Revista Unimar
Número 58
Diciembre 2011, 107-114
ISSN 0120-4327
ISSN -E 2216-0116

STEMMING EN ESPAÑOL PARA DOCUMENTOS RECUPERADOS DE LA WEB*

STEMMING IN THE SPANISH LANGUAGE FOR DOCUMENTS RECOVERED FROM THE WEB

Hugo Armando Ordoñez Eraso**
Docente Investigador, Facultad de Ingeniería, Universidad Mariana, San Juan de Pasto, Colombia

Carlos Alberto Cobos Lozada, Phd***
Docente Investigador Titular, Departamento de Sistemas, Universidad del Cauca, Popayán, Colombia

Fecha de recepción:
17 de noviembre de 2011
Fecha de aprobación:
20 de enero de 2012

Palabras claves:
Búsqueda web, stemming en español, evaluación n-gramas.

RESUMEN

La recuperación de información en internet, hoy en día se ha convertido en una importante área de investigación, debido al crecimiento acelerado de las fuentes de información que en la *web* se encuentran. La investigación en este campo se ha centrado en crear mecanismos, métodos y herramientas como los algoritmos de *stemming* y los meta-buscadores, que optimizan la precisión en este tipo de tareas, mejorando así los resultados retornados al usuario que utiliza el sistema. En este artículo se presenta y describe a nivel general, los componentes de un algoritmo denominado Filtro Español, que permite realizar *stemming* a documentos escritos en español en un sistema de búsqueda *web*. Este se diseña, con el fin de ampliar automáticamente la búsqueda a todas las variaciones morfológicas de las palabras digitadas por los usuarios en las consultas y el contenido de los documentos. Para validar la eficiencia del analizador, se realizó un cálculo de similitud entre los documentos, aplicando el método de *n-gramas*. La evaluación contempla la precisión del filtro donde los resultados de la primera valoración son interesantes.

Key words:
Web search, stemming in spanish, evaluation n-grams.

ABSTRACT

Information retrieval on the Internet today has become an important research tool due to the rapid growth of information sources. Research has focused on creating mechanisms, methods and tools such as stemming algorithms and meta-search engines, which improve accuracy in this kind of tasks. This article presents and generally describes the components of an algorithm called Spanish Filtering, which allows stemming to written documents in this language to the web search. The design automatically expands the search to all the morphological variations of the words entered by users in consultation with the content of the documents. To validate the efficiency of the analyzer it performed a calculation of similarity with the documents, using the method of n-grams. The evaluation considers the accuracy of the filter where the results of the first evaluation are interesting.

* Artículo Resultado de Investigación.

** Doctorando Ingeniería Telemática, Universidad del Cauca; Magíster en Computación, Universidad del Cauca; miembro del Grupo de I+D en Tecnología de Información (GTI), Universidad del Cauca; Miembro del Grupo de Investigación en Ingeniería de Sistemas, Universidad Mariana.
Correo electrónico: hogoordonez@unicauca.edu.co; hordonez@umariana.edu.co

*** Dr. Ingeniería de Sistemas y Computación, Universidad del Cauca; Magíster en Informática e Ingeniero de Sistemas, Universidad del Cauca; miembro del Grupo de I + D en Tecnologías de la Información (GTI). Correo electrónico: ccobos@unicauca.edu.co

El proceso de recuperación de información en internet se basa en la representación objetiva de los rasgos de los documentos a recuperar; esto juega un papel fundamental en la descripción y estandarización de los contenidos que estos presentan, como lo expresan Manning, Raghavan y Schütze (2007, p. 104). Por su parte, Baeza-Yates, Castillo y Keith (2006, p. 527-538) y Tait (2005, p. 388) manifiestan que la determinación de los rasgos semánticos aporten más información para describir los documentos más relevantes, recalcando que esto no es una tarea trivial y que de ella depende en gran medida el resultado de las tareas posteriores en la recuperación de información.

Por esta razón, según Rolleke (2006, p. 5) el proceso de *stemming* se encarga de la estandarización y la correcta representación de los rasgos semánticos descriptores de los documentos, de forma generalizadora. El *stemming* aplicado a la recuperación de la información se ha planteado de diversas maneras y comienza a estudiarse en los años 60, según Baeza-Yates (2006, p. 25), con el fin de reducir los tamaños de los índices de texto (Manning et al. 2007, p. 104; Rijsbergen 1979, p. 157), y como una forma de normalizar los términos.

De forma muy sencilla, se puede definir el *stemming* como el proceso que reduce un conjunto de palabras a su *stem* o raíz léxica común. Así, camión, sería la raíz léxica de camiones, camionero, camioneta, entre otros. Por esta razón, la eficacia del *stemming* ha sido objeto de discusión, concluyendo que éste mejora significativamente los resultados en los Sistemas de Recuperación de Información –SRI- (Manning et al. 2007, p. 329) como los meta buscadores que actualmente se convierten en nuevas estrategias, según Ordoñez (2010, p. 365) para las tareas de búsqueda y recuperación de información en la *web*.

A continuación se presenta algunos trabajos relacionados, se describe los elementos que componen el Analizador Español, la aplicación del filtro a documentos recuperados de tres fuentes de información primaria como: Google, Yahoo y Bing, la evaluación del filtro, las conclusiones y el trabajo a futuro.

OBJETIVOS E HIPÓTESIS DE LA INVESTIGACIÓN

Para profundizar y abordar puntos importantes en la presente investigación, se platea los siguientes objetivos e hipótesis:

- Realizar un estudio detallado de los mecanismos y algoritmos para realizar *stemming*.
- Identificar algoritmos específicos para el idioma Español.
- Estudiar los componentes metodológicos y conceptuales que componen los algoritmos de *stemming*.
- Identificar librerías que implementan mecanismos para la realización de *stemming*
- Desarrollar un algoritmo basado en los componentes aportados por las librerías estudiadas.
- Diseñar y desarrollar una herramienta prototipo que incorpore el algoritmo diseñado.
- Validar la eficiencia del algoritmo en pruebas realizadas con usuarios.

Hipótesis

El uso de algoritmos de *stemming* en documentos recuperados de la *web*, es adecuado como estrategia para aumentar la relevancia en las consultas realizadas por los usuarios.

Trabajos relacionados

Una de las labores más importantes en el campo de la recuperación de información contextual, es la de aumentar la relevancia a los documentos recuperados (Jansen 2006, p. 70) en las consultas que los usuarios realizan en los sistemas que pertenecen a esta área; la ejecución de estas labores puntualizando en el idioma español, se encuentra que una consulta puede ser vista como un problema de Procesamiento de Lenguaje Natural –NLP- (por sus siglas en inglés), debido a que la información deseada está codificada en formato de texto. Con esto, el aumento de la eficacia en las consultas realizadas en este idioma, depende del nivel de desarrollo de herramientas y recursos con que se cuente para el procesamiento de dicho lenguaje. Aunque éstas tienen un largo camino por recorrer, cuentan ya con diccionarios electrónicos, herramientas de lematización y/o *stemming*, como las que se menciona a continuación.

Con lo anterior y de acuerdo con Carmona (1998, p. 389), se propone, una Morfología Compu-

tacional, MACO, como el lematizador de referencia para el español, aclarando que es posible realizar el *stemming* mediante un algoritmo que use reglas gramaticales de derivación morfológica para el idioma en cuestión, o usando un diccionario informatizado que asocie a cada forma su lema (palabra) representante.

Por otra parte, Pombo (2009, p. 391) plantea una estrategia de normalización de términos basada en n-gramas (donde formalmente, un n-grama es una sub-secuencia de longitud *n* de una secuencia dada. Así, por ejemplo, podemos dividir la palabra "patata" en los 3-gramas de caracteres superpuestos -pat, -ata-, -tat- y -ata-) de caracteres como alternativa para el tratamiento de consultas realizadas en español, buscando además, una metodología simple que pueda ser utilizada independientemente de la base de datos documental considerada y, de los recursos lingüísticos disponibles, incorporando un corrector ortográfico contextual que funciona como un etiquetador sintáctico, basado en una extensión dinámica del algoritmo de Viterbi sobre Modelos Ocultos de Markov de segundo orden.

Velásquez y Sidorov (2005, p. 392) proponen el nombre de Agme, un modelo que consiste en un conjunto de reglas para obtener todas las raíces de una forma de palabra para cada lexema, su almacenamiento en el diccionario, la producción de todas las hipótesis posibles durante el análisis y, su comprobación a través de la generación morfológica. Se usa un diccionario de cuarenta mil lemas a través del cual se puede analizar más de dos millones y medio de formas gramáticas posibles. Para descubrir estas formas incorpora procesos flexivos que ocurren principalmente en los nombres (sustantivos y adjetivos) y verbos. Las demás categorías gramaticales (adverbios, signos de puntuación, conjunciones, preposiciones, etc.), presentan poca o nula alteración flexiva; el tratamiento de estas últimas se realiza mediante la consulta directa al diccionario.

Además, Moreda Leirado, Vázquez y Penabad (2005, p. 394) plantean un algoritmo de *stemming* para el idioma español en el dialecto gallego construido por reglas, tantas como sufijos existen en dicha lengua. Cuando una palabra debe ser reducida a su raíz, el algoritmo comprueba qué regla debe aplicarse, teniendo en cuenta sus sufijos. Para la construcción de estas reglas se basa en Gramática de la Lengua Gallega (Vol. II, III) y el Vocabulario Ortográfico de Lengua Gallega. En su proceso de ejecución para la realización de *stemming*, el algoritmo realiza cuatro etapas a saber:

1) **Sufijo a cambiar**. Denominada como la terminación que se sustituye o, en ocasiones, se elimina.

2) **Tamaño mínimo de la raíz**. Específica el tamaño mínimo que puede tener la raíz una vez se ha eliminado el sufijo.

3) **Sufijo sustituto**. Es el sufijo que sustituye al "sufijo a cambiar". Si se especifica una cadena vacía, esto indica que el sufijo no se cambia por nada, sino que se elimina.

4) **Lista de excepciones**. Contiene una relación de palabras para las que la regla no se debe aplicar.

Centrado más en la aplicación de *stemming* a documentos recuperados de la *web,* Bender (2005, p. 393) presenta un método basado en el algoritmo de Porter para ordenar los resultados, ejecutando los siguientes pasos: extraer prefijos y sufijos para normalizar las palabras y generar vectores representativos de cada documento, para esto, construyen un diccionario en función de las palabras que aparecen en el texto del documento del cual se extraerán los términos. Generando un vector para cada documento.

Las anteriores investigaciones, resaltan la fortaleza de la aplicación de *stemming* en los procesos de recuperación de información, en colecciones de documentos controladas como: repositorios y colecciones de documentos no controladas como la *web*, (donde se encuentra enmarcada la presente propuesta), obteniendo buenos resultados, lo cual ha aportado un gran campo de acción en la investigación de esta temática en la parte académica.

RESULTADOS

1. Elementos del Filtro Español

El Analizador Español se comporta como un filtro semántico para el idioma español. El cual tiene el objetivo reducir a su raíz léxica, los términos que se encuentran en los documentos recuperados de las fuentes primarias de consulta, las cuales pueden estar en: repositorios de archivos textuales, bases de datos, documentos indexados por los motores de búsqueda tradicionales (Google, Yahoo, Bing). Con el propósito de obtener búsquedas con alto grado de efectividad, se ha tomado como iniciativa el algoritmo de Porter (Velásquez & Sidorov 2005, p. 392), utili-

zando la librería *Snowball* del proyecto Apache Lucene (Schindler 2007, p. 398; Gospodnetic 2005, p. 407), que es una implementación de referencia del algoritmo de Porter. Cabe resaltar que en este algoritmo, se hace uso del método *Token Stream* que se encarga de reducir y filtrar el contenido del archivo o consulta en *tokens*; además, para este método fueron implementados los siguientes filtros: ***Standard Filter***, que elimina los signos de puntuación; ***Lower Case Filter***, que convierte el contenido a minúsculas; ***Stop Filter***, que filtra el contenido con el listado de palabras vacías; ***Spanish Stemmer***, el cual es un censor específico de español que contiene un gran conjunto de palabras comunes del idioma; asimismo los siguientes filtros desarrollados por el autor del proyecto: ***Filtro Español***, que filtra cada *token* al español; ***Analizador***, el cual retorna el contenido aplicando los filtros para ser estudiado por el ***Analizador Español***.

A continuación, se presenta pseudocódigo del algoritmo de *stemming* para español.

1.1. Filtro Español

```
privado filtro-español filtro;
privado palabra pal = nulo;
publico filtro  (palabra entrada) {
   principal (entrada);
   filtro= new filtro-español ();
 }
 publico final palabra siguiente(){
  si  ((palabra = entrada.siguiente()) == nulo) {
     retorne nulo;
   } sino {
     filtro.tomeactual (palabra.siguiente());
     filtro.palabra();
     cadena s = filtro.tomeactual();
     si-no (s.igual(palabra.terminotexto())) {
       retorne nuevo palabra(s, palbra.iniciopal
        bra(),
          palabra.finplabra(),palabra.tipo());
     }
     retorne palabra;
   }
 }
 publico  tomarfiltro(filtroespañol filtro) {
   si (filtro != nulo) {
     este.filtro = filtro;
   }
  }
}
```

1.2. Analizador

```
publico final cadena-vector pablas vacias ={};
privado conjunto<objeto> table- palabras = nuevo
- conjunto <objeto>();
privado conjunto<objeto> tabla-ect = nuevo-
conjuntot<objeto>();
   publico analizador() {
      tabla palabras = filtro parada .crearconjunto
      parada (palabras vacías);
   }
publico final flujo-palabra (cadena  nombre cam
po, lectura lector) {
 palabra flujo resultado = nueva palabra - standar(lector);
          resultado = nuevo filtrostandar(resultado);
         resultado = nuevo filtrominusculas(resultado);
        resultado = nuevo filtroparada (resultado,tabla-
        palabras );
     resultado = new filtro español(resultado);
    returne  resultado;
    }
}
```

1.3. Analizador Español

```
cadena-vector = analizador.palabrasvacías;
directorio = nuevo  directorioa-memoria();
escritor-indice  indice = nuevo indice escritor (directorio,
nuevo analizador-español(), verdadero);
 para (entero i = 0; i < tabla.totalregistro(); i++) {
          cadena contenido = (string) tabla.tomarfila(i, 0);
         documento documentofiltrado = new documento ();
          cadena docfiltrado = "";
          palabra-flujo  palabra-flujo = new analizador
 (palabrasvacias).palabra-flujo  ("contenido", nuevo ca-
dena-lectora(contenido));
flujo-palabra  flujo-filtrado = nuevo  filtro español (palabra
flujo);
        palabra palabra;
   mientras que ((palbra = palabra-flujo ( filtrado.siguien
   te()) diferente de nulo) {
      docfiltrado += palbra.textopalbra() + " ";
    }
    docfiltrado = docfiltrado.eliminarespaciosblancos();
 si (docfiltrado.longitud()  > 0) {
     documento.adicionar(nuevo campo("iddocumento",
docfil trado, campo.yes, campo.index, campo.vector));
 documento.adicionar(nuevo  campo ("título ", docfiltra-
do, campo.yes, campo.index));
      documento.add(nuevo  campo ("contenido ", (cade-
na) tabla.tomarfila(i, 0), campo.yes, campo.index., cam-
po.vector));
     documento.adicionar(nuevo  campo("contenidostem
```

```
ming", (cadena) tabla.tomarfila(i, 2), campo.yes, campo.
index,campo.vector));
    documento.add(nuevo campo ("url ", (string) tabla.
tomarfila (i, 1), campo.yes, campo.index));
 documento.adicionar(nuevo campo ("buscadororigen",
(string) tabla.tomarfila(i, 3), field.yes, campo.index.));
     index.adddocument(documento);
   } sino {
    eliminardoc(tabla, i);
   }
  }
  index.optimizar();
  index.cerrar();
```

2. Aplicación del Filtro a Documentos Recuperados de Google, Yahoo, Bing

Para la experimentación, se utilizó tres bases de datos de las fuentes principales de indexación de documentos en la *web* (Google, Yahoo!, Bing), en donde la recuperación de documentos y la aplicación del filtro son en tiempo real, haciendo uso de las Interfaces de Programación de Aplicaciones (Apis) que cada una de estas fuentes de información aporta para realizar consultas a sus bases de datos.

En el proceso de filtrado, se toma el resumen (o *snipet*) de cada uno de los documentos recuperados de las fuentes de consulta y, se realiza los procesos de eliminación de: acentos, comas, puntos y demás signos de puntuación, para tratar los términos de forma uniforme, se convierte el contenido a minúsculas para aumentar la similitud de los términos, se elimina palabras vacías (*stop words*), ya que todos los idiomas tienen un conjunto de palabras de frecuente aparición, utilizadas para garantizar la concordancia sintáctica de las frases en los contenidos, que no aportan ningún significado a un documento, sino que son utilizadas para seguir las reglas de idioma; este es el caso de las preposiciones, conjunciones, determinantes, etc. En la indexación existe un conjunto de estas palabras, llamadas palabras vacías, que sirven como referencia para excluir palabras en el proceso de indexación. Además se elimina documentos duplicados y sin contenido.

A continuación, la Tabla No. 1 muestra las consultas realizadas en temas de Ciencias de la Computación, los documentos recuperados y la aplicación del Filtro Español para cada uno de estos.

Tabla 1. ***Aplicación de filtro español a documentos recuperados***

Consulta	Consulta filtrada	Documento estándar	Documento filtrado
Recuperación de la información	recuper inform	Extracción y recuperación de información mediante la clasificación no supervisada de algoritmos: *clustering*, mapas auto-organizativos de *kohonen*	extraccion recuper inform clasif extraccion recuper inform mediant clasif supervis algoritm clustering map autoorganiz kohon
Algoritmos genéticos	algoritm genet	Modificación de la semántica de textos mediante algoritmos genéticos ... Utilizar genoma de longitud variable y algoritmos genéticos desordenados	modif semant text mediant algoritm genet utiliz genom longitud variabl algoritm genet desorden
Minería de datos	min dat	La minería de datos (DM, Data Mining) consiste en la extracción no trivial de información que reside de manera implícita en los datos	min dat dm dat mining cons extraccion no trivial inform que resid maner implicit dat
Sistemas colaborativos	sistem colabor	Cómo trabajar en equipo Sistemas Colaborativos, anécdotas, y también trabajar en equipo virtualmente. El trabajo en equipo muchas veces parece imposible ...	com trabaj equip sistem colabor anectod equip virtual equip much vec parec impos

En la Tabla No. 1 se muestra los resultados de aplicar el filtro a cuatro documentos recuperados de las fuentes de información, en ellos se aprecia que todo su contenido fue convertido a minúsculas, no contiene palabras vacías tales como (en, el, los, las, y, este…) y todos sus términos fueron reducidos a su raíz léxica o *stem*. Una vez aplicado el filtro, se comprueba que el vocabulario de los documentos recuperados, se reduce significativamente en todos ellos.

3. Evaluación de la precisión del filtro

En este proceso se definió como objetivo, evaluar el potencial discriminatorio de los *n-gramas* mediante combinaciones de secuencias de categorías gramaticales, como posibles marcas de autoría para los fines de la comparación de textos escritos en español. Esta técnica también puede emplearse para realizar eficientemente encajes por aproximación, como lo manifiesta Abdou (2006 p. 395), convirtiendo una secuencia de elementos en un conjunto de n-gramas, el cual puede introducirse en un espacio vectorial (en otras palabras, representarse como un histograma), permitiéndole a la secuencia compararse con otras de manera eficiente. Lo anterior es definido por la siguiente ecuación:

$$Sim = \frac{2C}{A+B}$$

Donde:

A = Número de digramas únicos en la primera palabra.

B = Número de digramas únicos en la segunda palabra.

C = Número de digramas únicos compartidos por la primera y segunda palabras.

La aplicación del método, se aplica a la comparación de las palabras que componen los documentos estándar y las palabras que componen los documentos filtrados, formando los n-gramas así:

Recuperación = re - ec – cu – up – pe – er – ra - ac-ci – io - on

Información = in – nf - fo – or – rm – ma – ac – ci –io - on

La Tabla No. 2 muestra las similitudes encontradas en este proceso.

Tabla 2. ***Cálculo de similitudes entre términos estándar y términos filtrados***

	Extracción	Recuperación	Información	Mediante	Clasificación	Supervisada	Algoritmos
extracc	1,00						
recuper		0,71					
inform			0,67				
mediant				0,92			
clasif					0,6		
supervis						0,82	
algoritm							0,88

La Tabla No. 2 muestra que la efectividad del filtro se encuentra entre 60% como punto mínimo y el 100% como punto máximo, con un promedio de precisión en la tarea de aplicación de *stemming* de 80% para cada documento.

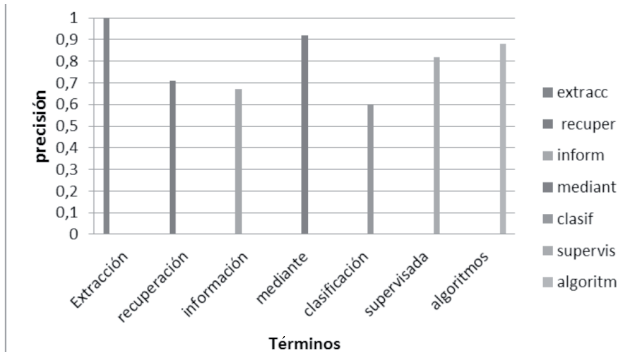

Figura 1. ***Nivel de Precisión***

La Figura 1 muestra de manera visual el nivel de presición alcanzado por el filtro en cada unos de los términos que se utilizaron en la evaluación.

Por otra parte, se destaca la aplicación del filtro en dos modelos de meta-buscadores, los cuales fueron evaluados destacando la relevancia de los resultados, para las aplicaciones de Recuperación de Información IR como lo menciona Massimo (2008, p. 289), en especial en búsqueda *web*. Teniendo presente que a los usuarios lo que más les importa es qué tan relevantes o útiles son los resultados en las primeras páginas (ya que normalmente ellos no revisan toda la lista de resultados). Esta medida se conoce como la "precisión en *K*" (Precisión *at- K*) y tiene la ventaja de no requerir ninguna estimación del conjunto total de resultados relevantes (factor

clave en la evaluación de meta buscadores web); por lo anterior, esta medida fue usada para evaluar los modelos denominados Ghobi y Onto Ghobi, cuyos resultados son muy prometedores. (Para mayor información ver Ordoñez 2010, p. 365 y 390).

CONCLUSIONES

- El desarrollo de la presente investigación aportó la conceptualización y apropiación profunda en temáticas de recuperación de información y *web* semántica, las cuales son de gran importancia en la actulidad.

- Con base en el modelo definido se diseñó, desarrolló un filtro para realizar *stemming* a documentos en idioma español recuperados de la *web*, que es el resultado de una especialización del algoritmo de Porter, utilizando la librería Snowball del proyecto Apache Lucene.

- En la evaluación permitió dar validez, en lo cual el filtro demostró poseer una precisión del 80% para cada documento de los documentos filtrados.

- La metodología de desarrollo basada en ciclos iterativos incrementales (modelo, aplicación, evaluación) permitió manejar la complejidad del problema y evaluar adecuadamente las características que el modelo debía cumplir en cada ciclo. Se logró con lo anterior, un control más efectivo del cumplimiento gradual de los objetivos de la investigación y la disminución de los riesgos de la misma.

- Como trabajo a futuro se espera diseñar e implementar un algoritmo con base en características más específicas de las lenguas romances, que se pudiera presentar en los documentos recuperados.

AGRADECIMIENTOS

A los directivos de la Facultad de Ingeniería de la Universidad Mariana por el apoyo prestado en el desarrollo de la investigación y al Grupo de I+D en Tecnologías de la Información (GTI) de la Universidad del Cauca por sus asesorías y aportes.

REFERENCIAS BIBLIOGRÁFICAS

Manning, C., Raghavan, P. & Schütze, H. (2008) *An Introduction to Information Retrieval*. Cambridge, England: Cambridge University Press.

Baeza-Yates, R., Castillo, C. & Keith, B. (2006) *Web Searching*. En: *Encyclopedia of Language & Linguistics*. Elsevier: Oxford. p. 527-538.

Rolleke, T., Tsikrika, T. & Kazai, G (2006) *A general matrix framework for modelling Information Retrieval*. Information Processing & Management Vol. 42, (1) 4-30.

Jardine & C.J.V. Rijsbergen. (2008) The Use of Hierarchic Clustering in Information Retrieval ... 193 - Usa: Pittsburgh.

Jansen, B. & Spink, A. (2006) *How are we searching the World Wide Web? A comparison of nine search engine transaction logs.* Information Processing & Management.

Carmona, J., Cervell, S., Màrquez, L., Martí, M., Padró, L., Placer, R., Rodríguez, H., Taulé M. & Turmo, J. (1998) *An Environment for Morphosyntactic Processing of Unrestricted Spanish text.*