Impacto de Recursos Léxicos Manuales y Automáticos en la WSD

Javier Tejada Cárcamo[1,2], Hiram Calvo[3,4], Alexander Gelbukh[3], José Villegas[5]

[1]School of Computer Science, San Pablo Catholic University, Arequipa, Perú
[2]Sociedad Peruana de Computación, Arequipa, Perú
[3]Center for Computing Research, National Polytechnic Institute, Mexico City, 07738, México
[4]Nara Institute of Science and Technology, Takayama, Ikoma, Nara 630-0192, Japan
[5]School of System Engineering, San Agustin National University
jawitejada@hotmail.com, hcalvo@cic.ipn.mx, calvo@is.naist.jp
gelbukh@gelbukh.com, josvil@gmail.com

Abstract. Las tareas de procesamiento automático de lenguaje natural usan diferentes tipos de recursos léxicos, que pueden clasificarse en manuales y automáticos. La comunidad académica tiene la creencia que los recursos manuales proporcionan datos con mayor semántica que los automáticos. No se puede adoptar esta creencia como verdadera o falsa para cada una de tales tareas. Hemos elegido la Desambiguación de Sentidos de Palabras como tarea, y muy en particular el método propuesto en Tejada *et al*. para su resolución. En dicho método, primero se obtiene automáticamente un conjunto de *vocablos relacionados* con el contexto del vocablo ambiguo. Luego, cada uno emite un voto por un sentido de la instancia ambigua, de tal manera que el sentido con mayor cantidad de votos es el elegido. En este artículo se varía el origen de los *vocablos relacionados*, usando recursos manuales, tal como el tesauro de Mobby, y automáticos, tales como el Tesauro de Lin y los Modelos de Espacios de Palabras. De esta manera, intentamos medir el impacto de ambos tipos de recursos en la Desambiguación de Sentidos de Palabras.

Abstract. The tasks of automatic Natural Language Processing (NLP) use different types of lexical resources, which can be classified into manual and automatic. The academic community has the belief that manual resources provide data with more semantic than automatic. It is not possible adopt this belief as true or false for each of the NLP tasks. We have selected the Word Sense Disambiguation (WSD) as a task, and particularly the method proposed in Tejada et al. for its resolution. In this method, first, we have got a set of related words with the context of the ambiguous word. Then each one casts a vote for a sense of the ambiguous instance, so that the meaning with the most votes is elected. This article changes the origin of related words, using manual resources, such as Mobby thesaurus, and automatic, such as the Thesaurus of Lin and Word Space Models. In this way, we try to measure the impact of both types of resources in the WSD.

Keywords: Lenguaje natural, recursos léxicos, tesauro, desambiguación de sentidos de palabras, modelo de espacio de palabras, no supervisados.

1 Introducción

La mayoría de tareas de procesamiento de lenguaje natural usan conjuntos de vocablos relacionados semánticamente con la finalidad de solucionar una tarea en particular. Estos vocablos pueden ser obtenidos de recursos manuales o automáticos. Se tiene la creencia que los vocablos obtenidos manualmente proveen mayor calidad semántica que los obtenidos de recursos creados automáticamente. Asimismo, se puede afirmar que dicho comportamiento varía, dependiendo de la tarea de procesamiento de lenguaje natural e incluso del método usado en dicha tarea.

Se ha elegido la Desambiguación de Sentidos de Palabras (WSD por sus siglas en inglés *Word Sense Disambiguation*) por ser una tarea intermedia que no tiene aplicabilidad práctica o comercial por sí misma; sin embargo es requerida por otras áreas de Lenguaje Natural, tales como la Recuperación de Información, la Traducción Automática, Question-Answering, entre otras.

El método propuesto en Tejada *et al*. para la resolución de la ambigüedad de sentidos de palabras se basa en obtener un conjunto de palabras relacionadas semánticamente con el contexto del vocablo ambiguo. Éstas determinan el sentido de la instancia ambigua mediante un algoritmo de maximización que acumula votos para cada sentido, de tal manera que el sentido con mayor número de votos es el elegido. En este trabajo se explora diferentes recursos de información como origen de las palabras relacionadas. Estos se pueden clasificar en manuales y automáticos. En el primer caso se ha tomado el Tesauro de Mobby (construido manualmente), el cual proporciona un conjunto de términos relacionados sin ponderación para un vocablo específico. En cuanto a los recursos automáticos se ha seleccionado el Tesauro de Lin [12], el cual proporciona términos relacionados con una valor de ponderación respecto a una palabra específica. Finalmente, también se han usado los Modelos de Espacios de Palabras como una alternativa más para la generación de términos relacionados.

Un modelo de espacio de palabras (WSM por sus siglas en inglés *Word Space Model*) es una representación espacial del significado de palabras [3]. Su idea fundamental radica en que la similitud semántica puede ser representada como proximidad o lejanía en un espacio de *n* dimensiones. Cada dimensión representa un vocablo. Las palabras se van ubicando en el espacio multidimensional tomando en cuenta su distribución (frecuencia u otras medidas estadísticas) con cada vocablo del lenguaje.

La factibilidad de representar una palabra en una arquitectura multidimensional tomando en cuenta su distribución en el lenguaje está muy demostrada [4]. Es necesario tener en cuenta que la representación espacial de una palabra por si sólo no tiene sentido (ya que sólo sería un punto en el espacio multidimensional); sin embargo si otras palabras se representan en este espacio, es posible calcular similitudes y lejanías semánticas.

Tradicionalmente, la investigación realizada sobre WSM siempre ha estado enfocada hacia la creación de métodos para su construcción automática, así como diferentes técnicas para la explotación de la información que almacena este recurso. Algunos trabajos típicos en esta área son: LSA (por sus siglas en inglés *Latent*

Semantic Analysis) [7], HAL (por sus siglas en inglés *Hyperspace Anlogue to Language*) [8], RI (por sus siglas en inglés *Random Indexing*) [9], etc.

Actualmente, no se sabe con certeza el tipo de información que provee un WSM. Sólo se sabe que un conjunto de vocablos próximos tienen relación semántica (sinonimia, antonimia, etc.). Muchas de los procesos que implementan tareas de procesamiento de lenguaje natural, tales como recuperación de información, desambiguación de sentidos de palabras, traducción automática, etc., requieren conjuntos de vocablos relacionados semánticamente, los cuales son utilizados por diferentes tipos de algoritmos.

La importancia de este grupo de vocablos en las tareas de procesamiento de lenguaje natural no está en tela de juicio; sin embargo, se tiene la *creencia* que un grupo de vocablos generados manualmente (por el ser humano) tiene mejor desempeño que uno generado automáticamente (por una computadora). En esta artículo, se investiga dicha *creencia*.

Para ello, se ha planteado un método orientado a resolver la desambiguación de sentidos de palabras. Este método toma el contexto del vocablo ambiguo para obtener un conjunto de vocablos relacionados semánticamente. Cada miembro del conjunto emite un voto por cada uno de los sentidos del vocablo ambiguo, de tal forma que el sentido con mayor número de votos es el elegido como el sentido de la instancia ambigua. El conjunto de vocablos relacionados será proporcionado por diferentes fuentes de información: Un WSM creado automáticamente, el Tesauro de Mobby creado manualmente y el Tesauro de Lin creado automáticamente.

El resto del documento se organiza de la siguiente manera: En la sección 2 se describe las tres fuentes de información que se han utilizado: WSM, Tesauro de Mobby y Tesauro de Lin. En la sección 3 se describe el método de desambiguación planteado. En la sección 4 se muestran los resultados obtenidos. En la sección 5 se presentan las conclusiones.

2 Orígenes de Información

En nuestro trabajo, se denomina origen de información a un recurso léxico (creado manual o automáticamente) que almacena información estructurada o sin estructurar, la que puede ser explotada para obtener un conjunto de términos relacionados. En esta sección se describen tres orígenes de información: WSM, Tesauro de Mobby y Tesauro de Lin.

2.1 Modelo de Espacio de Palabras (WSM)

WSM es una representación espacial del significado de palabras. Su idea fundamental radica en que la similitud semántica puede ser representada como proximidad o cercanía en un espacio de *n* dimensiones, donde *n* puede ser un número entero entre 1 y otro muy grande (ver fig. 1). Al respecto, Schütze afirmó: “La similitud de vectores es la única información presente en este modelo, de tal manera que palabras que se

relacionan semánticamente están cerca y aquellas que no se relacionan semánticamente están lejos".[19]

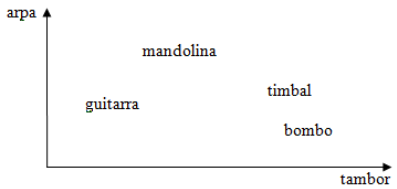

Fig. 1. *WSM de dos dimensiones*

En la fig. 1, cada una de sus dimensiones representa un vocablo: El eje *x* al vocablo *tambor* y el eje *y* al vocablo *arpa*. Los demás vocablos (*guitarra*, *mandolina*, *timbal*, *bombo*) se van posicionando en el espacio dependiendo de la similitud que tienen con *arpa* y *tambor*. Se puede observar que las palabras tienden a formar grupos semánticos. Dicha agrupación, depende de la distribución que éstas presentan en el lenguaje; por ejemplo en un corpus de texto los vocablos que se usan con *timbal*, *bombo* y *tambor* suelen ser los mismos.

Es necesario tener en cuenta que la representación espacial de una palabra por sí sólo no tiene sentido (ya que sólo sería un punto en el espacio multidimensional); sin embargo si otras palabras se representan en este espacio, es posible calcular similitudes y lejanías semánticas.

2.1.1 Procesamiento del Corpus

En esta sección se explica el tratamiento del corpus de entrenamiento, el cual varía dependiendo del tipo de modelo de espacio de palabras que se desea construir. La diferencia entre WSM sintagmático y paradigmático radica en el conjunto de *términos relacionados* que proporcionan. Si consultáramos a un WSM paradigmático los términos más similares al vocablo *universidad*, serían *academia*, *escuela*, *colegio*, etc., mientras que un WSM sintagmático hubiera proporcionado vocablos como: *privada*, *nacional*, *tecnológica*, etc.

Dos vocablos relacionados paradigmáticamente no co-ocurren entre ellos; sin embargo, los vocablos con los que co-ocurren suelen ser los mismos. Por ejemplo, adjetivos diferentes que modifican al mismo sustantivo, como *buena noticia* y *mala noticia*, o *querido padre* y *querida madre*. De dichas co-ocurrencias se puede determinar que los vocablos *malos y buenos*, *padre* y *madre* presentan una relación paradigmática o de sustitución.

Existen tres parámetros que influyen en la obtención de este tipo de relaciones: el tamaño de la ventana contextual, la posición de las palabras dentro de dicha ventana y la dirección en la que la región de contexto se extiende (hacia delante o atrás). Para ilustrar mejor este punto imaginemos dos secuencias de palabras:

bla bla bla *blo* bli bli bli
bla bla bla *ble* bli bli bli

Es fácil notar que las palabras *blo* y *ble* presentan una relación paradigmática ya que las palabras anteriores y posteriores de ambas son las mismas. En este caso no importa si tomamos una ventana de tamaño 1+1 (un vocablo a la izquierda y uno a la derecha), 2+2 e incluso 3+3, ya que en este caso particular, cada ventana confirma la relación paradigmática entre *blo* y *ble*.

Las ventanas contextuales pueden ser estáticas y dinámicas, es decir, con un número de vocablos fijos o variables a la derecha e izquierda de la palabra en cuestión. Actualmente los investigadores prefieren ventanas estáticas.
Ahora supongamos el siguiente contexto:

bla *blo* e bli → *blo*: (bla) + (0 bli) → *blo*: (1) + (0 1)
bla *ble* h bli → *ble*: (bla) + (0 bli) → *ble*: (1) + (0 1)

Quizás cueste un poco más darse cuenta que las palabras *blo* y *ble* presentan una relación paradigmática, ya que los vocablos de la derecha (*e* y *h*) de cada una no son las mismas. Esto se representa mediante notación binaria:

blo: (1) + (0 1)
blo: (1) + (0 1)

Existen diferentes maneras de asignar peso a los vocablos de una ventana, pero las más comunes son la binaria y la asignación de mayor peso a aquellos que se encuentran más cercanos a la palabra en cuestión.

2.1.2 Construcción de la Matriz

Ambos tipos de WSM son representados mediante una matriz, en la cual una fila representa a un vocablo existente en el corpus de entrenamiento, mientras que el número de columnas varía dependiendo del tipo de WSM que se construye:

- En el caso de los paradigmáticos se crea una columna por cada vocablo existente en el corpus, por ende una matriz paradigmática tiene dimensiones $n \times n$.
- En el caso de los sintagmáticos, el corpus de texto se divide en *regiones contextuales* del mismo tamaño (de 10 vocablos o más grandes, como 150). El número de columnas de esta matriz depende de la cantidad de *regiones contextuales d* en la que se divide el corpus de entrenamiento, por ende una matriz sintagmática tienen dimensiones $n \times d$.

Por ejemplo, dada la oración: *el gato se comió al ratón en el tejado*. La matriz paradigmática resultante tomando en cuenta dicha frase y además, una ventana contextual de un solo vocablo a la izquierda y otro a la derecha se puede apreciar en la Tabla 1.

Tabla 1. Matriz paradigmática

Vocablo	gato	Comer	ratón	tejado
Gato	0	1	0	0
Comer	0	0	1	0
Ratón	0	0	0	1
Tejado	0	0	0	0

El valor que se establece en las celdas de la matriz es un peso de ponderación, que se asigna tomando en cuenta diferentes esquemas estadísticos. En el ejemplo se usa un esquema binario, donde 1 significa que existe una co-ocurrencia entre la fila *i* y la columna *j* y 0 que dicha co-ocurrencia no existe.

En el método propuesto, sólo se toman en cuenta vocablos de contenido (*content words* en inglés), tales como sustantivos, adjetivos y verbos, desechando las palabras de parada (*stop words*), lo cual permite reducir las dimensiones de la matriz. Asimismo, se usan los *lemas* de los vocablos seleccionados.

2.1.3 Esquema de Ponderación

El esquema de ponderación hace referencia a un valor que denota la afinidad o relación semántica entre la fila *i* y la columna *j*. En nuestro método este valor inicialmente es la frecuencia de correlación entre dos vocablos en el corpus (en el caso de los paradigmáticos) o la cantidad de veces que sucede un vocablo en una región contextual (en el caso de los sintagmáticos).

En este método se utiliza otro tipo de ponderación, conocido como el esquema TF-IDF (por sus siglas en inglés *Term Frecuency - Inverse Document Frecuency*), el cual generalmente se aplica a tareas de clasificación y similitud de documentos [10]. En este esquema, cada documento es representado por un vector cuyo número de dimensiones corresponde a la cantidad de *vocablos* que existen en él.

En nuestro método, el valor en cada celda de la matriz está determinado por un peso *w* (ver ecuación 3), el cual se calcula como el producto del TF (ver ecuación 1) e IDF (ver ecuación 2). El peso $w_{(i,j)}$ determina el grado de relación semántica entre el *vocablo i* (la fila) y palabra *j* o sección *j* (la columna). El TF muestra la importancia de un *vocablo* respecto a la palabra que modifica o a la sección en la que se encuentra.

Por la tanto, el peso de la relación aumenta si el *vocablo* aparece más a menudo con dicha palabra o sección. El IDF denota la importancia de un *vocablo* respecto al resto de palabras del corpus, de tal manera que su peso disminuye si aparece más a menudo con los demás *vocablos* o *secciones* del corpus, y aumenta cuando aparece con la menor cantidad de estos, ya que los *vocablos* o *secciones* muy frecuentes discriminan poco a la hora de representar al *vocablo* mediante un vector.

$$f_{i,j} = \frac{freq_{i,j}}{max\ freq_{l,j}} \quad . \qquad (1)$$

$$idf_i = \log \frac{N}{n_i} \quad (2)$$

$$w_i = f_{i,j} \times idf_i \quad (3)$$

En las ecuaciones anteriores, i representa a la i-ésima fila (vocablos) y j (puede ser vocablos o secciones) representa a las j-ésima columna de nuestra matriz. La freqi,j es la frecuencia entre i y j, max freql,j es la más alta de cualquier vocablo i en una sección o vocablo j. N es el número de vocablos del corpus tomados en cuenta en la construcción de la matriz, ni es el número de vocablos con los que ocurre j, y wi es el peso final. El peso w que se calcula para todas las dimensiones de un vocablo i, forman un vector que se almacena en el WSM (ver ecuación 4).

$$\vec{V}(vocablo_i) = \{(dim_1, w_1), ..., (dim_n, w_n)\} \quad (4)$$

$\vec{V}$ *(vocablo$_i$)* es el vector que representa al vocablo *i* con respecto a la totalidad de vocablos o secciones del corpus, *dim$_n$* cada una de las dimensiones del WSM (el número de dimensiones es el número de vocablos o secciones del corpus de entrenamiento) y *w$_n$* es el peso asignado a *dim$_n$*. Muchos pesos son 0 (cero), lo que denota la inexistencia de una correlación entre el vocablo *i* y el vocablo o sección *j*. Realmente la cantidad de ceros en el sistema es elevada (*data sparseness*), lo que confirma la Ley de Zipf [11], que indica que sólo unas pocas palabras en el lenguaje (*non content words*) se comportan *promiscuamente*, es decir se relacionan con muchas palabras.

Tomando en cuenta el ejemplo anterior, se puede determinar la lista de co-ocurrencias de un vocablo; por ejemplo, en la oración mostrada en la Tabla 1, el vocablo *comer*, está representado por la tupla (1,0,1), la cual se convierte en un vector $\vec{V} = (x_1, x_2, x_3)$ donde el número de dimensiones es el número de vocablos en el sistema (en nuestro ejemplo son tres), y los valores (1,0,1) son las coordenadas de posicionamiento en el espacio vectorial. Representando vectorialmente las propiedades de distribución de una palabra podemos pasar a una representación geométrica de dicho vocablo.

2.2 Tesauro de Mobby

El tesauro de Mobby está considerado como una de las fuentes de información más grandes y coherentes que existen para el idioma inglés. Este tesauro es manual, es decir, creado por el ser humano. La segunda edición de dicho tesauro ha mejorado mucho con respecto a la primera, añadiendo mas de 5000 palabras principales, que en su totalidad superan los 30000 vocablos.

Asimismo, se le añadió también más de un millón de sinónimos y *términos relacionados*, que en su totalidad superan los 2.5 millones de sinónimos y *términos relacionados*. La fig. 2 muestra parte de una *entrada* del tesauro.

demon: baba yaga, lilith, mafioso, satan, young turk, addict, afreet, ape-man, atua, barghest, beast, beldam, berserk, berserker, bomber, brute,bug, cacodemon, collector, daemon, daeva, damned spirits, demonkind, demons, denizens of hell, devil, devil incarnate, dragon, dybbuk, eager beaver, energumen, enthusiast, evil genius, evil spirit, evil spirits, faddist,

2.3 Tesauro de Lin

Este tesauro, creado automáticamente, fue uno de los primeros recursos léxicos que se construyó usando una técnica de lenguaje natural, conocida como *Bootstrapping semantics* [12]. Para ello, primero se define una medida de similitud que se basa en los patrones de distribución de las palabras. Esta medida permite construir un tesauro usando un corpus parseado, el cual consta de 64 millones de palabras, las cuales fueron tomadas de diversos corpus tales como el Wall Street Journal (24 millones de palabras), San Jose Mercury (21 millones de palabras) y AP Newswire (19 millones de palabras). Una vez que el corpus fue parseado se extrajo 56.5 millones de tripletas de dependencia (de las cuales 8.7 millones fueron únicas). En el corpus parseado hubo 5469 sustantivos, 2173 verbos, y 2632 adjetivos/adverbios que ocurrieron al menos 100 veces.

Finalmente, se computó la similitud semántica de cada par de vocablos entre sustantivos, verbos, adjetivos y adverbios. Para cada una de estas palabras se creó una entrada en el tesauro, que contiene una *lista ponderada* de términos similares.

3 Método de Desambiguación

McCarthy et al. [2] propuso un algoritmo no supervisado para encontrar el sentido predominante de una palabra, sin tomar en cuenta la frecuencia de WordNet/SemCor. Se basa en el tesauro de Lin [13] para determinar la similitud entre palabras. Para una instancia ambigua *w* se considera todas las palabras *u* relacionadas a *w* en el tesauro de Lin. Usando un algoritmo de maximización cada vocablo *u* emite un voto por un sentido ws_i de *w*, de tal manera que el sentido con mayor cantidad de votos es el elegido como el sentido predominante de *w* y, en particular, esta heurística se puede utilizar en WSD.

Este método de desambiguación no utiliza el contexto del vocablo ambiguo para obtener un conjunto de términos relacionados, por ende siempre se le asignará el mismo sentido a una instancia ambigua. El sentido elegido como predominante para una palabra depende únicamente de los corpus utilizados para construir el tesauro.

En el método de desambiguación planteado en Tejada *et al*. [1], un WSM previamente construido proporciona una lista de vocablos relacionados con el contexto de la palabra ambigua. Al igual que McCarthy *et*. *al*. cada uno vota por un sentido de la palabra ambigua (mediante su algoritmo de maximización); pero en este caso se elige el sentido de la instancia ambigua en un contexto específico (ver fig. 3).

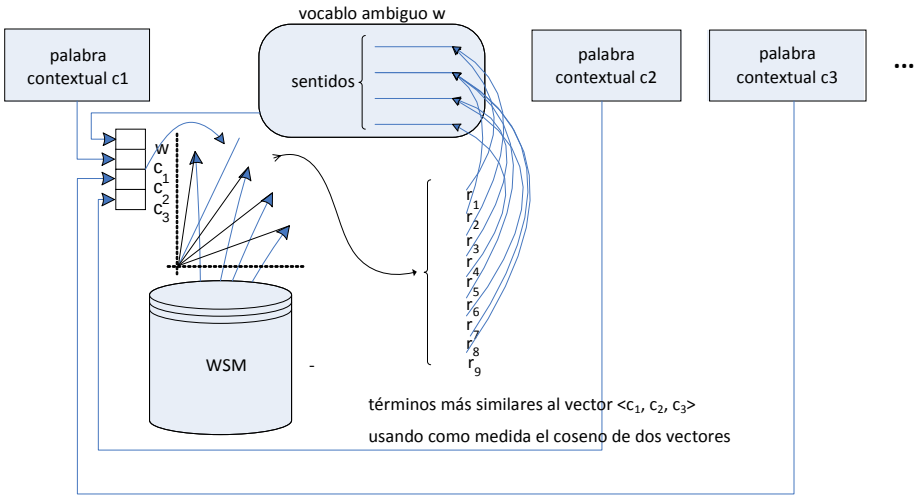

Fig. 3. *Método de desambiguación planteado*

3.1 Algoritmo de maximización

Este algoritmo permite que cada *vocablo ponderado* emita un voto para cada sentido de la palabra polisémica. El sentido con el puntaje más alto es seleccionado. Las ecuaciones siguientes muestran cómo la *lista de vocablos ponderados* acumula una puntuación para un sentido.

$$Weight(w_{si}) = \sum_{t_i \in LT_w} P(w, t_j) \times P_{Norm}(w_{si}). \quad (5)$$

$$P_{Norm}(w_{si}) = \frac{pswn(w_{si}, t_j)}{\sum_{w_{si} \in sentidos(w)} pswn(w_{si}, t_j)}. \quad (6)$$

$$pwsn(w_{si}, t_j) = max_{s_x \in sentidos(t_j)}(pswn(w_{si}, s_x)). \quad (7)$$

En estas ecuaciones, *w* es la palabra ambigua, w_{si} es cada uno de los sentidos de *w*, LT_w es la lista ponderada de los términos y t_j es cada término. $P(w, t_j)$ representa la similitud semántica entre w y t_j. Este valor se ha calculado en el WSM. P_{Norm} representa cómo vamos a normalizar el peso de w_{si} utilizando todos los sentidos de *w* y el t_j actual.

La función *pswn* devuelve el sentido de una palabra que tiene la mayor similitud semántica con un sentido particular. Por ejemplo, *pswn* (w_{si}, t_j) compara todos los sentidos de los *vocablos relacionados* t_j con w_{si} y obtiene el sentido de t_j que tiene

más similitud semántica con respecto a w_{si}. La fig. 4 muestra el algoritmo de maximización.

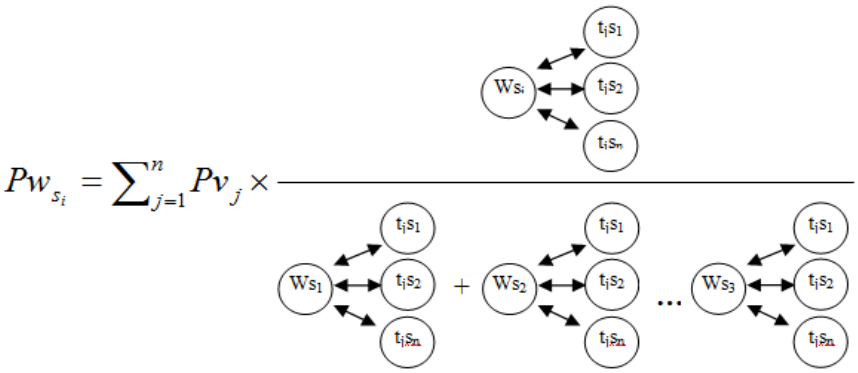

Fig. 4. Algoritmo de maximización

Nosotros usamos WordNet::Similarity presentado en [14] para medir la similitud semántica entre dos sentidos. Se trata de un conjunto de librerías que implementan medidas de similitud y relación semántica de WordNet. Hemos estado utilizando el Adapted Lesk (Extended Gloss Overlap) como medida de similitud [15], Pero es posible utilizar otras medidas semánticas como: JCN [17] Resnik [18], Lin [19], etc.

3.2 Recuperación de Términos Relacionados

En esta etapa se obtiene un conjunto de *vocablos relacionados* con la instancia ambigua. Este proceso de selección toma en cuenta el contexto del vocablo ambiguo. A continuación se describe dicho proceso en cada uno de los orígenes de información descritos en la sección 2.

3.2.1 Modelo de Espacio de Palabras

El WSM almacena vectores de dimensionalidad *n*. Para determinar los *vocablos relacionados* con la instancia ambigua, se crea un vector en el que se represente su contexto y se le compara con los existentes en el WSM. Los vectores más similares determinan los *vocablos relacionados*. La dimensionalidad del vector ambiguo es *n* y el peso de ponderación en cada dimensión, depende de los vocablos de su contexto.

En la Tabla 2, se muestran los vectores de un WSM. Las columnas d_i son las dimensiones de los vectores, y las filas son los vocablos del sistema. *w* es el vocablo ambiguo y su vector indica que los vocablos de su contexto son d_2 y d_4. La columna *similitud*, ordenada descendentemente, indica que el vocablo r_1 es más similar a *w*, que el vocablo r_n.

Table 2. Vectores en un WSM

	d_1	d_2	d_3	d_4	d_5	d_6	…	d_n	Similitud
w	0	1	0	1	0	0	…	0	
r_1	1	2	6	5	2	2	…	0	0.99
r_2	0	5	4	0	1	1	…	3	0.92
r_3	4	4	3	3	1	1	…	0	0.83
…								…	
r_n	0	4	0	5	2	2	…	4	0.68
...								…	

Existen varias maneras de computar la similitud de dos vectores y en nuestro método se determina mediante el valor del coseno del ángulo que forman expresado por el cociente entre el producto *punto* y el producto *cruz* (ver ecuación 8).

$$cos\_measure(\vec{w}, \vec{r}_i) = \frac{\vec{w} \cdot \vec{r}_i}{|\vec{w}| \times |\vec{r}_i|} = \frac{\sum_{j=1}^{n} w_j \times r_{i,j}}{\sqrt{\sum_{j=1}^{n} (w_j)^2} \times \sqrt{\sum_{j=1}^{n} (r_{i,j})^2}} \quad \textbf{(8)}$$

3.2.2 Tesauro de Mobby

Una *entrada* o vocablo en el tesauro de Mobby es un conjunto de palabras listadas alfabéticamente, que carecen de algún valor o peso de ponderación que determine el grado de similitud o relación semántica con dicha *entrada*. (ver sección 2.2). Para ello, se ha creado un método que tome en cuenta el contexto en la selección de *términos relacionados* proporcionados por dicho tesauro, el cual se detalla a continuación:

1. Obtener el contexto del vocablo ambiguo representado en: C(w) ={c_1, c_2,…, c_i}, donde *w* es la instancia ambigua y c_i es cada uno de los vocablos de su contexto.
2. Obtener el conjunto de palabras listadas alfabéticamente para el vocablo ambiguo, que proporciona este tesauro, representado en: S(w) ={sw_1, sw_2,…, sw_i}, donde *w* es el vocablo ambiguo y sw_i , cada una de las palabras que proporciona el tesauro.
3. Discriminar los vocablos obtenidos en S(w) tomando en cuenta C(w). Para ello, se construye un WSM (ver Tabla 3) donde cada columna es una palabra de C(w) y cada fila, un miembro de S(w). Se incluye el vocablo ambiguo *w* como una fila más.

Tabla 3. WSM usando tesauro de Mobby

	c_1	c_2	c_3	c_i
sw_1	(c_1, sw_1)	(c_2, sw_1)	(c_3, sw_1)	(c_i, sw_1)

sw_2	(c_1, sw_2)	(c_2, sw_2)	(c_3, sw_2)	(c_i, sw_2)
sw_3	(c_1, sw_3)	(c_2, sw_3)	(c_3, sw_3)	(c_i, sw_3)
sw_i	(c_1, sw_i)	(c_2, sw_i)	(c_3, sw_i)	(c_i, sw_i)
w	(c_1, w)	(c_2, w)	(c_3, w)	(c_i, w)

Cada par ordenado representa la máxima similitud semántica entre c_i y sw_i o en su caso, *w*. Este valor se computa comparando todos los sentidos de c_i con los de sw_i (ver figura 5) y se elige el más alto. La comparación de dos sentidos se realiza mediante el paquete WordNet::Similarity usando la medida de similitud semántica *extended gloss overlap* (una adaptación de algoritmo de Lesk) [15]

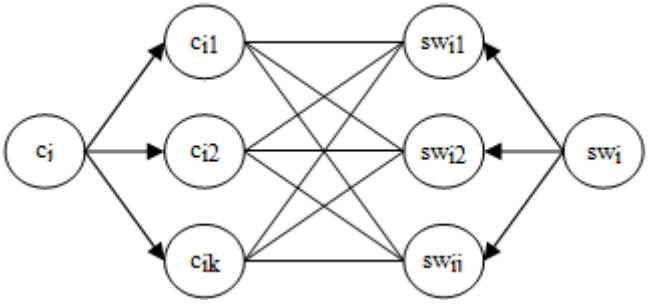

Fig. 5. *Comparación de los sentidos de dos vocablos.*

4. Finalmente, se compara el vector de *w*, con cada uno de los vectores de sw_i usando el coseno de dos vectores obteniendo de esta manera los vocablos más relacionados.

3.2.3 Tesauro de Lin

A diferencia del tesauro de Mobby, el tesauro de Lin proporciona un valor que cuantifica la similitud semántica para cada palabra correspondiente a una *entrada*. Este valor hace referencia a las palabras que usualmente se usan juntas. Por ejemplo, en la oración *Las estrellas del firmamento lucen más hermosas que nunca*, al consultar este tesauro por las palabras más relacionadas con *estrella*, obtendríamos *cantante*, *música*, *glamour*, mientras las menos relacionadas serían *planeta*, *astro*, *universo*.

En el contexto en el que aparece *firmamento*, los vocablos *planeta*, *astro*, *universo* tienen mayor relación que *cantante*, *música*, *glamour*. Por esto se presenta un método que usando este tesauro, obtenga los vocablos más relacionados a una instancia ambigua tomando en cuenta su contexto. Dicho método se detalla a continuación:

1. Obtener el contexto del vocablo ambiguo representado en: $C(w) = \{c_1, c_2, \ldots, c_i\}$, donde *w* es la instancia ambigua y c_i es cada uno de los vocablos de su contexto.
2. Obtener el conjunto de palabras relacionadas que proporciona este tesauro para la instancia ambigua, representado en: $S(w) = \{(sw_1, p_1), (sw_2, p_2), \ldots, (sw_i, p_i)\}$, donde

w es el vocablo ambiguo, sw_i cada vocablo similar que proporciona el tesauro de Lin y p_i es el peso que se le asigna a la relación semántica entre sw_i y *w*.

3. Discriminar los vocablos obtenidos en S(w) tomando en cuenta C(w). Las columnas representan cada uno de los miembros del contexto y las filas los vocablos similares. Asimismo, se incluye *w* como una fila más del WSM (ver Tabla 4), asignando la máxima similitud semántica entre *w* y cada uno de los miembros de su contexto. El peso que cuantifica la relación semántica entre c_i y sw_i es el proporcionado por este tesauro.

Tabla 4. WSM usando tesauro de Lin

	c_1	c_2	c_3	c_i
sw_1	(c_1, sw_1)	(c_2, sw_1)	(c_3, sw_1)	(c_i, sw_1)
sw_2	(c_1, sw_2)	(c_2, sw_2)	(c_3, sw_2)	(c_i, sw_2)
sw_3	(c_1, sw_3)	(c_2, sw_3)	(c_3, sw_3)	(c_i, sw_3)
sw_i	(c_1, sw_i)	(c_2, sw_i)	(c_3, sw_i)	(c_i, sw_i)
w	(c_1, w)	(c_2, w)	(c_3, w)	(c_i, w)

4. Finalmente, se compara el vector de *w*, con cada uno de los vectores de sw_i usando el coseno de dos vectores obteniendo de esta manera los vocablos más relacionados.

4 Experimentos y Resultados

El método de desambiguación utilizado ya ha sido publicado en uno de nuestros artículos anteriores [1]. Éste método superó los resultados obtenidos por el mejor método desambiguación existente hasta el momento que fue publicado por Diana Mc. Carthy *et. al*. [2]. Dichos resultados se muestran en la Tabla 5.

Tabla 5. Resultados de SENSEVAL-2

Orden	Sistema	Tipo	Precision	Recall	Cobertura (%)
1	SMUaw	supervisado	0.69	0.69	100
	Mc.Carthy *et al.*	no supervisado	0.64	0.63	100
2	CNTS-Antwerp	supervisado	0.636	0.636	100
3	Sinequa-LIA - HMM	supervisado	0.618	0.618	100
–	*WordNet most frequent sense*	supervisado	0.605	0.605	100
4	UNED - AW-U2	no supervisado	0.575	0.569	98.9
5	UNED - AW-U	no supervisado	0.556	0.55	98.9
6	UCLA - gchao2	supervisado	0.475	0.454	95.55
7	UCLA - gchao3	supervisado	0.474	0.453	95.55

9	CL Research - DIMAP (R)	no supervisado	0.451	0.451	100
10	UCLA - gchao	supervisado	0.5	0.449	89.72

En los experimentos realizados hemos usado como corpus de evaluación los sustantivos de SENSEVAL-2. El Modelo de Espacio de Palabras ha sido creado tomando British National Corpus (100 millones de palabras) como corpus de entrenamiento.

Tabla 6. Resultados de WSD usando diferentes fuentes de información

		WSM	MOBBY	LIN
Número de vocablos relacionados	10	61.62	58.9	61.82
	20	63.46	56.04	60.47
	30	64	57.62	60.86
	40	64	**60.16**	61.59
	50	64	58.26	64.46
	60	66.43	57.62	63.95
	70	68.66	58.57	64.89
	100	67.22	57.62	65.05
	200	69.08	57.31	**65.13**
	500	**70.55**	58.26	63.97
	1000	65.12	55.09	59.92
	2000	62.1	53.8	63.66
Promedio		65.52	57.44	62.98

La Tabla 6 muestra los resultados obtenidos por el método de desambiguación cuando el grupo de *vocablos relacionados* para una instancia ambigua son proporcionados por diferentes fuentes de información: WSM, Tesauro de Mobby y Tesauro de Lin. En dicha tabla se puede observar que los mejores resultados se consigue con el WSM, seguido por el tesauro de Lin y luego el tesauro de Mobby. Asimismo se aprecia que los mejores resultados se obtuvieron con 40, 200 y 500 vocablos relacionados. Finalmente en la fig. 6 se puede ver con mayor claridad como los vocablos suministrador por el WSM tienen mejores resultados en el método de WSD.

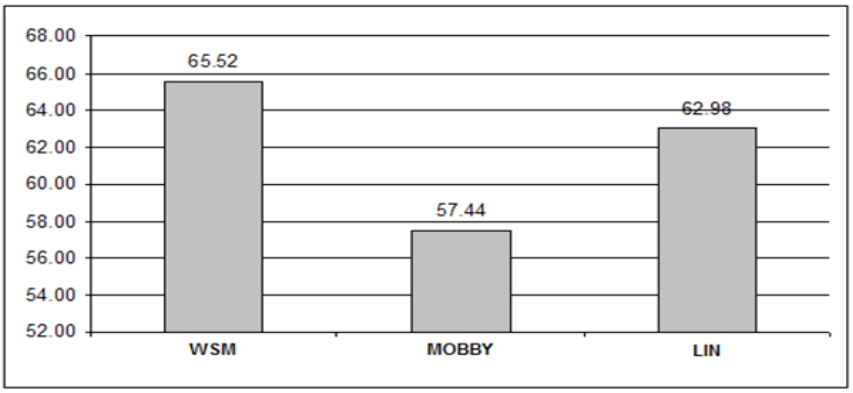

Fig. 6. Resultados de WSD usando diferentes fuentes de información

5 Conclusiones y Trabajo Futuro

Los resultados obtenidos nos permiten concluir que un grupo de vocablos suministrados por recursos construidos automáticamente tienen mejor rendimiento que aquellos que proporcionan recursos construidos manualmente cuando son utilizados en el método de desambiguación planteado. Los grupos de *vocablos relacionados* obtenidos del Modelo de Espacio de Palabras y del Tesauro de Lin obtienen mejores resultados que los *vocablos relacionados* obtenidos del Tesauro de Mobby. No podemos afirmar que este comportamiento sea el mismo para cualquier aplicación del lenguaje natural, sin embargo es un indicador importante a tomar en cuenta cuando se necesite elegir un conjunto de *vocablos relacionados*.

Nosotros creemos que este comportamiento se debe a que cuando un proceso no supervisado construye un recurso léxico, como el WSM o el Tesauro de Lin, de términos similares o relacionados semánticamente sólo toma en cuenta la información existente dentro del texto que se procesa; sin embargo el grupo de lexicógrafos que crean un recurso manual posiblemente se ven influenciados por un mundo pragmático que los métodos no supervisados no detectan.

Asimismo, otra razón que puede explicar la prevalencia de los *vocablos relacionados* obtenido de recursos automáticos sobre los manuales, es la dimensión de los recursos de entrenamiento que se utilizan en los métodos no supervisados. Por ejemplo para la construcción del WSM se ha utilizado un corpus de 100 millones de palabras. Si se compara la cantidad de vocablos relacionados que proveen los recursos creados automáticamente con aquellos que proveen los recursos manuales, la diferencia es muy notoria. Sea como fuere, creemos que la tendencia en las aplicaciones de lenguaje natural es utilizar métodos no supervisados para la creación de recursos léxicos cuyo objetivo sea proveer términos relacionados semánticamente.

6 Referencias

1. Tejada, J., Gelbukh A., Calvo, H. An Innovative Two-Stage WSD Unsupervised Method. SEPLN Journal 40, March 2008.
2. McCarthy, D. and R. Navigli (2009) The English Lexical Substitution Task, To appear in Language Resources and Evaluation 43 (2) Special Issue on Computational Semantic Analysis of Language: SemEval-2007 and Beyond, Agirre, E., Marquez, L. and Wicentowksi, R. (Eds). pp 139-159 Springer, 2009.
3. Sahlgren, Magnus. The Word-Space Model Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces, Ph.D. dissertation, Department of Linguistics, Stockholm University, 2006.
4. Schütze, Hinrich: Dimensions of meaning. Proceedings of Supercomputing'92. IEEE Computer Society Press, Los Alamitos, California. 787-796 (1992)
5. Landauer, T., & Dumais, S.: A solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. Psychological Review, 104 (2), 211{240 (1997)
6. Lund, K., Burgess, C., & Atchley, R.: Semantic and associative priming in high-dimensional semantic space. In Proceedings of the 17th Annual Conference of the Cognitive Science Society, CogSci'95 (pp. 660{665). Erlbaum (1995)
7. Landauer, T., & Dumais, S. (1997). A solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. Psychological Review, 104 (2), 211{240.
8. Lund, K., Burgess, C., & Atchley, R. (1995). Semantic and associative priming in high-dimensional semantic space. In Proceedings of the 17th Annual Conference of the Cognitive Science Society, CogSci'95 (pp. 660{665). Erlbaum.
9. Sahlgren, M. (2005). An introduction to random indexing. In H. Witschel (Ed.), Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering, TKE'05,Copenhagen, Denmark, august 16, 2005 (Vol. 87).
10. Pedersen, T.; Patwardhan, S. and Michelizzi (2004).WordNet::Similarity - Measuring the Relatedness of Concepts. Appears in the Proceedings of Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-04), May 3-5, 2004, Boston, MA (Demonstration System).
11. Zipf, G.(1949). Human behavior and the principle of least-e_ort. Cambridge, MA: Addison-Wesley.
12. Lin, D. (1998). Automatic retrieval and clustering of similar words. Proceedings of C I G C -. pp. 768-774. Montreal, Canada.
13. Lin, D. (1997). Using syntactic dependency as a local context to resolve word sense ambiguity. In Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics, pages 64–71, Madrid, July 1997.
14. Pedersen, T.; Patwardhan, S. and Michelizzi (2004).WordNet::Similarity - Measuring the Relatedness of Concepts. Appears in the Proceedings of Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-04), May 3-5, 2004, Boston, MA (Demonstration System).

15. Lesk, Michael (1986). Automated Sense Disambiguation Using Machine-readable Dictionaries: How to Tell a Pine one from an Ice Cream Cone. Proceedings of the 1986 SIGDOC Conference, Toronto, Canada, June 1986, 24-26.
16. Jiang, J. and Conrath D. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In Proceedings on International Conference on Research in Computational Linguistics, Taiwan, 1997.
17. Resnik, Philip, Using information content to evaluate semantic similarity in a taxonomy, Montreal. Proceedings of the 14th International Joint Conference on Artificial Intelligence, 1995, pp. 448-453.
18. Lin, D. (1998). Automatic retrieval and clustering of similar words. Proceedings of C I G C -. pp. 768-774. Montreal, Canada.
19. Schütze, Hinrich (1993). Word space. In Hanson, Stephen J.; Cowan, Jack D. and Giles, C. Lee (Eds.) Advances in Neural Information Processing Systems 5, Morgan Kauffman, San Mateo, California, 5, 895-902.