos de evaluación.

Por tanto propusimos el uso de precisión (3.1), cobertura (3.2) y medida F (3.3) sobre pares con valor de implicación YES. En otras palabras, propusimos medir la habilidad del sistema para detectar pares con implicación o de detectar si hay suficiente evidencia para aceptar una respuesta.

$$precision = \frac{|predichas\ como\ YES\ correctamente|}{|predichas\ como\ YES|} \tag{3.1}$$

$$cobertura = \frac{|predichas\ como\ YES\ correctamente|}{|pares\ YES|} \tag{3.2}$$

$$F = \frac{2 * cobertura * precision}{cobertura + precision} \tag{3.3}$$

3.5. Activación de una tarea internacional de evaluación competitiva

3.5.1. Definición AVE 2006

Con el propósito de poner en práctica la metodología de evaluación propuesta, desde la UNED lanzamos el primer Answer Validation Exercise (AVE 2006) (Peñas et al., 2007). El AVE 2006 se lanzó dentro del marco del CLEF con el objetivo de promover el desarrollo y evaluación de sistemas de Validación de Respuestas. En el AVE 2006 seguimos la metodología expuesta anteriormente para siete idiomas usando la salida de sistemas reales que participaron ese mismo año en la tarea de Búsqueda de Respuestas del CLEF.

El AVE 2006 está conectado a la tarea de Búsqueda de Respuestas como se puede ver en la figura 3.4. De acuerdo con lo expuesto arriba, se decidió excluir la generación automática de hipótesis y dar las hipótesis ya construidas a los participantes. De este modo, el AVE 2006 se centró en la evaluación de sistemas de Validación de Respuestas que reciben como entrada la tupla [Hipótesis formada a partir de Pregunta y Respuesta, Texto Soporte] y devuelven un valor [YES|NO] indicando si el Texto Soporte implica a la Hipótesis, o lo que es lo mismo, si la Respuesta a la Pregunta es correcta o no de acuerdo al Texto Soporte.

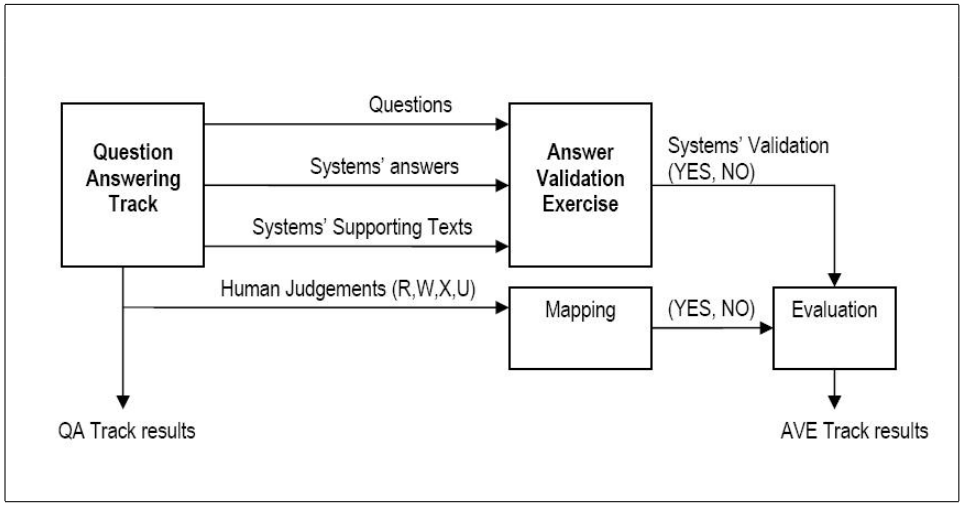

Figura 3.4: Relación entre la tarea de Búsqueda de Respuestas y el AVE 2006.

3.5.2. Colecciones de test AVE 2006

Para generar las colecciones de test del AVE 2006 se siguió la metodología expuesta en 3.3 tomando como entrada las preguntas, respuestas y textos soportes de los sistemas participantes en la tarea de Búsqueda de Respuestas del CLEF 2006. Las colecciones de evaluación se generaron para siete idiomas: español, inglés, alemán, portugués, francés, italiano y holandés.

Los cuadros 3.1 y 3.2 muestran el número de pares de las colecciones de test para cada idioma que se obtuvieron como resultado de la metodología descrita. Los pares con valor UNKNOWN se obtuvieron a partir de las respuestas evaluadas como inexactas o de las respuestas que no fueron evaluadas en la tarea de Búsqueda de Respuestas. Se decidió que estos pares se ignorarían a la hora de calcular el rendimiento de los sistemas.

El porcentaje de pares YES, NO y UNKNOWN es similar en todos los idiomas menos en el caso del porcentaje de pares UNKNOWN en inglés y portugués. En estos idiomas, hubo hasta 5 runs de la tarea de Búsqueda de

Cuadro 3.1: Pares YES, NO y UNKNOWN en las colecciones de test del AVE 2006

	Alemán	**Inglés**	**Español**	**Francés**
Pares YES	344(24 %)	198(9.5 %)	671(28 %)	705(22 %)
Pares NO	1064(74 %)	1048(50 %)	1615(68 %)	2359(72 %)
UNKNOWN	35(3 %)	842(40.5 %)	83(4 %)	202(6 %)
Total	1443	2088	2369	3266

Cuadro 3.2: Pares YES, NO y UNKNOWN en las colecciones de test del AVE 2006

	Italiano	**Holandés**	**Portugués**
Pares YES	187(16 %)	81(10 %)	188(14 %)
Pares NO	901(79 %)	696(86 %)	604(46 %)
UNKNOWN	52(5 %)	30(4 %)	532(40 %)
Total	1140	807	1324

Respuestas que no fueron evaluados.

Todas estas colecciones están disponibles en http://nlp.uned.es/QA/ave para investigadores registrados en el CLEF.

3.5.3. Resultados AVE 2006

En el AVE 2006 participaron once grupos con 38 runs en siete idiomas distintos. El cuadro 3.3 muestra a los grupos participantes y el número de runs que cada uno de ellos envió por cada idioma. En cada idioma participaron al menos dos grupos distintos, de tal modo que la comparación entre distintos enfoques es posible. Inglés y español fueron los idiomas con mayor participación con 11 y 9 runs respectivamente.

En las tablas 3.4-3.10 se muestran los resultados de los sistemas participantes en cada idioma. Además también se muestran los resultados de dos sistemas baseline: un sistema que acepta todas las respuestas (devuelve YES en el 100 % de los pares) y otro sistema que devolvería YES en el 50 % de los pares. Los resultados entre distintos idiomas no se pueden comparar debido a que el número de pares y la proporción de pares YES es distinto para cada idioma (por los envíos realizados por los sistemas de Búsqueda de Respuestas del CLEF).

Las técnicas más utilizadas por los participantes fueron las de aprendizaje automático y medidas de solapamiento entre texto e hipótesis. Los sistemas que hicieron uso de lógica mostraron un gran rendimiento. Al menos uno de ellos (COGEX de LCC) utilizó grandes cantidades de conocimiento que, de acuerdo a las conclusiones del RTE-2, parece ser un factor crítico

Cuadro 3.3: Participantes y runs por cada idioma del AVE 2006.

	Alemán	**Inglés**	**Español**	**Francés**	**Italiano**	**Holandés**	**Portugués**	**Total**
Fernuniversität in Hagen (FUH)	2							2
Language Computer Corporation (LCC)		1	1					2
U. Rome "Tor Vergata"		2						2
U. Alicante (Kozareva)	2	2	2	2	2	2	1	13
U. Politecnica de Valencia		1						1
U. Alicante (Ferrández)		2						2
LIMSI-CNRS				1				1
U. Twente	1	2	2	1	1	2	1	10
UNED (Herrera)			2					2
UNED (Rodrigo)			1					1
ITC-irst		1						1
R2D2 project			1					1
Total	5	11	9	4	3	4	2	38

Cuadro 3.4: Resultados para inglés en el AVE 2006.

Sistema	**F**	**Precisión**	**Cobertura**
LCC	0.4559	0.3261	0.7576
U. Rome_2	0.4106	0.2838	0.7424
itc-irst	0.3919	0.3090	0.5354
U. Rome_1	0.3780	0.2707	0.6263
Kozareva_2	0.3720	0.2487	0.7374
Ferrández_2	0.3177	0.2040	0.7172
Kozareva_1	0.3174	0.2114	0.6364
Ferrández_1	0.3070	0.2144	0.5404
utwente.ta	0.3022	0.3313	0.2778
utwente.lcs	0.2759	0.2692	0.2828
100 % YES Baseline	0.2742	0.1589	1
50 % YES Baseline	0.2412	0.1589	0.5
Valencia	0.075	0.2143	0.0455

Cuadro 3.5: Resultados para francés en el AVE 2006.

Sistema	**F**	**Precisión**	**Cobertura**
Kozareva_2	0.4693	0.3444	0.7362
Kozareva_1	0.4085	0.3836	0.4369
100 % YES Baseline	0.3741	0.2301	1
50 % YES Baseline	0.3152	0.2301	0.5
LIMSI-CNRS	0.1112	0.4327	0.0638
utwente.lcs	0.0943	0.4625	0.0525

Cuadro 3.6: Resultados para español en el AVE 2006.

Sistema	**F**	**Precisión**	**Cobertura**
LCC	0.6063	0.527	0.7139
Herrera_1	0.5655	0.467	0.7168
Herrera_2	0.5615	0.4652	0.7079
Rodrigo	0.5315	0.4364	0.6796
Kozareva_2	0.5301	0.4065	0.7615
R2D2	0.4938	0.4387	0.5648
utwente.ta	0.4682	0.4811	0.4560
100 % YES Baseline	0.4538	0.2935	1
utwente.lcs	0.4326	0.5507	0.3562
Kozareva_1	0.4303	0.4748	0.3934
50 % YES Baseline	0.3699	0.2935	0.5

Cuadro 3.7: Resultados para alemán en el AVE 2006.

Sistema	**F**	**Precisión**	**Cobertura**
FUH_1	0.5420	0.5839	0.5058
FUH_2	0.5029	0.7293	0.3837
Kozareva_2	0.4685	0.3573	0.6802
100 % YES Baseline	0.3927	0.2443	1
Kozareva_1	0.3874	0.4006	0.375
50 % YES Baseline	0.3282	0.2443	0.5
utwente.lcs	0.1432	0.4	0.0872

Cuadro 3.8: Resultados para holandés en el AVE 2006.

Sistema	**F**	**Precisión**	**Cobertura**
utwente.ta	0.3871	0.2874	0.5926
Kozareva_1	0.2957	0.189	0.6790
Kozareva_2	0.2548	0.1484	0.9012
utwente.lcs	0.2201	0.2	0.2469
100 % YES Baseline	0.1887	0.1042	1
50 % YES Baseline	0.1725	0.1042	0.5

Cuadro 3.9: Resultados para portugués en el AVE 2006.

Sistema	**F**	**Precisión**	**Cobertura**
100 % YES Baseline	0.3837	0.2374	1
utwente.lcs	0.3542	0.5783	0.2553
50 % YES Baseline	0.3219	0.2374	0.5
Kozareva	0.1529	0.1904	0.1277

Cuadro 3.10: Resultados para italiano en el AVE 2006.

Sistema	**F**	**Precisión**	**Cobertura**
Kozareva_2	0.4066	0.2830	0.7219
Kozareva_1	0.3480	0.2164	0.8877
100 % YES Baseline	0.2934	0.1719	1
50 % YES Baseline	0.2558	0.1719	0.5
utwente.lcs	0.1673	0.3281	0.1123

para el éxito de los sistemas.

3.6. Conclusiones

El punto de partida de la propuesta para evaluar sistemas de Validación de respuestas fue la reformulación de la Validación de Respuestas como un problema de reconocimiento de Implicación Textual bajo la suposición de que la hipótesis puede ser automáticamente obtenida instanciando patrones de hipótesis con las respuestas de los sistemas de Búsqueda de Respuestas. De este modo, las colecciones desarrolladas según la metodología expuesta están orientadas al desarrollo y evaluación de los sistemas de Validación de Respuestas que hacen uso de Implicación Textual. También se ha propuesto una metodología para la evaluación de este tipo de sistemas.

Con el fin de poner en práctica la metodología propuesta, se celebró el AVE 2006 dentro del marco del CLEF. En esta tarea se evaluaron sistemas en siete idiomas distintos y la participación en esta tarea muestra un interés en la investigación en este campo.

Capítulo 4

Experimentos realizados en Reconocimiento de Implicación Textual

En este capítulo se realiza la propuesta de un sistema de RTE realizado por el autor haciendo uso del reconocimiento de entidades (entidades nombradas, expresiones numéricas y expresiones temporales), se muestran los resultados obtenidos por el sistema en el RTE-3 (Giampiccolo et al., 2007) y se comentan las conclusiones que se han obtenido a tenor de los resultados.

4.1. Propuesta de participación en el tercer RTE Challenge

Debido a que un gran porcentaje de los pares de los PASCAL RTE Challenges contienen entidades (el 82.6 % de las hipótesis de la colección de test del RTE-3 contenían al menos una entidad), uno de los objetivos de este trabajo fue estudiar el efecto del reconocimiento de las entidades sobre la Implicación Textual.

Las técnicas que se utilizan en los experimentos realizados para alcanzar este objetivo fueron:

- Solapamiento léxico entre n-gramas de texto e hipótesis
- Implicación entre entidades
- Solapamiento entre las ramas de los árboles de dependencias de texto e hipótesis

4.1.1. Procesamiento lingüístico

Los sistemas presentados están basados en técnicas superficiales de análisis léxico y análisis sintáctico considerando por separado cada tarea (extracción de información, recuperación de información, sistemas de Búsqueda de Respuestas y resumen automático) del RTE. Los sistemas reciben como entrada pares de fragmentos de texto (un texto y una hipótesis) y devuelven como salida un valor booleano: YES si se considera que el texto implica a la hipótesis y NO en caso contrario. Dicho valor se obtiene como salida de un clasificador SVM.

El primer paso consiste en procesar cada par texto-hipótesis para obtener la siguiente información necesaria a la hora de tomar la decisión de implicación:

- POS: se hace uso del etiquetador morfosintáctico Freeling (Carreras et al., 2004) para obtener los lemas de texto e hipótesis.
- NER: también se hace uso del reconocedor de entidades nombradas de Freeling para obtener la información necesaria para el módulo de implicación de entidades que se describe en la siguiente sección. Se etiquetan las expresiones numéricas, los nombres propios y las expresiones temporales.
- Análisis de dependencias: haciendo uso de Minipar (Lin, 1998) se obtienen los árboles de dependencias de texto e hipótesis.

4.1.2. Implicación entre entidades

Una vez se han detectado las entidades de texto e hipótesis, el siguiente paso consiste en determinar las relaciones de implicación existentes entre las entidades del texto y las entidades de la hipótesis. En (Rodrigo et al., 2007) definimos las siguientes relaciones de implicación entre entidades:

1. Un nombre propio NP1 implica a un nombre propio NP2 si la cadena de texto de NP1 contiene a la cadena de texto de NP2.
2. Una expresión temporal T1 implica a una expresión temporal T2 si el rango de tiempo expresado por T1 está incluido en el rango de tiempo de T2.
3. Una expresión numérica N1 implica a una expresión numérica N2 si el rango asociado a N2 contiene al rango expresado por N1.

En determinadas ocasiones hay caracteres que cambian en diferentes expresiones de una misma entidad, como por ejemplo, en un nombre propio con distintas grafías (ej: Yasser, Yaser, Yasir). Con el fin de detectar implicación en estas situaciones, para cuando el proceso anteriormente descrito

falla, se implementó una modificación del proceso de implicación haciendo uso de la distancia de edición de Levenshtein (Levensthein, 1966). De este modo, si dos entidades difieren en menos de un 20 %, se asume que existe una relación de implicación entre dichas entidades.

Sin embargo, la definición de implicación entre entidades que se ha dado no es robusta frente a errores en la clasificación de entidades como se puede ver en la figura 4.1. En el ejemplo de la figura, la expresión 1990 representa un año pero en la hipótesis es reconocida como una expresión numérica. Sin embargo, la misma expresión es reconocida como una expresión temporal en el texto y por tanto, la expresión numérica de la hipótesis no puede ser implicada por ella según la definición dada arriba de implicación entre entidades.

```
<t>Iraq invaded Kuwait on <TIMEX>August_2_1990</TIMEX></t>
<h>Iraq invaded Kuwait in <NUMEX>1990</NUMEX></h>
```

Figura 4.1: Ejemplo de un error en la clasificación de entidades.

Se realizó un cálculo del efecto de dichos errores en la clasificación de entidades haciendo uso de las dos configuraciones siguientes:

1. Un sistema basado en análisis de dependencias y WordNet (Herrera et al., 2006a) que hace uso de la clasificación de entidades dada por el reconocedor de entidades nombradas, y donde las relaciones de implicación entre entidades son las descritas anteriormente.

2. El mismo sistema que el anterior basado en análisis de dependencias y WordNet, pero sin hacer uso de la clasificación del reconocedor de entidades nombradas. En esta configuración todas las entidades detectadas reciben la misma etiqueta y se considera que una entidad E1 implica a una entidad E2 si la cadena de texto de E1 contiene a la cadena de texto de E2 (ver figura 4.2).

```
<t>...Chernobyl accident began on
      <ENTITY>Saturday_April_26_1986</ENTITY>...</t>
<h>The Chernobyl disaster was in <ENTITY>1986</ENTITY></h>
```

Figura 4.2: Ejemplo de un par texto-hipótesis que justifica el procesamiento de implicación.

Se comparó el rendimiento de estas dos configuraciones sobre el conjunto de test del RTE-2. Los resultados obtenidos, usando la medida accuracy, se

muestran en el cuadro 4.1. Los resultados del cuadro muestran que usando un método más sencillo y robusto (sin usar la clasificación dada por el reconocedor de entidades nombradas) no solo se obtiene el mismo rendimiento, sino que incluso se mejora ligeramente.

Cuadro 4.1: Comparación entre distintos métodos de implicación de entidades.

	Accuracy
Configuración 1	0.610
Configuración 2	0.614

Este hecho hizo que se tomase la decisión de ignorar la categorización del reconocedor de entidades nombradas y se asumiese que tanto texto como hipótesis son textos relacionados donde las mismas expresiones deben recibir las mismas categorías, sin la necesidad de clasificarlas. Por tanto, todas las entidades detectadas reciben la misma etiqueta y se considera que una entidad E1 implica a una entidad E2 si la cadena de texto de E1 contiene a la cadena de texto de E2.

4.1.3. Solapamiento a nivel de frase

En este punto se hace uso de un módulo de solapamiento de árboles que busca ramas solapantes dentro del árbol de dependencias de la hipótesis. Hay una posible rama solapante por cada hoja del árbol. Se considera que una rama del árbol de la hipótesis es solapante si todos sus nodos desde la raíz hasta la hoja están implicados léxicamente (Herrera et al., 2006b). De este modo, el subárbol formado por todas las ramas solapantes del árbol de la hipótesis está incluido en el correspondiente árbol de dependencias del texto, dando una idea de inclusión de árboles.

Se asume que con que mayor sea el subárbol de la hipótesis incluido en el del texto, mayor similaridad semántica habrá entre texto e hipótesis. De este modo, la existencia o no de implicación entre texto e hipótesis considera la porción del árbol de dependencias de la hipótesis que está incluido en el árbol de dependencias del texto.

4.1.4. Decisión de implicación

Para tomar la decisión final de implicación se hace uso de un clasificador SVM entrenado con un conjunto de atributos obtenidos a partir del procesamiento descrito arriba. Los atributos y las estrategias de entrenamiento empleadas se describen a continuación:

Se dispone de los siguientes atributos para ser empleados en el clasificador SVM:

1. Porcentaje de nodos del árbol de dependencias de la hipótesis que pertenecen a ramas solapantes de acuerdo a lo dicho arriba considerando, respectivamente:
   - Implicación léxica entre las palabras de los fragmentos de texto involucrados.
   - Implicación léxica entre los lemas de los fragmentos de texto involucrados.
2. Porcentaje de palabras de la hipótesis que están en el texto(tratadas como bolsas de palabras).
3. Porcentaje de unigramas (lemas) de la hipótesis que están en el texto (tratadas como bolsas de lemas).
4. Porcentaje de bigramas (lemas) de la hipótesis que están en el texto (tratadas como bolsas de lemas).
5. Porcentaje de trigramas (lemas) de la hipótesis que están en el texto (tratadas como bolsas de lemas).
6. Un valor booleano indicando si hay o no alguna entidad en la hipótesis que no esté implicada por una o más entidades del texto según la decisión de implicación entre entidades descrita arriba.

Cuadro 4.2: Experimentos usando entrenamiento por separado sobre la colección de desarrollo mediante validación cruzada.

	Accuracy usando el mismo modelo para todas las tareas	Accuracy con un modelo distinto por cada tarea
Configuración 1	0.64	0.67
Configuración 2	0.62	0.66

Cuadro 4.3: Experimentos haciendo uso de entrenamiento por separado sobre la colección de test.

	Accuracy usando el mismo modelo para todas las tareas	Accuracy con un modelo distinto por cada tarea
Configuración 1	0.59	0.62
Configuración 2	0.60	0.64

Sobre la decisión de cómo realizar el entrenamiento de los clasificadores SVM, se quería estudiar el resultado de entrenar un único modelo comparado con el resultado de entrenar un modelo distinto por cada tarea.

Con el fin de realizar este estudio, se realizó un experimento haciendo uso de las siguientes dos configuraciones:

1. Un modelo SVM que hace uso de los atributos 2, 3, 4 y 5 descritos arriba.
2. Un modelo SVM que hace uso de los atributos 2, 3, 4, 5 y 6 descritos arriba.

Cada configuración fue entrenada usando validación cruzada sobre la colección de desarrollo del RTE-3 de dos formas distintas:

1. Entrenando un modelo para todos los pares.
2. Entrenando un modelo distinto para cada tarea. Cada modelo fue entrenado usando solamente los pares de la tarea que iba a predecir dicho modelo.

Los resultados obtenidos en los experimentos se pueden ver en el cuadro 4.2, donde se puede comprobar que entrenando un modelo distinto por cada tarea los resultados son ligeramente mejores, incrementando el rendimiento de ambas configuraciones. Teniendo en cuenta estos resultados, se tomó la decisión de usar un entrenamiento distinto por cada tarea en los runs enviados.

La decisión fue confirmada después del envío de runs al RTE-3 con nuevos experimentos sobre la colección de test del RTE-3 usando como entrenamiento la colección de desarrollo del RTE-3 (ver resultados en el cuadro 4.3).

4.1.5. Runs enviados

Dos runs fueron enviados al RTE-3. Cada run fue entrenado utilizando el método descrito anteriormente haciendo uso de los siguientes atributos descritos arriba:

- El run 1 se obtuvo usando los atributos 2, 3, 4 y 5 descritos arriba. Estos atributos obtuvieron buenos resultados en Implicación Textual sobre sistemas de pregunta respuesta, como se puede ver en el capítulo 5, y se quería comprobar su rendimiento sobre otras tareas. Además, con este conjunto de atributos se habían obtenido buenos resultados sobre la colección de desarrollo.
- El run 2 se obtuvo utilizando para cada tarea un conjunto distinto de atributos. El objetivo era comprobar si se podían obtener mejores resultados combinando distintos sistemas. Por ello, se hizo uso de los siguientes atributos para cada tarea:

- IE: atributos 2, 3, 4, 5 y 6 descritos arriba. Se eligió este conjunto de atributos puesto que con ellos se habían obtenido los mejores resultados para esta tarea en la colección de desarrollo.
- IR: atributos 2, 3, 4 y 5 descritos arriba. Al igual que en la tarea anterior, la motivación para elegir estos atributos fue que con ellos se obtuvieron los mejores resultados sobre la colección de desarrollo.
- QA: atributo 6 descrito arriba. Se hizo uso de este atributo, con el que se había obtenido un valor de accuracy superior al 70 % sobre los pares de QA en los experimentos realizados sobre la colección de desarrollo, para estudiar el efecto de las entidades en los pares de QA.
- SUM: atributos 1, 2 y 3 descritos arriba. Se eligieron estos atributos para mostrar la importancia del análisis de dependencias en los pares SUM como se explica en la sección 4.2.

4.2. Resultados

En el cuadro 4.4 se muestran los resultados obtenidos sobre el conjunto de test por los dos runs enviados y que se han descrito arriba. El accuracy fue la principal medida que se aplicó a los sistemas participantes.

Como se puede ver en ambos runs, se obtienen diferentes valores en cada tarea. El mejor resultado se obtiene para los pares de QA con un valor de accuracy del 72 % en ambos runs a pesar de que se usan métodos distintos. Estos resultados nos animan a usar este sistema para la Validación de Respuestas. En el run 2, donde se usan distintas configuraciones para cada tarea, se obtienen resultados ligeramente mejores, aunque solo en IE y SUM. Sin embargo, los resultados son demasiado semejantes como para aceptar nuestra intuición inicial de que los pares de diferentes tareas podrían necesitar no solo un entrenamiento diferente, sino también el uso de diferentes enfoques para tomar la decisión de implicación.

Cuadro 4.4: Resultados del run 1 y del run 2

	Accuracy	
	run 1	run 2
IE	52.50 %	53.50 %
IR	67 %	67 %
QA	72 %	72 %
SUM	58 %	60 %
Global	62.38 %	63.12 %

4.3. Discusión

En el run 2 se usó el procesamiento de entidades para IE y QA, las dos tareas con el mayor porcentaje de pares con al menos una entidad en la hipótesis (98.5 % en IE y 97 % en QA).

El trabajo realizado previamente sobre el uso de entidades en Implicación Textual (ver capítulo 5) sugería que el uso de información sobre entidades permitiría obtener buenos resultados. Sin embargo, después de la experiencia del RTE-3, se descubrió que el uso de entidades no mejoraba notablemente el resultado en todas las tareas, sino solamente en QA.

Se realizó un estudio cualitativo sobre los pares de IE observando que, como cabía esperar, en dichos pares las relaciones entre entidades eran más importantes que las entidades.

La figura 4.3 muestra algunos ejemplos donde todas las entidades están implicadas pero no las relaciones entre ellas. En el par 5, tanto *Michael Laski* como *China* están implicadas, pero la relación entre ellas es *took the side of* en el texto, y *was an opponent of* en la hipótesis. El mismo problema aparece en los otros pares con la relación *left* en lugar de *was born in* (par 8) o el uso de pasiva en lugar de activa (par 7).

Comparando los resultados del run 1 y el run 2, el análisis de dependencias muestra su utilidad en los pares SUM, donde los textos y las hipótesis tienen un mayor paralelismo sintáctico que en los pares de las otras tareas. Este hecho se muestra en el cuadro 4.5, donde el porcentaje de nodos del árbol de dependencias de la hipótesis que pertenecen a ramas solapantes es mucho mayor en los pares SUM que en el resto de los pares.

Cuadro 4.5: Porcentaje de nodos de la hipótesis en ramas solapantes.

	Porcentaje
SUM	75,505 %
IE	7,353 %
IR	6,422 %
QA	8,496 %

Este paralelismo sintáctico parece ser el responsable del incremento del 2 % entre el primer y el segundo run en pares SUM.

4.4. Conclusiones

Los experimentos de este trabajo han estado centrados en el estudio de la importancia de considerar implicación entre entidades para el reconocimiento de Implicación Textual y en el uso de un entrenamiento distinto por cada tarea. Como se ha podido ver, ambos enfoques han incrementado ligeramente

```
<pair id=‘‘5’’ entailment=‘‘NO’’ task=‘‘IE’’>
  <t>The Communist Party USA was a small Maoist political
  party which was founded in 1965 by members of the Communist
  Party around Michael Laski who took the side of China in the
  Sino-Soviet split.</t>
  <h>Michael Laski was an opponent of China.</h>
</pair>

<pair id=‘‘7’’ entailment=‘‘NO’’ task=‘‘IE’’>
  <t>Sandra Goudie was first elected to Parliament in the
  2002 elections, narrowly winning the seat of Coromandel by
  defeating Labour candidate Max Purnell and pushing incumbent
  Green MP Jeanette Fitzsimons into third place.</t>
  <h>Sandra Goudie was defeated by Max Purnell.</h>
</pair>

<pair id=‘‘8’’ entailment=‘‘NO’’ task=‘‘IE’’>
  <t>Ms. Minton left Australia in 1961 to pursue her
  studies in London.</t>
  <h>Ms. Minton was born in Australia.</h>
</pair>
```

Figura 4.3: Pares de IE con implicación entre las entidades pero no entre las relaciones de las entidades.

el rendimiento de los sistemas propuestos. Además, también se ha podido comprobar que el uso de distintos enfoques para cada tarea incrementa el rendimiento del sistema.

Capítulo 5

Experimentos realizados en validación de respuestas

En este capítulo se muestran los experimentos llevados a cabo en el marco del AVE 2006.

5.1. Propuesta

En la tarea de búsqueda de respuestas del CLEF (que es de donde se generan las colecciones del AVE) las preguntas y respuestas contienen una gran cantidad de entidades (como nombres de personas, organizaciones, localidades, números, fechas, etc) debido a la naturaleza de las preguntas en esta tarea: el 75 % de las preguntas en español fueron del tipo factual en los años anteriores (Vallin et al., 2005). Por este motivo se estudió la posibilidad de usar una combinación de una relación de implicación entre entidades con un clasificador SVM para solucionar el problema de Validación de Respuestas.

5.2. Participación en AVE 2006

El sistema desarrollado para participar en el AVE 2006 en español se basó en los sistemas desarrollados para el RTE-1 (Herrera et al., 2006b) y el RTE-2 (Herrera et al., 2006a). Sin embargo, estos dos sistemas estaban diseñados para inglés y por tanto el sistema del AVE 2006 fue diseñado y desarrollado de acuerdo a los recursos disponibles para español, sin poder usar algunos subsistemas implementados para inglés como el analizador de dependencias.

El sistema propuesto está basado en lematización. El sistema acepta pares de fragmentos de texto (pares texto-hipótesis) como entrada y devuelve un valor booleano: YES si el texto implica a la hipótesis y NO en caso

contrario. Este valor se obtiene aplicando un clasificador SVM previamente entrenado. Los componentes del sistema son los siguientes:

- Procesamiento lingüístico: usando Freeling (Carreras et al., 2004) se obtienen los lemas de los pares texto-hipótesis.
- Solapamiento a nivel de frase: se usa un módulo de solapamiento de texto que calcula respectivamente el porcentaje de palabras, unigramas (lemas), bigramas (lemas) y trigramas (lemas) de la hipótesis que son implicados por unidades léxicas (palabras o ngramas) del texto, considerándolos como bolsas de unidades léxicas.
- Decisión de implicación: se hace uso de un clasificador SVM entrenado sobre la colección de desarrollo y aplicado a la colección de test.

5.2.1. Experimentos

Con estos componentes se crearon varios sistemas y se combinaron y evaluaron para ver su comportamiento. Los sistemas creados se describen a continuación:

Clasificador SVM

El clasificador SVM fue entrenado con el siguiente conjunto de atributos obtenido del módulo de solapamiento a nivel de frase para cada par texto-hipótesis:

1. Porcentaje de palabras de la hipótesis presentes en el texto (tratadas como bolsa de palabras).
2. Porcentaje de unigramas (lemas) de la hipótesis presentes en el texto (tratados como bolsa de palabras).
3. Porcentaje de bigramas (lemas) de la hipótesis presentes en el texto (tratados como bolsa de palabras).
4. Porcentaje de trigramas (lemas) de la hipótesis presentes en el texto (tratados como bolsa de palabras).

El primer experimento evaluó el resultado obtenido por este sistema baseline.

Decisión de implicación basada solamente en entidades

La tesis de la que se partió fue que en el reconocimiento de implicación textual todos los elementos de la hipótesis deben estar implicados por elementos del texto. En especial, todas las entidades de la hipótesis deben estar

implicadas por entidades del texto. Por tanto, el sistema asume que si hay una entidad en la hipótesis que no está implicada (de acuerdo a la definición de implicación entre entidades dada en el capítulo 4) por una o más entidades del texto, entonces la respuesta no está soportada y el sistema debe devolver el valor NO para el par.

Sin embargo, en los pares donde todas las entidades de la hipótesis están implicadas, no hay evidencias suficientes para decidir si el valor del par es YES o NO.

En este experimento se decidió evaluar los resultados obtenidos cuando el valor por defecto es siempre YES excepto si existen entidades no implicadas en la hipótesis (evidencia de una respuesta no soportada).

Evaluar la implicación de entidades antes de usar el clasificador SVM

Como se ha dicho arriba, la información sobre entidades es útil para detectar pares sin implicación. Sin embargo, cuando todas las entidades de la hipótesis están implicadas, no hay evidencia suficiente para decidir el valor de implicación. Por tanto, una solución es usar el clasificador SVM anteriormente descrito solamente en este caso. Primero, el sistema da el valor NO a los pares con entidades no implicadas en la hipótesis, a continuación el resto de pares son clasificados con el clasificador SVM.

Clasificación SVM añadiendo el atributo sobre implicación entre entidades

El último experimento fue usar la información sobre implicación entre entidades como un atributo más en el clasificador SVM.

5.3. Resultados

Los sistemas propuestos se evaluaron sobre la colección de test del AVE 2006 en español. El cuadro 5.1 muestra la precisión, cobertura y medida F para los distintos experimentos comparados con un sistema baseline que siempre devuelve YES y con el sistema que obtuvo el mejor resultado en el AVE 2006 para español (Tatu et al., 2006a).

El sistema que usa el clasificador SVM con los atributos básicos obtiene una f de 0.56 que no está nada mal comparada con la del baseline (0.45). Sin embargo, este resultado es peor que si solo se considera la relación de implicación entre entidades (F de 0.59). Esto se debe a la alta cobertura obtenida (0.83). Como se ha mencionado antes, la Validación de Respuestas basada solo en entidades tiene un buen rendimiento para detectar pares sin implicación (consigue un 89 % de precisión en este tipo de pares), pero para el resto de los pares no tiene información suficiente. Por este motivo

Cuadro 5.1: Resultados de los experimentos comparados con el mejor sistema del AVE 2006 y con los sistemas baseline.

Sistema	**Medida F**	**Precisión sobre YES**	**Recall sobre YES**
Mejor sistema AVE 2006	0.61	0.53	0.71
Clasificador SVM con atributo implicación entidades	0.60	0.49	0.77
Decisión de implicación basada solo en implicación de entidades	0.59	0.46	0.83
SVM con atributos baseline	0.56	0.47	0.71
Uso implicación de entidades antes del clasificador SVM	0.55	0.46	0.71
Baseline 100 % YES	0.45	0.29	1

se usó el clasificador SVM básico tomar la decisión de implicación sobre el resto de los pares. Sin embargo, este sistema obtiene pero resultado (F de 0.55). Nos dimos cuenta de que la configuración más apropiada es incluir la relación de implicación entre entidades como un atributo del clasificador SVM, obteniendo una f de 0.60, muy cercana al 0.61 obtenido por el mejor sistema del AVE 2006.

Al analizar los resultados observamos que la mayoría de los errores en la relación de implicación entre entidades se debió a fallos en la detección de las entidades. En muchos casos, los sistemas de Búsqueda de Respuestas cambian el texto original, ignorando la escritura de las entidades y devolviendo todas las palabras del texto soporte en minúscula. En estos casos el reconocedor de entidades nombradas no puede reconocer las entidades del texto y por tanto las entidades de la hipótesis no estarían implicadas (según el procedimiento explicado arriba).

En otros casos, las respuestas se dan en mayúsculas y el reconocedor de entidades nombradas reconoce todas las palabras como entidades a pesar de que a menudo no lo son. Por tanto, en este caso habría falsas entidades en la hipótesis que no pueden ser implicadas.

Para solucionar estos problemas, se requiere de reconocedores de entidades nombradas más robustos y de sistemas de Búsqueda de Respuestas que tengan en cuenta que su salida puede ser tomada como entrada para sistemas de Validación de Respuestas. Por tanto, los sistemas de Búsqueda de Respuestas deberían devolver sus respuestas tal y como aparecen en los textos originales.

5.4. Conclusiones

Se ha definido un criterio para detectar implicación entre dos textos usando implicación entre entidades. Esta definición se puede extender a sistemas de Validación de Respuestas que utilizan RTE. Además se ha comprobado que el uso de esta información como un atributo más en un clasificador SVM mejora los resultados del sistema acercándolo al mejor resultado del AVE 2006.

Comparado con el mejor resultado en el AVE 2006, nuestro sistema obtiene una mayor cobertura y una menor precisión sugiriéndonos que todavía hay trabajo por hacer en filtros más restrictivos para detectar pares sin implicación.

Capítulo 6

Conclusiones

El reconocimiento y clasificación de entidades nombradas es una tarea de estudio bastante asentada que ha sido investigada desde hace varios años. La existencia de foros de evaluación internacionales como las MUC, CoNLL y ACE, al poner al alcance de los investigadores conjuntos de entrenamiento y de prueba así como medidas de evaluación comunes, han ayudado al desarrollo de este tipo de sistemas. Además, el gran número de aplicaciones que tienen en tareas como la búsqueda de respuestas, la recuperación de información o la extracción de información, han ayudado a su rápido desarrollo. Sin embargo, todavía queda por realizar el disponer de sistemas portables a distintos idiomas y dominios.

Una tarea relativamente nueva y en la que también tiene utilidad el reconocimiento de entidades nombradas es la tarea de reconocimiento de Implicación Textual, la cuál ha alcanzado un notable nivel de madurez como demuestra el alto interés por parte de la comunidad de procesamiento del lenguaje natural y el incremento continuo en el número de participantes en las distintas ediciones del PASCAL RTE Challenge. Además, las numerosas publicaciones sobre Implicación Textual han contribuido a un mayor entendimiento de la tarea. En cada nueva edición los participantes muestran que la tecnología aplicada al reconocimiento de implicación está haciendo grandes progresos, confirmados por los resultados.

Son numerosas las tareas de procesamiento del lenguaje natural que se verían favorecidas por el uso de sistemas de Implicación Textual. Un ejemplo de estas tareas lo presenta la Búsqueda de Respuestas, donde la respuesta dada por el sistema debería estar implicada por el texto que soporta dicha respuesta. Tomando esta idea como punto de partida y con el fin de promover el desarrollo de sistemas de validación de respuestas, en 2006 tuvo lugar la primera celebración del AVE.

El punto de partida del AVE 2006 fue la reformulación de la validación de respuestas como un problema de reconocimiento de Implicación Textual bajo la suposición de que la hipótesis puede ser automáticamente obteni-

da instanciando patrones de hipótesis con las respuestas de los sistemas de Búsqueda de Respuestas. De este modo, las colecciones desarrolladas en el AVE están orientadas al desarrollo y evaluación de los sistemas de Validación de Respuestas. Se ha mostrado la metodología para el desarrollo de las colecciones aprovechándose de los juicios humanos hechos en la evaluación de sistemas de Búsqueda de Respuestas. También se ha propuesto una metodología para la evaluación.

Teniendo en cuenta que los documentos escritos contienen un gran número de entidades y que dichas entidades suelen ser importantes para el significado del texto, se ha estudiado la importancia de considerar implicación entre entidades para el reconocimiento de Implicación Textual. Para ello, se ha definido un criterio para detectar implicación entre dos textos usando implicación entre entidades. Como se ha podido comprobar, este enfoque ha incrementado ligeramente el rendimiento de los sistemas propuestos.

Para comprobar la importancia de las entidades en la Implicación Textual aplicada a una determinada tarea, se ha aplicado la definición de implicación entre entidades a sistemas de Validación de Respuestas que utilizan RTE. Además se ha comprobado que el uso de esta información como un atributo más en un clasificador SVM mejora los resultados del sistema acercándolo al mejor resultado del AVE 2006. Comparado con el mejor resultado en el AVE 2006, nuestro sistema obtiene una mayor cobertura y una menor precisión sugiriéndonos que todavía hay trabajo por hacer en filtros más restrictivos para detectar pares sin implicación.

Capítulo 7

Trabajo Futuro

7.1. Relaciones entre entidades

Como se pudo comprobar en el PASCAL RTE Challenge, a veces no vale solo con el reconocimiento de entidades. Hay ocasiones en las que es necesario tener más información sobre dichas entidades. Una información muy útil la constituye el tener conocimiento acerca de las relaciones en las que están involucradas dichas entidades. De este modo se podría añadir un mayor conocimiento semántico al reconocimiento de implicación textual. Este tipo de conocimiento consistiría en comprobar si las relaciones entre las entidades de la hipótesis son compatibles con las relaciones entre las mismas entidades del texto.

Con el fin de llevar a cabo esta tarea, el primer paso será estudiar el estado actual del reconocimiento de relaciones. En concreto este estudio se realizará sobre los sistemas que participan en este tipo de tarea del ACE.

En la tarea de detección y reconocimiento de relaciones del ACE, en inglés Relation Detection and Recognition task (RDR), los sistemas participantes deben detectar determinados tipos de relaciones e indicar determinada información sobre las relaciones reconocidas. En esta tarea se tienen que detectar las relaciones que tienen lugar entre dos entidades del ACE, es decir, entre dos elementos que se consideren entidades según la definición de entidad del ACE (ver 2.1.1). A estas dos entidades se las consideran los argumentos de la relación.

La información que hay que dar sobre cada relación hace referencia a los atributos de la relación (como tipo y subtipo), los argumentos y las distintas menciones de la relación en el documento.

7.2. Clasificación de Preguntas

7.2.1. Tipo esperado de respuesta

En los sistemas de búsqueda de respuestas es importante conocer el tipo de cada pregunta para así facilitar la extracción de la respuesta (ayuda el saber si por ejemplo se está buscando una persona o una fecha) (Li and Roth, 2002). Esto sucede porque a veces no es suficiente con buscar la respuesta usando palabras de la pregunta ya que la respuesta puede contener sinónimos. El tener información sobre el tipo de la pregunta es útil para:

1. Aportar restricciones a los tipos de las respuestas que permiten determinados procesamientos para localizar y verificar la respuesta.

2. Aportar información que se puede usar para determinar la estrategia de selección de la respuesta (que podría ser específica del tipo de respuesta).

Por tanto la tarea de clasificación de preguntas consistiría en dada una pregunta asignarle una de las k clases de la jerarquía que se esté usando. Por ejemplo para la pregunta *¿Qué ciudad canadiense tiene la mayor población?*, el objetivo sería considerar como respuestas candidatas solamente las que sean ciudades.

7.2.2. Taxonomías

Son varias las jerarquías de preguntas que existen. En (Li and Roth, 2002) se muestra una jerarquía que se usa en un sistema real y que está dividida en dos capas y que representa una clasificación semántica para las preguntas típicas de la tarea de búsqueda de respuestas del Text REtrieval Conference (TREC)[1]. La jerarquía consta de seis clases de grano grueso (abreviaturas, entidades, descripciones, humanos, localizaciones y valores numéricos) que se subdividen en 50 clases de grano fino. Cada clase de grano grueso contiene un conjunto no solapado de clases de grano fino.

Otra jerarquía es la de Satoshi Sekine que ya se mencionó en la sección 2.1.2. A la hora de crear esta jerarquía, se tuvieron en cuenta también los sistemas de Búsqueda de Respuestas ya que se creó una serie de preguntas para clasificarlas en dicha jerarquía, y las clases que no existían fueron añadidas.

7.2.3. Trabajos existentes

Debido a la gran cantidad de trabajo manual que es necesario para construir un clasificador para una taxonomía complicada de preguntas, la mayo-

[1] http://trec.nist.gov/

ría de los sistemas de búsqueda de respuestas solo pueden realizar una clasificación de grano grueso con no más de 20 clases.

En (Li and Roth, 2002) se muestra el proceso seguido para construir un clasificador de preguntas usando aprendizaje automático con características como las palabras de la pregunta y la categoría gramatical, obteniendo una precisión superior al 90 %. Para crear el conjunto de entrenamiento, el tipo de las preguntas se anotó a mano de acuerdo a una jerarquía que se estableció previamente. Dado que a veces una pregunta puede pertenecer a dos clases distintas, el clasificador permite asignar más de una clase a cada pregunta. Este método es mejor que asignar una sola clase porque así pueden usar todas las clases en procesamientos posteriores sin perder información.

En (Li et al., 2004), se estudia la importancia de añadir a un clasificador de preguntas (se utiliza el descrito en (Li and Roth, 2002)) información semántica:

- Categorías de entidades nombradas, añadiendo a las categorías que aporta un reconocedor tradicional otras que pueden ser útiles para QA como profesión, evento, deportes...
- Synsets de WordNet. Para ello usan todos los de una palabra sin usar desambiguación.
- Listas de palabras construidas a mano y que están agrupadas en función de cada clase del clasificador.
- Listas de palabras generadas automáticamente y que están agrupadas por su similaridad semántica.

Para la realización del experimento se añadió esta información semántica obteniendo el mejor resultado (89.3 % de precisión en las clases de grano fino) combinando toda la información semántica.

Desde el punto de vista de las entidades nombradas, este tipo de información semántica es la que menos información les aporta por cada pregunta (ya que no todas las preguntas contienen entidades nombradas) y de los tipos de información semántica, que son cuatro, es la única sensible al contexto (ya que en las demás no hay desambiguación).

Capítulo 8

Publicaciones del autor relacionadas con el trabajo

Para dar a conocer el trabajo realizado y sus resultados a la comunidad científica internacional, el autor ha publicado los siguientes artículos:

- Anselmo Peñas, Álvaro Rodrigo and Felisa Verdejo. Sparte, a test suite for recognising textual entailment in Spanish. In Alexander F. Gelbukh, editor, CICLing, volume 3878 of Lecture Notes in Computer Science, pages 275-286. Springer, 2006.

- Jesús Herrera, Anselmo Peñas, Álvaro Rodrigo and Felisa Verdejo. UNED at PASCAL RTE-2 Challenge. Proceedings of the Second PASCAL Recognizing Textual Entailment Workshop. 2006.

- Jesús Herrera, Álvaro Rodrigo, Anselmo Peñas and Felisa Verdejo. UNED Submission to AVE 2006. Results of the CLEF 2006 Cross-Language System Evaluation Campaign. Working Notes. 2006.

- Álvaro Rodrigo, Anselmo Peñas and Felisa Verdejo. The Effect of Entity Recognition in Answer Validation. Results of the CLEF 2006 Cross-Language System Evaluation Campaign. Working Notes. 2006.

- Anselmo Peñas, Álvaro Rodrigo, Valentín Sama and Felisa Verdejo. Overview of the Answer Validation Exercise 2006. Results of the CLEF 2006 Cross-Language System Evaluation Campaign. Working Notes. 2006.

- Álvaro Rodrigo, Anselmo Peñas, Jesús Herrera and Felisa Verdejo. Experiments of UNED at the Third Recognising Textual Entailment Challenge. Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, pp. 89-94, Prague 2007.

- Anselmo Peñas, Álvaro Rodrigo, Valentín Sama and Felisa Verdejo. Testing the Reasoning for Question Answering Validation. Special Issue on Natural Language and Knowledge Representation, Journal of Logic and Computation. To appear.
- Anselmo Peñas, Álvaro Rodrigo, Valentín Sama and Felisa Verdejo. Overview of the Answer Validation Exercise 2006. CLEF 2006, Lecture Notes in Computer Science LNCS 4730. Springer-Verlag, Berlín.
- Álvaro Rodrigo, Anselmo Peñas, Jesús Herrera and Felisa Verdejo. The effect of entity recognition on answer validation. CLEF 2006, Lecture Notes in Computer Science LNCS 4730. Springer-Verlag, Berlín.
- Anselmo Peñas, Álvaro Rodrigo and Felisa Verdejo. Overview of the Answer Validation Exercise 2007. Working Notes for the CLEF 2007 Workshop, 19-21 September, Budapest, Hungary.
- Álvaro Rodrigo, Anselmo Penas and Felisa Verdejo. UNED at Answer Validation Exercise 2007. Working Notes for the CLEF 2007 Workshop, 19-21 September, Budapest, Hungary.

Capítulo 9

Agradecimientos

9.1. Agradecimientos institucionales

El presente trabajo de investigación ha sido financiado en parte por el Ministerio de Ciencia y Tecnología, con cargo a los presupuestos del proyecto SyEMBRA (Sistemas y Evaluación Multilingües de Búsqueda de Respuestas) TIC-2003-07158-C04-02, el programa de becas predoctorales UNED, la Comunidad de Madrid bajo la Red de Investigación MAVIR (S-0505/TIC-0267), la Consejería de Educación de la Comunidad de Madrid y el Fondo Social Europeo (F.S.E.),

9.2. Agradecimientos personales

La primera persona a quién le tengo que estar agradecido es a mi director Anselmo Peñas. Desde que entré en el mundo de la investigación él ha sido mi guía, sabiendo ejercer a la vez de mentor y amigo. Sin él no habría sido posible llegar hasta aquí.

También tengo mucho que agradecer al culpable de que esté aquí, me refiero a Jesús Herrera. Siempre que le he necesitado ha estado ahí para ayudarme tanto en lo personal como en lo profesional.

Junto a Jesús tengo que dar las gracias a todos los demás que han sido compañeros míos de despacho. Me refiero a Iñaki, José Luis, Cristina y Manuel. Ellos han hecho que el ambiente de trabajo sea inmejorable y me han ayudado a resolver muchas dudas.

También le estoy muy agradecido a Felisa por sus sabios consejos y por dirigir un grupo de gente tan estupendo. La verdad es que da gusto trabajar en un ambiente tan cordial y con gente tan maravillosa. Gracias a todos vosotros de verdad. Gracias a Cova, Tim, Alberto, Teresa, Víctor Fresno, Juanci, Ana, Raquel, Beti, Javi Artiles, Víctor Peinado, Enrique, Valentín, Carlos Vicente, David, Sergio, Nacho, Fernando, Miguel, Julio, Carlos Celorrio, Emilio, Javi Vélez, Lourdes, Fátima y Yoli.

De la UNED también quiero dar las gracias a Juan Antonio, Rubén, Manuel, Eduardo y Félix.

Fuera del ámbito de la universidad, los primeros a los que tengo que dar las gracias es a mi familia por confiar en mí y apoyarme. En especial tengo mucho que agradecer a mis padres y a mi hermano por estar siempre a mi lado y darme fuerzas para continuar hacia delante.

Gracias también a Javi, Bea, Raquel y a mis compañeros de inglés por vuestro apoyo y amistad.

Por último, no quiero terminar sin dar las gracias a Santiago por sus enseñanzas, las cuáles me han sido de gran utilidad durante este tiempo.

Bibliografía

R. Bar-Haim, I. Dagan, B. Dolan, L. Ferro, D. Giampiccolo, B. Magnini, and Idan Szpektor. The second pascal recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment, Venezia, Italy*, April 2006.

D. Bikel, S. Miller, R. Schwartz, and R. Weischedel. Nymble: a high-performance learning name-finder. In *ANLP*, pages 194–201, 1997.

X. Carreras, I. Chao, L. Padró, and M. Padró. Freeling: An open-source suite of language analyzers. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC04). Lisbon, Portugal*, 2004.

N. Chinchor. Overview of muc-7. In *MUC-7*, 1998.

M. Collins and Y. Singer. Unsupervised models for named entity classification. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999.

S. Cucerzan and D. Yarowsky. Language independent named entity recognition combining morphological and contextual evidence. In *1999 Joint SIGDAT Conference on EMNLP and VLC*, 1999.

S. Cucerzan and D. Yarowsky. Language independent ner using a unified model of internal and contextual evidence. In *Proceedings of The Sixth Conference on Natural Language Learning (CoNLL). Taipei, 2002.*, 2002.

I. Dagan, O. Glickman, and B. Magnini. The pascal recognising textual entailment challenge. In *Quiñonero-Candela et al., editors, MLCW 2005, LNAI Volume 3944, pages 177-190. Springer-Verlag.*, pages 177–190, 2006.

G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, and R. Weischedel S. Strassel. The automatic content extraction (ace) program tasks, data, and evaluation. In *Proceedings of LREC 2004*, 2004.

D. Giampiccolo, B. Magnini, I. Dagan, and B.l Dolan. The third pascal recognizing textual entailment challenge. In *ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, 2007.

Ralph Grishman and Beth Sundheim. Message understanding conference-6: a brief history. In *Proceedings of the 16th conference on Computational linguistics - Volume 1.*, 1996.

S. Harabagiu and A. Hickl. Methods for using textual entailment in open-domain question answering. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL, pages 905-912, Sydney.*, 2006.

S. Harabagiu, D. Moldovan, C. Clark, M. Bowden, J. Williams, and J. Bensley. Answer mining by combining extraction techniques with abductive reasoning. In *TREC*, pages 375–382, 2003.

J. Herrera, A. Peñas, Á. Rodrigo, and F. Verdejo. Uned at pascal rte-2 challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment, Venice, Italy.*, 2006a.

J. Herrera, A. Peñas, and F. Verdejo. Textual entailment recognition based on dependency analysis and wordnet. In *Quiñonero-Candela et al., editors, MLCW 2005, LNAI Volume 3944, Jan 2006, Pages 231-239*, 2006b.

A. Hickl, J. Williams, J. Bensley, K. Roberts, B. Rink, and Y. Shi. Recognizing textual entailment with lcc groundhog system. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment, Venice, Italy.*, 2006.

B. Kwak and J. Cha. Named entity tagging for korean using dl-cotrain algorithm. In *Information Retrieval Technology, Second Asia Information Retrieval Symposium, AIRS 2005, Jeju Island, Korea, October 13-15, 2005, Proceedings*, pages 589–594, 2005.

V. Levensthein. Binary codes capable of correcting deletions, insertions and reversals. In *Soviet Physics - Doklady*, volume 10, pages 707–710, 1966.

X. Li and D. Roth. Learning question classifiers. In *Proceedings 19th Itnernational conference on Computacional Linguistics*, 2002.

Xin Li, Dan Roth, and Kavin Small. The role of semantic information in learning questions classifiers. In *First International Joint Conference on Natural Language Processing.*, 2004.

D. Lin. Dependency-based evaluation of minipar. In *Workshop on the Evaluation of Parsing Systems, Granada, Spain*, May 1998.

P. McNamee and J. Mayfield. Entity extraction without language-specific resources. In *COLING-02: proceeding of the 6th conference on Natural language learning*, pages 1–4, Morristown, NJ, USA, 2002. Association for Computational Linguistics.

D. Moldovan, C. Clark, S. Harabagiu, and S. Maiorano. Cogex: A logic prover for question answering. In *HLT-NAACL*, 2003.

D. Palmer and D. Day. A statistical profile of the named entity task. In *ANLP*, pages 190–193, 1997.

A. Peñas, Á. Rodrigo, V. Sama, and F. Verdejo. Overview of the answer validation exercise 2006. In *Lecture Notes in Computer Science*, 2007.

Á. Rodrigo, A. Peñas, J. Herrera, and F. Verdejo. The effect of entity recognition on answer validation. In *Lecture Notes in Computer Science.*, 2007.

E. Sang. Introduction to the conll-2002 shared task: Language-independent named entity recognition. *CoRR*, 2002.

E. Sang and F. De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *CoRR*, 2003.

S. Sekine and H. Isahara. Irex: Ir and ie evaluation project in japanese. 2000.

S. Sekine, K. Sudo, and C. Nobata. Extended named entity hierarchy. In *Proceedings of the LREC-2002.*, 2002.

H. Tanev and B. Magnini. Weakly supervised approaches for ontology population. In *EACL 2006, 11st Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference, April 3-7, 2006, Trento, Italy*, 2006.

M. Tatu and D. Moldovan. Cogex at rte 3. In *ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, 2007.

M. Tatu, B. Iles, and D. Moldovan. Automatic answer validation using cogex. In *Working Notes. CLEF 2006 Workshop, 20-22 September, Alicante, Spain*, 2006a.

M. Tatu, B. Iles, J. Slavick, A. Novischi, and D. Moldovan. Cogex at the second recognizing textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment, Venice, Italy.*, 2006b.

A. Vallin, B. Magnini, D. Giampiccolo, L. Aunimo, C. Ayache, P. Osenova, A. Peñas, M. de Rijke, B. Sacaleanu, D. Santos, and R. Sutcliffe. Overview of the clef 2005 multilingual question answering track. In *Proceedings of CLEF 2005. LNCS*, 2005.

L. Vanderwende and W. Dolan. What syntax can contribute in entailment task. in mlcw 2005, lnai 3944, pp. 205-216. j. quinonero-candela et al. (eds.). springer-verlag. In *MLCW 2005, LNAI 3944, pp. 205-216. J. Quinonero-Candela et al. (eds.). Springer-Verlag.*, 2005.

J. Vicedo. La búsqueda de respuestas: Estado actual y perspectivas de futuro. *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*, 22:37–56, 2003.