SISTEMA BASELINE PARA DETECCIÓN DE PARÁFRASIS EN DOMINIOS ABIERTOS.

M. Sc. Yaniseth Garcia Vasconcelo, M. Sc. . Antonio Fernández Orquín

Universidad de Matanzas, Autopista a Varadero Km 31/2, Cuba.

CD de Monografías 2008

Resumen.

Se ha desarrollado un método para la detección de paráfrasis basado en una combinación de técnicas y herramientas del procesamiento del lenguaje natural, lo que permite conocer cuándo dos frases son semánticamente equivalentes. Esto resulta de gran importancia para los sistemas que intentan hacer "pensar" a la computadora, por lo que sus aplicaciones son múltiples: extracción de información, búsqueda de respuesta, localización de información, traducción automática, generación de resúmenes, detección de plagios, etc. En el caso específico de la educación se han realizados trabajos vinculados con la evaluación y la detección de fraudes. Se expone un estudio del estado del arte, se plasman los elementos teóricos que sirven de base para el desarrollo de la propuesta. Para el análisis de los resultados obtenidos se utilizan los indicadores de exactitud, precisión y cobertura, calculando la F-medida y comparando sus valores con los obtenidos por sistemas internacionales probados sobre el mismo corpus.

***Palabras claves**: Detección de paráfrasis; Paráfrasis en dominios abiertos; baseline para detección de paráfrasis.*

Introducción.

Una de las ramas más importantes de la Inteligencia Artificial (IA) es aquella que formula mecanismos computacionalmente efectivos para facilitar la comunicación hombre-máquina por medio del lenguaje natural, logrando que sea esta comunicación mucho más fluida y menos rígida que los lenguajes formales. El Procesamiento del Lenguaje Natural (PLN) es el encargado de producir sistemas informáticos que posibiliten esta tarea, por medio de la voz o del texto. El uso del lenguaje natural (LN) en la comunicación hombre-máquina tiene ventajas que implican grandes dificultades, entre ellas el hecho de que existen muchas formas de expresar la misma idea. Si las expresiones que se comparan son palabras o frases muy cortas, se dice que son sinónimos y existen investigaciones que han arrojado muy buenos resultados en el análisis de este tipo de relación léxica, como es el caso del recurso *WordNet*[1], pero si las expresiones a comparar son largas o complicadas, se consideran paráfrasis.

Recolectar paráfrasis manualmente es una tarea altamente consumidora de tiempo y dinero, por lo que recientemente este tema ha tenido mayor atención a nivel mundial. Han existido algunos esfuerzos por descubrir la paráfrasis automáticamente a partir de corpus, pero resultan insuficientes. Se encuentran limitados por el uso de diferentes umbrales de frecuencia que imposibilitan detectar relaciones entre entidades menos frecuentes y por tanto las paráfrasis que contengan estas entidades. (Hasegawa, 2005) no son capaces de detectar una gran variedad de paráfrasis. No reconocen cuando se hace referencia a una misma entidad con palabras diferentes, situación esta que se da muy a menudo, por lo que se deja de detectar una suma considerable de relaciones de equivalencia semántica.

En los patrones de generación de paráfrasis le dan el mismo peso a todas las anclas identificadas por las entidades, aspecto este que se vuelve vulnerable para las entidades ampliamente usadas,

[1] Aplicación desarrollada por bajo la dirección del profesor de psicología George A. Miller, en el Laboratorio de Ciencia Cognitiva de la Universidad de Princeton, EE.UU, a partir de 1985.

que deberían tener menos crédito. Se consideran las entidades intercambiables, cuando esto no es posible en todos los contextos (Shinyama, 2003). Utilizan umbrales de frecuencia, pero para la paráfrasis completa, lo que hace que deje de analizar aquellas que se repitan poco a través de los documentos (Shinyama, 2005). Se aferran a las entidades para la búsqueda de paráfrasis y calculan la similitud entre las oraciones sólo por las coincidencias de estas. Los patrones de generación de paráfrasis están limitados a un único camino en un árbol de dependencias, lo que acota considerablemente su variedad (Shinyama, 2002). No consideran la posibilidad de que existan más de cinco palabras entre un par de entidades (Sekine, 2005). La generalización de paráfrasis se lleva a cabo solamente a nivel de palabras y no de frases (Kauchak, 2006).

En el contexto anterior se enuncia el siguiente problema de investigación: ¿Podrá un método formado por una nueva combinación de técnicas y herramientas del procesamiento del lenguaje natural, detectar la paráfrasis en dominios abiertos?. Como objeto de investigación se ha determinado la detección de la paráfrasis, definiéndose como campo de acción los dominios abiertos para el idioma inglés. Como principal elemento a demostrar se ha propuesto la siguiente hipótesis**:** Si se diseña un método que extraiga las categorías gramaticales, analice la existencia de relación de sinonimia, hiperonimia[2] o hiponimia[3] entre los verbos de dos frases y compare las palabras restantes de acuerdo a un umbral de coincidencias, puede detectarse la paráfrasis en dominios abiertos. El objetivo de la investigación es desarrollar un método para la detección de la paráfrasis en dominios abiertos empleando una nueva combinación de técnicas y herramientas de PLN.

Un método basado en una combinación innovadora de técnicas y herramientas de PLN capaz de detectar paráfrasis y un sistema *baseline*[4] que lo implemente constituyen la novedad científica de esta investigación. Para ello se estructura este trabajo de la siguiente forma:

Fundamentación teórica.

1.1 Introducción.

Un sistema de detección de paráfrasis intenta obtener pares (o conjuntos) de expresiones que constituyan paráfrasis entre sí, por lo que el "Sistema *baseline* para la detección de paráfrasis en dominios abiertos", PADET de ahora en adelante, no constituye una excepción, de ahí la importancia de tener claridad en el significado del término "paráfrasis".

Según Regina Barzilay en (Barzilay, 2001) es un conjunto de expresiones intercambiables que son ligeramente diferentes en su matiz, verbosidad, etc. Para Dekang Lin y Patrick Pantel en (Lin, 2002) constituye un conjunto de patrones para capturar varios tipos de información a partir de documentos, sin embargo, en (Yamamoto, 2002) Kazuhide Yamamoto considera que es el grado de replicabilidad de dos expresiones E1 y E2, las cuales son diferentes entre sí en algunos sentidos. Yusuke Shinyama; Satoshi Sekine y Kiyoshi Sudo exponen una idea mucho más escueta en (Shinyama, 2002). Para ellos las paráfrasis son simplemente expresiones similares, pero el primer autor, Yusuke Shinyama en un trabajo posterior (Shinyama, 2005) las considera

---

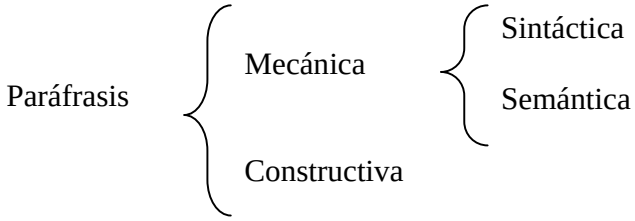
[2] Describe la relación "es un". A es un hiperónimo de B si B es un subconjunto de A. (A→B)
[3] Describe el otro lado de la relación "es un". B es un hipónimo de A si B es un subconjunto de A. (A←B)
[4] Sistema base con el cual se comparan los trabajos futuros. Establece una línea de comparación.

ocurrencias de pares de caminos extraídos, mientras que Takaaki Hasegawa, Satoshi Sekine y Ralph Grishman piensan en (Hasegawa, 2005) que son un conjunto de frases las cuales expresan la misma cosa o evento. Otras definiciones son la de Yves Lepage y Etienne Denoual en (Lepage, 2005): oraciones diferentes que tienen la misma traducción y la de Chris Callison-Burch; Philipp Koehn y Miles Osborne en (Callison-Burch, 2006): formas alternativas de expresar la misma información dentro de un lenguaje. Con el estudio de las diferentes definiciones se pudo apreciar que la mayoría de los autores consideran el término paráfrasis en función del proceso de detección que aplican y la funcionalidad que va a desarrollar.

Los autores de esta investigación definen paráfrasis como: aquellas frases que pueden o no tener pequeñas variaciones en su estructura sintáctica y son formuladas con palabras diferentes, pero manteniendo el mismo significado.

1.2 Clasificación de la paráfrasis.

Las paráfrasis pueden clasificarse para su estudio en:



- Paráfrasis mecánica: Consiste en sustituir alguna palabra por sinónimos o frases alternas con cambios sintácticos mínimos.
    - Sintáctica: Está dada básicamente por la relación existente entre lo activo, lo pasivo y lo relativo. Ejemplo:

El gato arañó a la niña.	I
A la niña la arañó el gato.	II
La niña fue arañada por el gato.	III

    - Semántica: Está dada por la utilización de sinónimos para expresar el mismo significado de palabra sin alterar la estructura de la oración. Ejemplo: utilizando las combinaciones sólo de los sinónimos que a continuación se presentan se obtienen ciento veinte variantes de la oración I.

El	gato	arañó	a la	niña.
	felino	rasguñó		nena.
	miau	escarbó		infanta.
	minino	raspó		pequeña.
		desgarró		chica.
				chiquilla.

- Paráfrasis constructiva: Esta otra en cambio reelabora el enunciado dando origen a otro con características muy distintas conservando el mismo significado. Ejemplos:

Sólo el 30% de los estudiantes aprobó el examen. I

La mayoría de los estudiantes desaprobaron el examen. II

Emma lloró. I

Emma estalló en lágrimas. II

La detección de paráfrasis se encarga precisamente de determinar estos pares o conjuntos de frases con el mismo significado de forma automática.

1.3 Aplicaciones de la paráfrasis.

- Extracción de Información: conocer si una información parafrasea a otra que ya fue almacenada, para evitar redundancia en la base de datos, sería de gran importancia. También para extraer información, que esté escrita de alguna forma específica, pudiera ser importante buscarla por sus formas alternativas.

- Generación de Resúmenes: Comparar las frases de aquel o aquellos artículos que se deseen resumir para garantizar que no falte alguna idea importante o se repita una redundante garantiza la calidad y el tamaño óptimo del resumen.

- Traducción Automática: La paráfrasis multilingüe sería capaz de conocer que dos frases dichas o escritas en lenguajes diferentes, con palabras y estilos propios de cada autor, conservan el mismo significado.

- Recuperación de Información: Se puede recuperar la información no en la forma exacta que se solicita, sino también otras palabras en el mismo contexto que son capaces de expresar igual significado.

- Búsqueda de Respuesta: Si la pregunta que se hace, a pesar de contener palabras y estructuras diferentes, posee un significado conocido e igual al de alguna pregunta anterior, no hay necesidad de volver a buscar, la respuesta ya se sabe, la misma.

- Detección de plagios: Con un sistema de detección de paráfrasis se podría facilitar la ardua labor de desenmascarar a los piratas de la propiedad intelectual.

1.4 Estado del arte.

Entre los estudiosos del tema se encuentran algunos que sobresalen por sus resultados y cuyos trabajos serán referidos a continuación:

Regina Barzilay y Kathleen R. McKeown en (Barzilay, 2001) implementan un algoritmo de aprendizaje no supervisado para la identificación de paráfrasis en una colección de traducciones en inglés de una misma fuente, atendiendo al principio de que cada autor, por muy creativo que sea, para contar una misma historia en el mismo idioma debe mantener un conjunto de palabras

constantes, es decir, buscar pares de expresiones A C B y A D B con los mismos argumentos (A, B) o con aquellos que brinden la misma información: entonces C y D tienen una buena probabilidad de formar parte de una paráfrasis. Luego se establece el contexto que define esa paráfrasis (A, B) y se buscan otras oraciones que macheen con este patrón. De esta manera analiza la paráfrasis léxica y la sintáctica.

Yusuke Shinyama; Satoshi Sekine y Sudo Kiyoshi en (Shinyama, 2002) parten de la misma idea de Regina Barzilay y Kathleen R. McKeown en (Barzilay, 2001), pero utilizan entidades, es decir, obtienen paráfrasis a partir de artículos de noticias que describen el mismo evento, asumiendo que para contar un mismo hecho, el mismo día, es necesario mantener constante un conjunto de entidades (nombres de organizaciones o personas, fechas, localizaciones, expresiones numéricas, etc) las cuales son preservadas a través de la paráfrasis.

Kazuhide Yamamoto en (Yamamoto, 2002) propone un método de adquisición automática de paráfrasis basado en el conocimiento del contexto de las palabras. Utiliza como entrada artículos de periódicos, los cuales analiza morfológicamente y parsea[5] para obtener relaciones triples: (c1, r, c2) donde la palabra c1 depende de c2 por la relación r. c1, c2 y r son verbos y sustantivos y definen un contextos, pero contextos iguales no garantizan similitud semántica, por lo que utilizan diccionarios de pares de antónimos para una mejor precisión, puesto que considera que los sinónimos y los hiperónimos introducen mucho ruido. Luego convierte cada tripleta en un bigrafo: (c1, r→c2) y (c2, r←c1) que le permite medir el grado de similitud de dos contextos en término de sus relaciones, atendiendo al principio de que mientras menor sea la frecuencia de una relación en el corpus, mayor será la probabilidad de que sean paráfrasis.

Ali Ibrahim, Boris Katz y Jimmy Lin, continúan en (Ibrahim, 2003) la idea de Regina Barzilay y Kathleen R. McKeown en (Barzilay, 2001), pero a diferencia de esta que utiliza patrones rígidos con palabras contiguas, a lo sumo separadas por otra palabra, aplica la idea de Dekang Lin y Patrick Pantel en (Lin, 2002) de implementar un árbol de dependencia para hacer más flexible el patrón, considerando que siempre que exista un camino de dependencia entre las palabras que forman el patrón hay paráfrasis.

Takaaki Hasegawa; Satoshi Sekine, y Ralph Grishman en (Hasegawa, 2005) implementan un método no supervisado para descubrir paráfrasis que contengan dos entidades desde un gran corpus no etiquetado. Al igual que Yusuke Shinyama; Satoshi Sekine y Sudo Kiyoshi en (Shinyama, 2002), utilizan un etiquetador de entidades para determinar estas, pero escogen las dos entidades cuando se encuentran relacionadas por menos de 5 palabras y aparecen más de 30 veces en el corpus. Luego aplican la medida del coseno al vector resultante del contexto y forman grupos jerárquicos con los pares de entidades respetando el nivel de similitud, para poder producir paráfrasis.

Weigang Li... et al. en (Li, 2005) describen una nueva representación de paráfrasis en plantillas y un método de generalización. Para ello reciben como entrada un conjunto de paráfrasis que son analizadas con un diccionario semántico que desambigua el sentido de las palabras y define los espacios en blanco con un código, es decir, los lugares donde podrá ir otra palabra que mantenga

---

[5] Acción que ejecuta un parser o analizador sintáctico y que concluye con la construcción de un árbol sintáctico.

cierta similitud con la que había según ese código. Como es obvia la limitación que introduce un código único, aplican una ingeniería de búsqueda (un *parser* de dependencias) en cada ejemplo para extender los grupos de palabras de los espacios y generalizar los ejemplos.

Yusuke Shinyama en (Shinyama, 2005) se centra en expresiones que podrían producir la misma información, y lo aplica a la extracción de información. Estas expresiones las adquiere a partir de corpus comparables cuyas palabras con valor semántico son utilizadas para calcular la medida del coseno a la primera oración de cada artículo. Esta operación permite conocer si su grado de similitud es mayor a cierto umbral. Una vez escogido el par de corpus de fuentes diferentes con que va a trabajar, selecciona la primera oración de cada corpus por ser la más importante y las parsean. Luego utiliza un solucionador de correferencias y prueba todas las combinaciones posibles entre las frases hasta encontrar las anclas. Una vez encontrados los puntos de contactos comienza a generar paráfrasis a través del recorrido por los caminos de los árboles.

Jesús Herrera de la Cruz implementa en (Herrera de la Cruz, 2005) un sistema para una de las tareas de El *PASCAL RTE*[6] *Challenge*: la implicación textual, específicamente la detección automática de implicación semántica entre parejas de textos en lenguaje natural (monolingüe inglés), lo que incluye entre sus aplicaciones la Adquisición de Paráfrasis. El sistema extrae del corpus de entrada los textos e hipótesis y alimenta el analizador de dependencias *Minipar*[7] para generar los árboles. Luego ejecuta un módulo de implicación léxica sobre los nodos y obtiene como salida una lista de pares <T, H>, donde la unidad léxica de T implica a la unidad léxica de H. Esta implicación a nivel léxico se determina teniendo en cuenta relaciones de sinonimia e hiperonimia de *WordNet*. La negación no genera implicación, pero se analiza utilizando la relación de antonimia. *WordNet*, unido a un reconocimiento difuso de la coincidencia entre candidatos a multipalabras, mediante la distancia de edición de Levenshtein, permiten el análisis de las multipalabras. Finalmente interviene un módulo de evaluación de solapamiento, que busca ramas en el árbol de dependencias de las hipótesis conformadas por nodos léxicamente implicados por nodos del texto. Cuando el árbol de dependencias de una hipótesis muestra un porcentaje de solapamiento de nodos mayor o igual al 50 %, considera que hay implicación entre el texto y la hipótesis.

Thierry Poibeau en (Poibeau 2005) describe un método supervisado por el analista, quien le tiene que proporcionar al sistema un ejemplo inicial. Los resultados dependen en del ejemplo dado, pues de este obtiene el patrón que busca en el corpus para determinar paráfrasis por similitud semántica. Comienza la comparación por el sustantivo cabecera y si sobrepasa cierto umbral continúan analizando el verbo y los complementos. Finalmente obtiene una tabla con las estructuras predicativas que son semánticamente equivalentes al patrón del ejemplo inicial. El proceso usa los corpus y la red semántica como dos fuentes de conocimiento complementarias diferentes. La primera le brinda expresiones posibles y filtra las irrelevantes, mientras que la segunda le provee información acerca de las semánticas léxicas y las relaciones entre palabras.

---

[6] Textual Entailment Recognition, el reconocimiento de inferencias textuales fue propuesto como una tarea genérica que intenta capturar las mayores referencias semánticas necesitadas a través de las diferentes aplicaciones del PLN.

[7] Analizador léxico sintáctico desarrollado por Dekang Lin.

Yves Lepage y Etienne Denoual en (Lepage, 2005) enfocan su método hacia la traducción automática, por lo que utilizan un corpus multilingüe donde la detección de paráfrasis se reduce a la búsqueda de oraciones que compartan la misma traducción en el recurso. Luego genera paráfrasis basado en el hecho de que cualquier oración dada puede compartir permutaciones con otras oraciones del corpus. Finalmente aplica un algoritmo de filtrado a las paráfrasis candidatas para eliminar las que no tengan una semántica lógica.

Chris Callison-Burch; Philipp Koehn y Miles Osborne en (Callison-Burch, 2006) utilizan la paráfrasis para la traducción automática. La frase original que se desea parafrasear la traducen a un idioma conocido (inglés) y luego extraen palabras de esta para volverlas a traducir al lenguaje fuente. Estas nuevas palabras son sustituidas en la oración original, logrando así generar paráfrasis.

David Kauchak y Regina Barzilay en (Kauchak, 2006) exploran el uso de los métodos de paráfrasis en las técnicas de refinamiento de evaluación automática. Dada una oración de referencia y una oración generada por la máquina (paráfrasis) devuelven otra que esté más cercana a la oración de referencia que la anterior. Para ello trata de sustituir la mayor cantidad posible de palabras de la oración generada en la oración de referencia, sin que esta última pierda su sentido, buscando si existe relación de sinonimia entre las palabras que no se repiten de ambas oraciones. Como no en todos los contextos los sinónimos son intercambiables utiliza un clasificador de dependencias contextuales, el cual entrena con un gran corpus de oraciones que contienen la palabra usada en ese sentido e igual cantidad de oraciones para el caso contrario.

Yitao Zhang y Jon Patrick continúan extendiendo su idea, y ya en (Zhang, 2006) ofrecen un método para la detección de pares de paráfrasis que transforma el texto para que sea más genérico y simple en alguna medida que el original. La idea principal es que si dos oraciones son paráfrasis entre sí y se transforman, tienen una mayor oportunidad de ser transformadas en oraciones similares que si no son paráfrasis. En el estado de aprendizaje supervisado utiliza el módulo de árbol de decisión de *Weka*[8] que aplica varias características:

Samuel Fernando en (Fernando, 2007) hace un estudio de trabajos recientes en el área de la identificación de paráfrasis y a continuación se mencionan alguno de ellos.

matrixLin implementa la métrica *lin* del *WordNet Similarity* (Pedersen, 2004) (Ver Glosario de términos para fórmula de la métrica)

Mihalcea06 presenta un método de búsqueda de similitud semántica de segmentos de textos cortos. La motivación detrás de este método es que mientras más sofisticadas sean las medidas de similitud entre palabras más exactos serán los valores de similitud entre oraciones, por lo que se prueba un conjunto de métodos para mejorar la medida de solicitud palabra-palabra resultando ser los mejores, las seis métricas implementadas en *WordNet Similarity*

[8] Es un software programado en Java que está orientado a la extracción de conocimientos desde bases de datos con grandes cantidades de información, pero el hecho de que sea desarrollado bajo licencia GPL lo ha hecho una alternativa muy interesante.

Qiu06 identifica los trozos de información común y los aparea. Luego utiliza un *parser* sintáctico que le permite extraer tuplas de información y usarlas como argumento del predicado. Estas se comparan directamente y se les da un factor de peso, lo que se considera el indicador de equivalencia más importante. Las tuplas con mayor similitud son apareadas hasta un cierto umbral y las que queden sin aparear son analizadas con un método heurístico basado principalmente en los sustantivos.

Zhang05 una vez delimitado el contenido que presenta probabilidad de similitud, se realizan transformaciones a las oraciones y se van comparando léxicamente hasta obtener una equivalencia. Utiliza tres transformaciones: reemplazo, activo-pasivo, futuro.

Construcción de la solución propuesta.

Introducción

Se desarrollaron las tareas y los casos de pruebas funcionales de cada HU. Se le dio seguimiento al plan de iteraciones y entregas y se analizaron las incidencias que ocurrieron en la medida que avanzo el proyecto. No se incluyen todas las tareas ni casos de pruebas con el objetivo de no hacer engorroso y extenso el trabajo.

Descripción de la solución propuesta.

Se declara una clase principal llamada *BaseLine* que consta de una serie de métodos para facilitar la solución del problema planteado. Estos métodos permiten extraer un par de frases a la vez y almacenarlas en un fichero texto que se le pasa al *FreeLing*. Dicho recurso devuelve otro fichero texto con ambas frases tokenizadas y correctamente identificadas. Luego se procesa éste y se recogen las formas verbales simples y las palabras con valor semántico de cada frase por separado. La lista que almacena las formas verbales simples de la hipótesis se itera y se ejecuta en cada iteración el *WordNet*, pasándole como parámetro el verbo a buscar. Este tesauro deposita sus salidas también en un fichero texto, que es recorrido y su información almacenada en otra lista antes de pasar a la siguiente iteración. Después ambas listas son recorridas: la que recoge las formas verbales simples de la oración texto y la que contiene los *synsets* encontrados en *WordNet*. Cuando un elemento de la primera lista es encontrado en la segunda, se abandona la búsqueda y se incrementa un contador. Finalmente este contador indica la cantidad de verbos del texto que guardan relación de sinonimia, hiperonimia o hiponimia con los verbos de la hipótesis. Si esta cantidad supera el 50% se analizan los complementos, en caso contrario se devuelve 0, lo que quiere decir que no son paráfrasis. Para analizar los complementos se utiliza el mismo método de comparación entre listas que se explicó anteriormente para los verbos, pero esta vez con parámetros diferentes: listas de palabras con valor semántico y se utiliza el mismo criterio de medida, lo que quiere decir que si la hipótesis contiene la mitad de las palabras del texto presentan equivalencia semántica. Por último, estos valores resultantes (0 ó 1) que hasta entonces estaban siendo recogidos en una cadena son almacenados en un fichero texto, desde donde se utilizan para realizar los cálculos pertinentes.

.

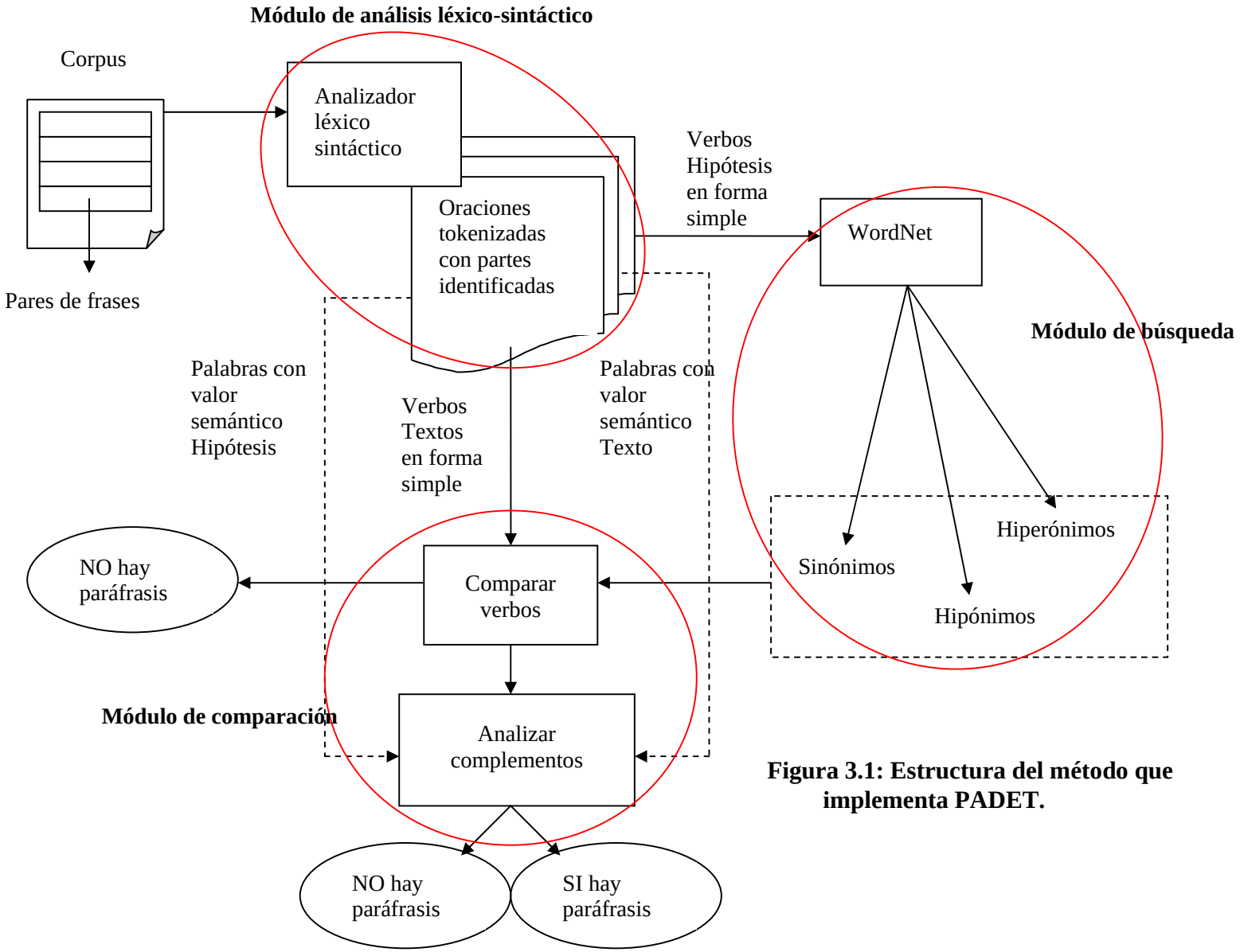


Figura 3.1: Estructura del método que implementa PADET.

Validación del sistema.

Introducción.

Se abordaron en el trabajo los elementos de prueba de *software* que ayudan a validar los resultados. Se trazo una estrategia de prueba que incluye el diseño de casos de pruebas, con el uso de técnicas de pruebas de caja blanca y caja negra. Sólo se expondrán por su importancia las pruebas de caja negra.

Pruebas de caja negra.

Las pruebas de caja negra también son conocidas como pruebas de comportamiento o funcionales porque se basan en los requisitos funcionales del *software*. No es una alternativa a las técnicas de prueba de caja blanca, sino más bien un enfoque complementario que intenta descubrir otros tipos de errores. (Pressman, 2002)

El componente se ve como una "Caja Negra" cuyo comportamiento sólo puede ser determinado estudiando sus entradas y las salidas obtenidas a partir de ellas. Se selecciona un conjunto de entradas, que en este caso es el fichero de prueba del Corpus de la *Microsoft* y se realizan un conjunto de cálculos sobre la salida para tener una medida de cuán funcional resultó ser el sistema.

A continuación se detallan las métricas calculadas a partir de las fórmulas presentadas por Samuel Fernando en la página 18 de "*Paraphrase Identification*" (Fernando, 2007) y sus significados:

Exactitud: cuantifica la capacidad que tiene el sistema para clasificar como paráfrasis aquellas que realmente lo son y como no paráfrasis las que no lo son.

$$exactitud(Acc) = \frac{TP + TN}{TP + TN + FP + FN}; \qquad (4.2)$$

Precisión: cuantifica la capacidad que tiene el sistema para detectar correctamente las paráfrasis.

$$precisión(\mathrm{Pr}ec) = \frac{TP}{TP + FP}; \qquad (4.3)$$

Cobertura: cuantifica la capacidad que tiene el sistema para capturar la mayor cantidad posible de paráfrasis.

$$cobertura(\mathrm{Re}c) = \frac{TP}{TP + FN}; \qquad (4.4)$$

Donde:

TP: Verdaderos positivos, TN: Verdaderos negativos, FP: Falsos positivos, FN: Falsos negativos.

F_Medida: es una función que regula el balance entre precisión y cobertura permitiendo compara sistemas con diferentes valores en ambos parámetros.

$$F\_Medida(F) = \frac{2 * precisión * cobertura}{precisión + cobertura} \quad ; \qquad (4.5)$$

Calculando en las fórmulas anteriores con los valores arrojados por el sistema, se obtiene una exactitud de 66.3, para una precisión de 70.6, en una cobertura de 84.5, lo que representa una F Medida de 76.9. Estos valores son incluidos en la Tabla 4.1 extraída de Samuel Fernando (Fernando, 2007) para compararlos con otros sistemas cuyo funcionamiento fue explicado en el epígrafe de Estado del Arte.

Nota: Los valores mostrados por estos sistemas fueron obtenidos sobre el mismo corpus de prueba que se utilizó en esta investigación y las métricas calculadas con la misma fórmula, lo cual permite establecer una comparación válida.

Tabla 4.1: Resultados obtenidos por diferentes sistemas.

Metric	**Acc.**	**Prec.**	**Rec.**	**F**
matrixLin	**73.8**	**74.9**	91.2	**82.2**
Mihalcea06	70.3	69.6	**97.7**	81.3
Qiu06	72.0	72.5	93.4	81.6
Zhang05	71.9	74.3	88.2	80.7
PADET08	66.3	70.6	84.5	76.9

Como se puede observar, los valores obtenidos con el sistema no resultan óptimos ante las herramientas existentes, resultado este que ya se esperaba puesto que, como se había dicho anteriormente, es sólo un sistema *baseline*.

Los valores obtenidos para el cálculo fueron:

TP: 970 → paráfrasis reales que fueron detectadas por el sistema.

TN: 175 → pares de frases que el sistema considera paráfrasis y realmente no lo son.

FP: 403 → pares de frases que el sistema considera que no son paráfrasis y realmente no lo son.

FN: 177 → paráfrasis reales que el sistema fue incapaz de detectar.

La mayor limitación que introduce este método está dada por el uso de umbrales de coincidencia, puesto que se pudo observar que las paráfrasis que no fueron detectadas por el sistema se quedaban por debajo de este límite. La mayoría alcanzaba a relacionar la mitad de los verbos, pero no lo conseguía con los complementos. Bajar el umbral no constituye una solución, porque aumentaría considerablemente el número de frases no paráfrasis consideradas como tal, por lo que se piensa trabajar en la distancia de edición de los complementos.

En cuanto al uso del recurso *FreeLing* se apreció dificultad para reconocer determinadas abreviaturas, como a.m. y p.m. lo que trae consigo que etiquete el punto intermedio como punto final de la oración (Fp). La solución de este problema también puede contribuir a mejorar los resultados del sistema, por lo que se piensa realizar un pre-procesamiento del corpus antes de pasárselo a la aplicación, eliminando aquellos puntos innecesarios que puedan introducir ruido.

Respecto al *WordNet*, se pudo observar que para las categorías utilizadas (sinónimos, hiperónimos e hipónimos) devuelve una gran cantidad de palabras repetidas, que son recogidas en el sistema y comparadas reiteradamente, por lo que se pretende chequear que la palabra no exista en la lista de salida del *WordNet* antes de adicionarla, con el fin de hacer más eficiente el sistema.

Conclusiones.

Una vez transcurridas todas las etapas de la investigación y reflejados los principales resultados en este documento se puede plantear que:

- El estudio del estado actual de la temática permitió conocer que el campo está abierto a futuras investigaciones.

- Con una nueva combinación de técnicas y herramientas de PLN, se pudo detectar paráfrasis, dando así cumplimiento al objetivo general propuesto.

- El prototipo fue satisfactoriamente implementado con el método propuesto y sus resultados comparados con otros sistemas internacionales.

- Las métricas calculadas al sistema (exactitud, precisión, cobertura y F Medida) están por debajo de los valores obtenidos con algunos sistemas existentes, por lo que hay que introducirle nuevas técnicas y herramientas para mejorarlo.

Bibliografía.

Barzilay, R. Y. K. R., Mckeown (2001) Extracting Paraphrases from a Parallel Corpus.

Callison-Burch, C. P., Koehn Y Miles, Osborne (2006) Improved Statistical Machine Translation Using Paraphrases

Fernando, S. (2007) Paraphrase Identification. *Department of Computer Science.* University of Sheffield.

Hasegawa, T. S., Sekine Y Ralph, Grishman (2005) Unsupervised Paraphrase Acquisition via Relation Discovery.

Herrera De La Cruz, J. (2005) Un Modelo Fundamentado en Análisis de Dependencias y WordNet para el Reconocimiento de Implicación Textual. *Departamento de Lenguajes y Sistemas Informáticos.* Madrid, España, Universidad Nacional de Educación a Distancia.

Ibrahim, A. B., Katz Y Jimmy, Lin (2003) Extracting Structural Paraphrases from Aligned Monolingual Corpora.

Kauchak, D. Y. R., Barzilay (2006) Paraphrasing for Automatic Evaluation.

Lepage, Y. Y. E., Denoual (2005) Automatic generation of paraphrases to be used as translation references in objective evaluation measures of machine translation.

Li, W. E. A. (2005) Automated Generalization of Phrasal Paraphrases from the Web.

Lin, D. Y. P., Pantel (2002) Discovery of Inference Rules for Question Answering. Alberta, Canadá.

Pedersen, T. S., Patwardhan Y Jason, Michelizzi (2004) Wordnet::Similarity - measuring the relatedness of concepts. *In Association for the Advancement of Artificial Intelligence.*

Poibeau , T. (2005) Automatic extraction of paraphrastic phrases from medium size corpora

Pressman, R. S. (2002) *Ingeniería de Software. Un enfoque práctico.,* España Mc Graw-Hill Interamericana

Sekine, S. (2005) Automatic Paraphrase Discovery based on Context and Keywords between NE Pairs.

Shinyama, Y. (2005) Using Repeated Patterns across Comparable Articles for Paraphrase Acquisition.

Shinyama, Y. S., Sekine Y Kiyoshi, Sudo (2002) Automatic Paraphrase Acquisition from News Articles.

Shinyama, Y. Y. S., Sekine (2003) Paraphrase Acquisition for Information Extraction.

Yamamoto, K. (2002) Acquisition of Lexical Paraphrases from Texts.

Zhang, Y. Y. J., Patrick (2006) Paraphrase Identification by Text Canonicalization