Procesamiento del Lenguaje Natural, num. 30 (2003), pp.99-106 recibido 1-03-2003; aceptado 20-05-2003

Manejando la variación morfológica y léxica en la Recuperación de Información Textual*

Jesús Vilares Ferro y **Fco. Mario Barcala Rodríguez**
Depto. de Computación, Univ. de A Coruña
Campus de Elviña s/n, 15071 - A Coruña
jvilares@udc.es, barcala@dc.fi.udc.es

Santiago Fernández Lanza
Depto. de Lógica y Filosofía Moral
Univ. de Santiago de Compostela
Campus Sur s/n
15782 - Santiago de Compostela
sflanza@usc.es

Juan Otero Pombo
Depto. de Informática
Univ. de Vigo
Campus As Lagoas s/n
32004 - Ourense
jop@uvigo.es

Resumen: En este artículo presentamos una serie de técnicas de Procesamiento de Lenguaje Natural aplicadas la normalización de términos en Recuperación de Información Textual. El objetivo de dichas técnicas es el tratamiento de los fenómenos de variación lingüística morfológica y léxica. En concreto explorará la utilización de la lematización, su empleo combinado con el *stemming* y la expansión de consultas mediante umbrales de sinonimia.
Palabras clave: Recuperación de Información Textual, Técnicas de normalización, Expansión de consultas.

Abstract: This article describes several Natural Language Processing techniques applied to term conflation in Text Retrieval. The aim of these techniques is to manage both morphological and lexical linguistic variation. To be exact, we will study the application of lemmatization, its combined employment with stemming, and query expansion through synonymy thresholds.
Keywords: Text Retrieval, Conflation techniques, Query expansion.

1. *Introducción*

En el ámbito de la Recuperación de Información Textual (TR), la tarea de decidir sobre la relevancia o no de un documento respecto a una consulta puede ser vista como un problema de Procesamiento de Lenguaje Natural (NLP), dado que la información deseada está codificada en forma de texto. En el caso del español, la necesidad de aplicar tales técnicas para solventar la *variación lingüística* existente es mayor, debido a la abundancia de fenómenos morfológicos y léxicos.

Por otra parte, los continuos avances en el campo del NLP han desembocado en el desarrollo de una nueva generación de herramientas, más eficientes, robustas y precisas. Esto, junto a la cada vez mayor potencia de los ordenadores, hace posible que las técnicas de NLP sean ya aplicables en ámbitos prácticos.

Presentaremos a continuación una serie de herramientas de NLP diseñadas para tratar la variación lingüística del español tanto en su nivel morfológico como léxico. Dichas herramientas emplean tecnología de estado finito para hacer viable su empleo a gran escala.

Este artículo se organiza de la siguiente manera. La sección 2 describe nuestro tratamiento de la variación morfológica, mientras que la sección 3 aborda el manejo de la variación léxica. En la sección 4 presentamos nuestro módulo de tratamiento de frases en mayúscula. Los experimentos llevados a cabo para estas técnicas se muestran en la sección 5. Finalmente, nuestras conclusiones y trabajo futuro se explican en la sección 6.

* Parcialmente financiado por el Plan Nacional de Investigación Científica, Desarrollo e Innovación Tecnológica (TIC2000-0370-C02-01), Ministerio de Ciencia y Tecnología (HP2001-0044), becas FPU de la Secretaría de Estado de Educación y Universidades, Xunta de Galicia (PGIDT01PXI10506PN, PGIDIT02PXIB30501PR y PGIDIT02SIN01E) y Universidade da Coruña.

ISSN 1135-5948

2. La variación morfológica

La *variación morfológica* hace referencia a los cambios a nivel de la estructura interna de la palabra, pudiendo considerarse dos niveles diferentes, la *variación flexiva* y la *derivativa*. La primera contempla las modificaciones resultado de la sintaxis tales como género, número, tiempo, etc. Dichos cambios nunca alteran la categoría gramatical de la palabra, y apenas afectan a su significado. Sin embargo los cambios semánticos causados por la variación derivativa son mayores, y están frecuentemente acompañados por cambios en la categoría sintáctica (p.ej. *perro - perrera*).

En inglés, suele emplearse un *stemmer* para eliminar estos fenómenos variacionales (Lennon, Pierce, y Willett, 1981). Se trata de una herramienta muy sencilla desde el punto de vista lingüístico y computacional, que trata de obtener la raíz gramatical de una palabra mediante la eliminación de sufijos en base a una lista. Los resultados obtenidos para el caso del inglés son satisfactorios gracias a que su morfología, sobre todo flexiva, es relativamente simple.

En el caso del español nos encontramos con fenómenos de mayor complejidad tales como:

- Flexión mucho más compleja. Para sustantivos y adjetivos se pueden identificar en torno a 20 grupos de variación para género y 10 para número. En los verbos existen 3 grupos regulares y unos 40 irregulares, con más de 100 formas flexionadas cada uno.
- Pronombres enclíticos asociados a formas verbales.
- Ambigüedades en la segmentación. Por ejemplo, `ténse+lo` vs. `tén+se+lo`.

Los *stemmers* no tratan adecuadamente tales fenómenos, lo cual redunda frecuentemente en normalizaciones erróneas. Nuestro sistema emplea un módulo de preprocesado lingüístico avanzado y un módulo lematizador para suplir tales deficiencias.

2.1. El preprocesador

Los etiquetadores actuales asumen que el texto de entrada está ya correctamente segmentado en *tokens* o unidades de información de alto nivel de significado que identifican cada uno de los componentes de dicho texto. Sin embargo dicha hipótesis de trabajo no es realista debido a la heterogeneidad de las fuentes de dicho texto. Es por ello que hemos desarrollado un módulo preprocesador (Graña, Barcala, y Vilares, 2002; Barcala et al., 2002) para realizar las tareas asociadas a dicho proceso de segmentación:

Filtrado. Conversión del formato de entrada y compactación de separadores.

Segmentación. Cada palabra individual y cada signo de puntuación debe constituir un *token* diferente, teniendo en cuenta abreviaturas, siglas, números decimales y fechas numéricas.

Separación de Frases. La regla general consiste en separar una frase en presencia de un punto seguido de mayúscula, prestando atención a las abreviaturas.

Descomposición de contracciones. Cada contracción se descompone en sus constituyentes, que ya se etiquetan.

Manejo de enclíticos. La forma verbal es separada de sus clíticos, etiquetando cada componente.

Identificación de locuciones. Los diferentes *tokens* que componen la locución son unidos y etiquetados como una unidad (Chanod y Tapanainen, 1996) en base a la información contenida en dos diccionarios: uno con las locuciones que se sabe con seguridad que siempre son locuciones (p.ej. `a pesar de`), y otro con aquellas locuciones que pueden serlo o no (p.ej. `sin embargo`). El preprocesador generará las segmentaciones posibles, encargándose posteriormente el etiquetador de seleccionar la válida.

Manejo de nombres propios. El procesamiento de propios se realiza en dos fases. Primero, y a partir de una muestra de los textos a procesar, el submódulo de entrenamiento identifica los propios situados en posiciones no ambiguas, aquéllas en las que no existe duda alguna de que estamos ante un propio. Con esta información se genera un *diccionario de entrenamiento*, a emplear en la fase de identificación. En ella el sistema identifica los propios del texto, tanto simples como compuestos y tanto en posiciones ambiguas como no,

empleando a mayores otro *diccionario externo* de propios.

El algoritmo inicial de identificación de propios compuestos (Graña, Barcala, y Vilares, 2002), basado en la etiquetación *conjunta* de secuencias de palabras en mayúscula y conectivas válidas, influía negativamente en el rendimiento del sistema de TR al no contemplar correspondencias parciales con sus componentes. Para resolver esto el algoritmo fue modificado para contemplar los componentes individuales de los propios compuestos (Barcala et al., 2002), de forma que para cada palabra en mayúscula W del propio se aplica el algoritmo:

si W aparece en el dicc. externo de propios
entonces W se etiqueta como propio
sino si W aparece en el lexicón con etiq. T
  entonces W es etiquetada como T
  sino si (W está en posición no ambigua)
o (W está en posición ambigua **y** W está en el dicc. de entrenamiento)
    entonces W se etiqueta como propio
    sino W se etiqueta como desconocida

2.2. El etiquetador-lematizador

La salida del preprocesador alimenta el módulo de etiquetación-lematización. Si bien podría utilizarse cualquier tipo de etiquetador, nuestro sistema emplea un etiquetador basado en un Modelo de Markov Oculto (HMM) de segundo orden, con capacidad para lematización, procesamiento de palabras desconocidas e integración de diccionarios externos implementados mediante autómatas finitos acíclicos (Graña, Barcala, y Alonso, 2001).

Los estados del modelo representan pares de etiquetas, y sus salidas, palabras. La etiqueta más probable para cada palabra de la frase se calcula aplicando el algoritmo de Viterbi. Las probabilidades del modelo se estiman a partir de un corpus de entrenamiento etiquetado, empleando técnicas de suavizado mediante la interpolación lineal de unigramas, bigramas y trigramas (Graña, 2000).

En el caso de las palabras desconocidas, sus etiquetas candidatas y sus probabilidades se fijan según su terminación. La distribución de probabilidades para cada sufijo particular de una longitud dada se genera a partir de todas las palabras pertenecientes al conjunto de entrenamiento que comparten el sufijo en cuestión. Dichas probabilidades son luego suavizadas mediante abstracción sucesiva (Samuelsson, 1993).

Existe una escasa disponibilidad de textos de entrenamiento para español, pero por contra se cuenta con amplios y completos lexicones. Para integrar esta información externa (Graña, Chappelier, y Vilares, 2001) se emplean las fórmulas de Good-Turing (Jelinek, 1998). Cada par palabra-etiqueta presente únicamente en el lexicón externo puede verse como un evento de frecuencia nula en el corpus de entrenamiento. Las fórmulas Good-Turing constituyen un método capaz de asignar probabilidades mayores que 0 a tales eventos.

El etiquetador debe ser capaz de manejar las segmentaciones ambiguas generadas por el preprocesador, tal y como se describe en el punto 2.1. No se trata sólo de decidir qué etiqueta asignar a cada *token*, sino también de decidir si algunos de ellos se agrupan o no un solo *token*. Para dicha tarea hemos considerado la evaluación de cada posible secuencia de *tokens* para su posterior comparación, con objeto de seleccionar la más probable (Graña, Alonso, y Vilares, 2002).

Una vez que el texto ha sido etiquetado, los términos de indexación los conforman los *lemas* de las *palabras con contenido* (nombres, verbos y adjetivos). De esta forma resolvemos el problema de la flexión, incrementando notablemente la cobertura del sistema de TR. El coste temporal del etiquetador-lematizador es lineal respecto a la longitud de la palabra, y cúbico respecto al tamaño del conjunto de etiquetas. Dado que nos basta con conocer la categoría gramatical de la palabra, el conjunto de etiquetas es pequeño, con lo que el aumento del coste computacional con respecto a los *stemmers* es mínimo.

2.3. Los fenómenos derivativos

La lematización del texto nos permite eliminar la flexión, permaneciendo todavía sin tratar los fenómenos de variación derivativa. Para abordarlos, y así minimizar la variación morfológica, se han combinado lematización y *stemming*. La variación flexiva del texto es eliminada en primer lugar empleando nuestro preprocesador y nuestro etiquetador-lematizador. A continuación, el texto resultante es procesado mediante un *stemmer*, para aprovechar su capacidad de tratamiento de los fenómenos derivativos.

3. La variación léxica

Entendemos por *variación léxica* el fenómeno asociado a la existencia de diferentes palabras para representar un mismo significado, como en el caso de la *sinonimia*. Existen dos aproximaciones comunes al problema: la expansión de la consulta con términos semánticamente relacionados y el empleo de distancias conceptuales.

Nuestra sistema opta por expandir la consulta mediante sinónimos. Esta técnica no es en absoluto nueva, aunque hasta ahora no se asignaba un peso al grado de sinonimia existente entre los términos originales y aquéllos generados en la expansión (Greenberg, 2001). Al disponer nuestro sistema de tal información, puede fijarse un umbral de sinonimia para controlar la amplitud de la expansión.

La sinonimia, en su definición más frecuente, se concibe como una relación entre dos expresiones con idéntico o similar significado. La controversia respecto a si entender la sinonimia con una concepción precisa o con una concepción aproximada, esto es, como una relación de identidad o como una relación de similaridad, ha estado presente desde que se empezó a estudiar dicha relación semántica. Nuestro sistema contempla la sinonimia como una relación gradual entre palabras, donde el *coeficiente de Jaccard* es empleado para calcular dicho grado de sinonimia, en base a la medida de similitud entre los conjuntos de sinónimos contemplados en un diccionario de sinonimia (Fernández, Graña, y Sobrino, 2002). Dados dos conjuntos X e Y, su *similaridad* se mide como:

$$sm(X,Y) = \frac{|X \cap Y|}{|X \cup Y|}$$

Considérese una palabra w con m_i significados posibles, y otra palabra w' con m_j significados posibles, donde $dc(w, m_i)$ representa la función que devuelve el conjunto de sinónimos contenidos en el diccionario para toda entrada w en el significado m_i. El grado de sinonimia entre w y w' para el significado m_i de w se calcula como:
$dg(w, m_i, w') = \text{máx}_j\ sm[dc(w, m_i), dc(w', m_j)]$

4. Frases en mayúscula

Una característica importante de las colecciones de prueba en TR, y que tiene un impacto considerable en el rendimiento de las técnicas de indexación con base lingüística, es la presencia de un gran número de errores tipográficos en los documentos, tal como se indica en (Figuerola et al., 2001) para el caso del corpus CLEF para español que utilizaremos (Peters, 2002). En particular, los títulos de documentos y secciones se escriben por lo general totalmente en mayúscula y sin signos ortográficos, confundiendo a los módulos de preprocesado y etiquetación, y produciendo, en consecuencia, lematizaciones erróneas. Esto constituye un problema pues dichos títulos suelen ser muy indicativos del tema tratado en el documento.

Para minimizar este problema, hemos incorporado en nuestro sistema un módulo, denominado *mayúsculas-a-minúsculas*, para el procesado de este tipo de entradas, pasándolas a minúscula y recuperando los signos ortográficos necesarios. Nuestro sistema emplea el contexto léxico de una frase en mayúscula para recuperar la información perdida. Para ello el primer paso consiste en identificar dichas frases en mayúscula. Consideraremos que una secuencia de palabras totalmente en mayúscula forma una *frase en mayúscula* cuando contenga tres o más palabras de las cuales al menos tres contengan más de tres caracteres. Para cada una de estas frases:

1. Se obtiene su contexto circundante (3 frases).
2. Para cada una de las palabras de la frase:
   - *a*) Se examina el contexto en busca de entradas con su misma *forma plana*[1]. Éstas serán sus candidatas.
   - *b*) Si se encuentran candidatas, se toma la más numerosa y, en caso de empate, la más cercana a la frase.
   - *c*) Si no se encuentran, se consulta el lexicón buscando todas las entradas con su misma forma plana, agrupándolas de acuerdo con su categoría gramatical y lema (la forma no es de interés).
     1) Si no se encuentra entrada alguna, se mantienen la etiqueta y el lema actuales.
     2) Si sólo se encuentra una entrada, se emplea ésa.
     3) Si se encuentran varias, se elige aquélla más numerosa en el contexto (en base a la categoría y

[1] Forma en minúscula y sin signos ortográficos.

el lema) y, en caso de empate, la más próxima a la frase.

En ocasiones, algunas frases en mayúscula preservan alguno de sus signos ortográficos, por ejemplo la ˜ en una Ñ. En tales situaciones los candidatos del contexto o del lexicón deberán observar tales restricciones.

Existen otras aproximaciones a este problema, capaces incluso de manejar documentos que carecen por completo de signos ortográficos (Yarowsky, 1999). Sin embargo nuestra situación es diferente, puesto que la práctica totalidad del documento está escrita correctamente, respetando minúsculas y signos, y sólo algunas frases están en mayúscula. Otra ventaja a nuestro favor es que nuestro sistema sólo precisa conocer la categoría y lema de la palabra, no su forma.

5. *Evaluación del sistema*[2]

En estos experimentos se ha empleado el corpus CLEF para español (Peters, 2002), 215.738 noticias emitidas por la Agencia EFE en 1994, formateadas en SGML, y ocupando un espacio en disco de 509 MBs. Por otra parte, las consultas constan de tres campos: un breve *título*, una *descripción* del tema en una frase, y una *narrativa* de mayor complejidad especificando los criterios de relevancia.

Las técnicas propuestas en este artículo son independientes con respecto al motor de indexación empleado, ya que el documento es primero procesado para obtener sus términos índice y, a continuación, el motor recibe el documento, ya normalizado, para su indexación. Por lo tanto, puede utilizarse cualquier motor estándar de indexación de texto, lo cual constituye una gran ventaja. Sin embargo, cada motor seguirá comportándose acorde con sus características, esto es, modelo de indexación, algoritmo de ordenación, etc. (Vilares, Vilares, y Alonso, 2001). En nuestro caso, se ha empleado el motor vectorial SMART (Buckley, 1985), con un esquema de pesos `atn-ntc`.

5.1. Puesta a punto

Antes de evaluar nuestras técnicas, el sistema fue afinado empleando las consultas de CLEF 2001. Esta fase previa de entrenamiento se realizó sobre la lematización, ya que en experimentos previos demostró ser un buen punto de partida para el estudio de la aplicación de técnicas de NLP (Vilares, Vilares, y Alonso, 2001). Se emplearon los tres campos de la consultas: título, descripción y narrativa. Sin embargo, experimentos paralelos empleando únicamente los campos título y descripción, los requeridos en la competición CLEF, obtuvieron iguales conclusiones.

El conjunto de entrenamiento estaba constituido por 50 consultas, de la 41 a la 90, para una de las cuales no existía documento relevante alguno. En el Cuadro 1 se muestran las medidas de rendimiento obtenidas en las diferentes fases del proceso. Dicho rendimiento se midió en base a los parámetros indicados en cada fila; de arriba a abajo: número de documentos devueltos, número de documentos relevantes devueltos (2694 esperados), *R-precision*, precisión media no interpolada para todos los documentos relevantes, precisión media por documento para todos los documentos relevantes, precisión media interpolada en 11 puntos de cobertura y precisión media interpolada en 3 puntos de cobertura.

En esta fase de puesta a punto, el caso base consistió en la no aplicación del módulo mayúsculas-a-minúsculas, y en el empleo de una lista de *stopwords* muy restringida, formada únicamente por los lemas de los verbos más comunes en español (*ser*, *estar*, *haber*, *tener*, *ir* y *hacer*). Los resultados obtenidos para este caso base pueden verse en la columna *p1* del Cuadro 1.

La primera mejora aplicada consistió en ampliar la lista de *stopwords* empleando los lemas de las palabras con contenido de la lista de *stopwords* de SMART para el español. Tal como puede verse en la columna *p2*, se produce una ligera mejora, además de una reducción extra del 6 % en el tamaño del fichero invertido del índice. Por ello se decidió seguir empleando esta lista ampliada.

El siguiente paso consistió en introducir el módulo mayúsculas-a-minúsculas. Los resultados, mostrados en la columna *p3*, muestran una mejora del comportamiento del sistema.

Sin embargo, llegados a este punto, todavía existe un gran número de errores tipográficos en el texto de los documentos, muchos de ellos consistentes en vocales no tildadas. Parte de este problema puede resolverse eliminando los signos ortográficos del texto

[2] Si bien no todos los experimentos reseñados en este artículo son *oficiales*, entendiéndose como tales aquéllos cuyo rendimiento ha sido evaluado por los correctores asignados por el CLEF, se ha seguido siempre el procedimiento indicado por la organización.

Cuadro 1: Puesta a punto del sistema con CLEF 2001 (2694 docs. relevantes esperados)

	p1	*p2*	*p3*	*p4*	*p5*	*p6*
Docs.	49k	49k	49k	49k	49k	49k
Rlvs. dev.	2602	2602	2607	2609	2621	2623
R-pr.	.507	.512	.509	.516	.525	.527
Pr. no int.	.523	.524	.531	.540	.551	.554
Pr. doc.	.628	.627	.634	.639	.648	.648
Pr. 11-p.	.529	.530	.538	.547	.557	.560
Pr. 3-p.	.542	.544	.551	.561	.573	.574

una vez normalizado, puesto que cuando se ha identificado el lema de la palabra ya no existe razón alguna para preservarlos. Puede argumentarse que esta medida eliminará las *tildes diacríticas*[3], pero al trabajar con los lemas de las palabras con contenido, tal problema desaparece. Sin embargo conservaremos las *'ñ'* del texto, no convirtiéndolas en *'n'*, pues esta medida podría introducir ruido en el sistema al normalizar palabras como *cana* y *caña* en un mismo término. Por otra parte, si bien en español es frecuente olvidarse de una tilde, una confusión entre *'ñ'* y *'n'* es extremadamente rara. En la columna *p4* puede verse la mejora obtenida con esta medida.

De una forma similar, se ha pasado a minúscula el texto resultante, a modo de un *stemmer*, obteniendo una mejora extra, tal como se aprecia en la columna *p5*.

Finalmente, se dobló la relevancia del campo título de la consulta con respecto a la descripción y la narrativa, puesto que es de suponer que concentra la semántica básica de la consulta. La mejora lograda de esta forma puede verse en la columna *p6*.

Las condiciones empleadas en este último caso serán las aplicadas en la evaluación: esquema de pesos `atn-ntc`, *stopwords* correspondientes a los lemas de las palabras con contenido de la lista de *stopwords* de SMART para español, empleo del módulo mayúsculas-a-minúsculas, eliminación de signos ortográficos y paso a minúsculas tras la normalización, y doble relevancia del campo título.

5.2. Experimentos finales

Se han comparado los resultados obtenidos para cuatro técnicas de normalización:

Stemming (*stm*). Tras la eliminación de *stopwords* en base a la lista suministrada por SMART, se ha aplicado el *stemmer* empleado por el motor de búsqueda de código abierto Muscat (Porter y Boulton, 2000), basado en el algoritmo de Porter (Baeza-Yates y Ribeiro-Neto, 1999). En este proceso el *stemmer* elimina las tildes y pasa el texto a minúscula.

Lematización (*lem*). Indexación de las palabras con contenido del texto vía preprocesado y lematización, resolviéndose de este modo la variación flexiva.

Lematización & stemming (*l&s*). Combinación de ambas técnicas, *lem* y *stm*, para abarcar también la variación derivativa.

Expansión de sinónimos (*sin*). El proceso es el mismo que para la lematización (*lem*), si bien la consulta es expandida con sus sinónimos para así tratar la variación léxica. Dos palabras se consideran sinónimas si su grado de similaridad es al menos de un 80 %. Experimentos previos habían mostrado que la expansión de la narrativa introduce excesivo ruido, por lo que sólo se expanden los campos título y descripción.

Los métodos *lem* y *sin*, junto con la parte flexiva de *l&s*, están fundamentados lingüísticamente, lo que permite hacer frente a fenómenos lingüísticos complejos tales como pronombres enclíticos, contracciones, expresiones y reconocimiento de propios. Por contra, la técnica *stm* se limita a realizar una mera eliminación de sufijos sin tener en cuenta tales fenómenos lingüísticos, dando lugar a normalizaciones incorrectas que introducen ruido en el sistema. En el caso de los pronombres enclíticos, por ejemplo, estos son considerados como meros sufijos a eliminar.

Para estos experimentos finales, cuyos resultados pueden verse en los Cuadros 2 y 3, se han empleado las 50 consultas correspondientes al CLEF 2002, 91 a 140, bajo las condiciones fijadas en el punto 5.1.

En lo referente a la variación morfológica, nuestros resultados muestran un mejor comportamiento de la lematización (*lem*) frente al *stemming* (*stm*), debido a su capacidad para tratar fenómenos lingüísticos complejos. Debe tenerse en cuenta que *lem* sólo elimina

[3]Tildes que diferencian palabras con igual forma gráfica pero significado diferente, p.ej. *mí* - *mi*.

Cuadro 2: Experimentos finales con CLEF 2002 (2854 docs. relevantes esperados)

	stm	*lem*	*sin*	*l&s*
Docs.	50k	50k	50k	50k
Rlvs. dev.	2570	2593	2582	2565
R-pr.	0.4892	0.4924	0.4721	0.5038
Pr. no int.	0.5097	0.5186	0.5057	0.5201
Pr. doc.	0.5255	0.5385	0.5264	0.5273
Pr. 11-p.	0.5239	0.5338	0.5192	0.5313
Pr. 3-p.	0.5193	0.5378	0.5249	0.5339

Cuadro 3: Precisión en los 11 niveles de cobertura estándar con CLEF 2002

Co.	Pr.			
	stm	*lem*	*sin*	*l&s*
0.00	0.8887	0.8859	0.8492	0.8671
0.10	0.7727	0.7888	0.7753	0.7624
0.20	0.6883	0.7096	0.6965	0.6927
0.30	0.6327	0.6417	0.6246	0.6392
0.40	0.5909	0.6025	0.5848	0.5995
0.50	0.5465	0.5628	0.5447	0.5583
0.60	0.5041	0.4918	0.4720	0.4937
0.70	0.4278	0.4214	0.4109	0.4340
0.80	0.3231	0.3410	0.3336	0.3508
0.90	0.2456	0.2595	0.2547	0.2735
1.00	0.1422	0.1666	0.1653	0.1735

la variación flexiva y, sin embargo, sus resultados son mejores que los de *stm*, que trata también la derivativa, incluso a nivel de cobertura (2593 documentos relevantes devueltos frente a 2570). La combinación de ambas soluciones (*l&s*) no aporta mejoras respecto a la sola lematización, probablemente debido al ruido introducido en el sistema por la normalización de términos supuestamente relacionados desde el punto de vista derivativo.

En lo que respecta a la variación léxica, los resultados obtenidos mediante la expansión mediante umbrales de sinonimia (*sin*) no obtienen mejoras respecto a la lematización. Ello probablemente se deba a la distorsión causada en el sistema al aplicar una expansión *total* ya que, al no disponer de procedimientos de desambiguación del sentido de las palabras, se emplean todos los sinónimos de todos los términos a expandir en la consulta.

6. Conclusiones

Partiendo de los resultados obtenidos, la lematización de palabras con contenido (*lem*) semeja ser la mejor opción para la normalización de textos en TR, a pesar de que sólo maneja la variación flexiva. Los intentos por manejar la variación derivativa mediante *stemming* (*stm*) o su empleo combinado con la lematización (*l&s*) no mejoran sus resultados. Por otra parte, nuestra aproximación al problema de la variación léxica por medio de la expansión mediante umbrales de sinonimia (*sin*) no aporta tampoco mejoras.

Estos resultados, junto a otros anteriores obtenidos con diferentes esquemas de peso y modelos de recuperación (Alonso, Vilares, y Darriba, 2002; Vilares, Barcala, y Alonso, 2002; Vilares, Vilares, y Alonso, 2001), apuntan a la lematización como el mejor punto de inicio para el desarrollo de métodos de normalización que hagan frente a niveles de variación lingüística más complejos, probablemente mediante soluciones basadas en *feedback* o postprocesado lingüístico para la reordenación de los documentos recuperados. Creemos que un enfoque similar al *relevance feedback*, y basado en la expansión de la consulta mediante las variantes lingüísticas de los términos iniciales contenidas en los documentos más relevantes, podría mostrarse efectivo. Otra opción a estudiar sería la aplicación de un *relevance feedback* automático tradicional seguido por una fase extra de filtrado y ponderación de las variantes contenidas en los nuevos términos generados.

Bibliografía

Alonso, Miguel A., Jesús Vilares, y Víctor M. Darriba. 2002. On the usefulness of extracting syntactic dependencies for text indexing. En volumen 2464 de *Lecture Notes in Artificial Intelligence*. Springer-Verlag, Berlín-Heidelberg-Nueva York, páginas 3–11.

Baeza-Yates, Ricardo y Berthier Ribeiro-Neto. 1999. *Modern information retrieval*. Addison-Wesley, Harlow, Inglaterra.

Barcala, Fco. Mario, Jesús Vilares, Miguel A. Alonso, Jorge Graña, y Manuel Vilares. 2002. Tokenization and proper noun recognition for information retrieval. En *3rd International Workshop on Natural Language and Information Systems (NLIS 2002)*, Los Alamitos, California, USA. IEEE Computer Society Press.

Buckley, Chris. 1985. Implementation of the SMART information re-

trieval system. Informe técnico, Department of Computer Science, Cornell University. Fuentes disponibles en `ftp://ftp.cs.cornell.edu/pub/smart`.

Chanod, Jean-Pierre y Pasi Tapanainen. 1996. A non-deterministic tokeniser for finite-state parsing. En *Proceedings of the Workshop on Extended finite state models of language (ECAI'96)*, Budapest, Hungría.

Fernández, Santiago, Jorge Graña, y Alejandro Sobrino. 2002. A Spanish e-dictionary of synonyms as a fuzzy tool for infor mation retrieval. En *Actas de las I Jornadas de Tratamiento y Recuperación de Información (JOTRI 2002)*, León, España.

Figuerola, Carlos G., Raquel Gómez, Angel F. Zazo, y José Luis Alonso. 2001. Stemming in Spanish: A first approach to its impact on information retrieval. En Carol Peters, editor, *Working notes for the CLEF 2001 workshop*, Darmstadt, Alemania.

Graña, J. 2000. *Técnicas de Análisis Sintáctico Robusto para la Etiquetación del Lenguaje Natural.* Ph.D. tesis, Universidad de La Coruña, La Coruña, España.

Graña, Jorge, Miguel A. Alonso, y Manuel Vilares. 2002. A common solution for tokenization and part-of-speech tagging: One-pass Viterbi algorithm vs. iterative approaches. En *Text, Speech and Dialogue*, Lecture Notes in Computer Science. Springer-Verlag, Berlín-Heidelberg-Nueva York.

Graña, Jorge, Fco. Mario Barcala, y Miguel A. Alonso. 2001. Compilation methods of minimal acyclic automata for large dictionaries. En Bruce W. Watson y Derick Wood, editores, *Proc. of the 6th Conference on Implementations and Applications of Automata (CIAA 2001)*, páginas 116–129, Pretoria, Sudáfrica.

Graña, Jorge, Fco. Mario Barcala, y Jesús Vilares. 2002. Formal methods of tokenization for part-of-speech tagging. En volumen 2276 de *Lecture Notes in Computer Science.* Springer-Verlag, Berlín-Heidelberg-Nueva York, páginas 240–249.

Graña, Jorge, Jean-Cédric Chappelier, y Manuel Vilares. 2001. Integrating external dictionaries into stochastic part-of-speech taggers. En *Proceedings of the Euroconference Recent Advances in Natural Language Processing (RANLP 2001)*, páginas 122–128, Tzigov Chark, Bulgaria.

Greenberg, Jane. 2001. Automatic query expansion via lexical-semantic relationships. *Journal of the American Society for Information Science and Technology*, 52(5):402–415.

Jelinek, Frederick. 1998. *Statistical Methods for Speech Recognition.* MIT Press, Cambridge, MA.

Lennon, M., D.S. Pierce, y P. Willett. 1981. An evaluation of some conflation algorithms. *Journal of Information Science*, 3:177–183.

Peters, Carol, editor. 2002. *Results of the CLEF 2002 Cross-Language System Evaluation Campaign. Working Notes for the CLEF 2002 Workshop*, Roma, Italia. Página oficial del CLEF: `http://www.clef-campaign.org`.

Porter, M. y R. Boulton. 2000. Object muscat, an open source search engine. En volumen 34 de *ACM SIGIR Forum.* ACM Press, Nueva York.

Samuelsson, Christer. 1993. Morphological tagging based entirely on bayesian inference. En Robert Eklund, editor, *Proceedings of the 9th Nordic Conference on Computational Linguistics*, Estocolmo, Suecia.

Vilares, Jesús, Fco. Mario Barcala, y Miguel A. Alonso. 2002. Using syntactic dependency-pairs conflation to improve retrieval performance in Spanish. En volumen 2276 de *Lecture Notes in Computer Science,.* Springer-Verlag, Berlín-Heidelberg-Nueva York, páginas 381–390.

Vilares, Jesús, Manuel Vilares, y Miguel A. Alonso. 2001. Towards the development of heuristics for automatic query expansion. En volumen 2113 de *Lecture Notes in Computer Science.* Springer-Verlag, Berlín-Heidelberg-Nueva York, páginas 887–896.

Yarowsky, David. 1999. A comparison of corpus-based techniques for restoring accents in Spanish and French text. En *Natural Language Processing Using Very Large Corpora.* Kluwer Academic Publishers, páginas 99–120.