PROCESAMIENTO DE LENGUAJE NATURAL

Elaboración de un resumen para los temas 2 y 3 del curso.

Coincidiendo con el cierre del primer bloque, correspondiente a los temas 2 y 3 (capítulos 2,3,4 y 5 del libro base) se deberá entregar una sintesis, que muestre el conocimiento adquirido.

Tengase en cuenta que en la evaluación de la misma se va a considerar que las respuestas a las siguientes preguntas estén incluidas. No se trata de elaborar una secuencia de respuestas, sino una síntesis elaborada.

TEMA 2: Autómatas finitos, procesamiento de unidades mofologico-léxicas, ngramas

1. Describir la relación existente entre las expresiones regulares y los autómatas finitos. A modo de ejemplo, ilustrarlo con alguna de las relacionadas con la práctica y su autómata correspondiente.
2. Describir la relación entre automátas deterministas y no deterministas. El concepto de estrategia de búsqueda
3. Describir para el ingles y el castellano (si procede) los elementos de la morfología: ejemplos de morfemas, prefijos, sufijos, infijos, circunfijos, derivación, inflexión, concordancia e enclíticos. Utilizar los recursos recomendados (lematizador de la U.de las Palmas y el MACO)
4. Describir que es un transductor secuencial. ¿qué es la morfología de dos niveles?
5. Describir en qué consiste la intersección, proyección y composición de transductores. Qué fenome lingüísticos pueden tratarse con esta aproximación.
6. Explicar qué herramientas emplearía y cómo para implementar el algoritmo de Porter.
7. Describir la relación que existe entre los conceptos corpus de entrenamiento (training set), corpus de test (test set), vocabulario abierto, vocabulario cerrado y perplejidad.
8. ¿Qué ocurre si tenemos un corpus de entrenamiento grande y un corpus de test pequeño y viceversa?
9. Describir qué problema se pretende resolver mediante las técnicas de “smoothing”.
10. ¿Qué implicaciones tiene el analisis morfológico-léxico en los motores de búsqueda textual?

Tema 3. Etiquetado

1. Describir el problema de la ambigüedad gramatical (par-of-speech).
2. ¿Qué información emplea el etiquetador gramatical EngCG ENGTWOL?
3. Si en una frase aparece únicamente un término con ambiguedad gramatical (dos etiquetas posibles), ¿por qué supone un problema de complejidad computacional decidir cual es estadísticamente más apropiada? ¿Cómo resuelve este problema de complejidad el “Hidden Markov Model” (HMM)?
4. ¿Cómo incluye la aproximación de Brill (Transformation-Based Tagging) las ventajas del modelo estocástico y de la aproximación basada en reglas?

CONSIDERACIONES

1 Terminología

Hay que utilizar una terminología correcta, y si hay términos informáticos estandarizados en español, usarlos.

Por ejemplo son terminos habituales en algoritmia, para búsqueda: "primero en profundidad" (depth first) , primero en anchura, vuelta atrás.

A veces he encontrado términos en algunos de los trabajos presentados que son erróneos: **disambigüedad** (desambiguación), **encíclico** (enclítico), ect. Por favor, es suficiente comprobarlo con una simple búsqueda.

Algunas veces no hay que fiarse sin más del traductor de google, y conviene hacer una búsqueda con la traducción propuesta para ver si se ajusta a lo que se describe. Por ejemplo "part of speech" en el tema del etiquetado sintáctico se traduce por "parte de la oración", no por "parte del discurso ".

2-Elaboración personal

Los trabajos tienen que mostrar una elaboración personal, no tiene sentido copiar y pegar ejemplos del libro o de otros textos. Sí tiene sentido incorporar ejemplos propios. Se sugieren otras herramientas o bibliografía, que conviene explorar, y una manera de mostrarlo es incluir algún ejemplo relevante que muestre que se han consultado.

3- Estilo de redacción

La redacción tiene que ser precisa, por ejemplo si se afirma que un modelo es bueno hay que justificar por qué. La redacción tiene que ser técnica, no pura literatura descriptiva.

4- Sobre la bibliografía

No puede ser una lista sin más, tiene que estar referenciada en el texto. Las entradas de Wikipedia se pueden utilizar como orientación personal, pero no son una referencia adecuada en un trabajo científico.

5- Contenido del documento

Debe quedar claro que se han entendido los conceptos y modelos que se describen. No es suficiente decir que hay varias técnicas, hay que expresar cuáles son y qué ventajas e inconvenientes presentan.

6- Antes de la entrega final

Hay que leerse el documento completo antes de entregarlo para comprobar su corrección (al menos comprobar que no tenga erratas) , y legibilidad (sucede que a veces hay párrafos con una redacción incomprensible). Además conviene comprobar que el trabajo está equilibrado en extensión en cuanto a los temas que hay que tratar, y que se cubren todos los aspectos relevantes.

A continuación pongo algunos ejemplos extraídos de trabajos de otros años, para ilustrar algunas valoraciones:

1- Sobre el resumen

Un ejemplo de resumen informativo y contextualizado

El incremento de los recursos disponibles a través de la Web y la disponibilidad de acceso móvil inalámbrico han colocado a las aplicaciones del procesamiento de la voz y del lenguaje en el centro de atención de las investigaciones tecnológicas. El procesamiento del lenguaje natural (PLN) es una disciplina que diseña, implementa e investiga sobre distintos mecanismos computacionales para establecer una comunicación efectiva entre los ordenadores y las personas. Lo que distingue el procesamiento del lenguaje de otros sistemas de procesamiento de datos es que hace uso del *conocimiento del lenguaje* en todas sus facetas: fonética, fonológica, morfológica, sintáctica, semántica, pragmática y discursiva. Este trabajo se centra en el estudio de modelos y algoritmos que emplea el PLN, como las gramáticas regulares, los autómatas finitos y los transductores. Cada uno de estos modelos se puede ampliar con sistemas probabilísticos, cuya ventaja principal es su capacidad para resolver problemas de ambigüedad. Se analizan distintos procesos de tratamiento de unidades morfológico-léxicas que sirve de apoyo al etiquetado gramatical.

Un ejemplo de resumen informativo
El presente documento resume los conceptos mas importantes de las primeras fases del procesamiento de textos: procesamiento léxico para identificar palabras, etiquetado de las mismas en función del contexto, solución de problemas de ambigüedad del lenguaje... así como las diferentes técnicas que se han utilizado en estos procesos para abordar cada uno de los retos que el tratamiento de lenguaje natural plantea para cada uno de ellos.

Un ejemplo de resumen que no es un resumen

Este documento es un resumen de los temas 2 y 3 de la asignatura "Procesamiento del Lenguaje Natural" que se corresponden con los capítulos 2, 3, 4 y 5 de la bibliografía básica.
Cada uno de los temas a tratar se ha estructurado en una sección cuyas sub-secciones serán los principales aspectos que queremos resaltar. Se incluye para cada una de las secciones una pequeña introducción a modo de resumen. Finalmente se incorpora la bibliografía que se ha consultado durante la redacción de este resumen.

2- **Sobre la necesidad de incluir detalles técnicos en las descripciones**

Copio aquí dos descripciones para que compareis

Descripción a)

El etiquetador estocástico Modelo Oculto de Markov (HMM)

Los etiquetadores estocásticos, calculan la probabilidad de que una palabra tenga una determinada etiqueta en un contexto dado,

basándose en un modelo generado automáticamente a partir de un corpus de entrenamiento.
Este modelo es muy útil, cuando nos encontramos en la situación de que en una frase aparece un término con ambigüedad gramatical, puesto que esto genera un problema de complejidad computacional, al tener que elegir entre una serie de etiquetas posibles, el problema en sí radica en que se tiene hacer el cómputo de la siguiente ecuación para saber que etiqueta elegir:
tn1= argmaxP(wn1 |tn1)P(tn1)
Para realizar esta ecuación, es necesario calcular la secuencia de etiqueta con mayor probabilidad tn1 dada una cadena wn1 , multiplicando dos probabilidades por cada secuencia de etiquetas y eligiendo la secuencia de etiqueta, la cual tiene el valor del producto más elevado. Estas probabilidades son la probabilidad da la secuencia de etiquetas P(tn1) y la probabilidad de la cadenaP(wn1 |tn1).
El modelo oculto de Markov lo soluciona haciendo dos conjeturas:

1. La probabilidad de una palabra es independiente de otras palabras y otras etiquetas que se encuentren en su entorno cercano.
2. La probabilidad de una etiqueta depende solo de la etiqueta previa.

Descripción b)
Un modelo oculto de Markov o HMM (por sus siglas del inglés, Hidden Markov Model) es un modelo estadístico en el que se asume que el sistema a modelar es un proceso de Markov de parámetros desconocidos. El objetivo es determinar los parámetros desconocidos (u ocultos, de ahí el nombre) de dicha cadena a partir de los parámetros observables. Por lo tanto, dada una cadena de palabras a etiquetar, la ambigüedad de alguna de ellas (desconocida) se resuelve mediante probabilidades de la categoría gramatical de esa palabra dadas las palabras precedente y sucesora en la cadena.

3- Definiciones correctas, con argumentos justificados, que muestren que se han comprendido los conceptos.

Por ejemplo, comparar las descripciones a y b sobre el algoritmo de Porter
Descripción a)
Transductores de estados finitos de léxico libre. El algoritmo de Porter.
Entre las estrategias existentes para la creación de transductores cuyo objetivo final es la devolución de resultados frente a consultas (por ejemplo, consultas web), y que no requieran un alto grado de cómputo, tenemos el algoritmo de Porter. Este algoritmo es el más simple y eficiente que se encarga de descomponer las palabras para encontrar su lexema.

La estrategia seguida por este algoritmo consiste en una serie de reglas de reescritura en cascada,fácilmente codificables mediante transductores de estados finitos que analizan la morfología de las palabras.
Su uso es especialmente indicado para la recuperación de la información frente a consultas en entornos con documentos pequeños. Un transductor eficiente sería capaz de devolver no sólo las palabras buscadas, sino también aquellas palabras cuyo lexema coincida.
En cuanto al diseño de este algoritmo, emplearíamos por ejemplo, una herramienta que ayude a analizar morfológicamente textos (como MACO), y una vez definido y preparado un transductor que refleje esos esquemas morfológicos, se pueden codificar mediante cualquier lenguaje de programación (por ejemplo, java o phyton). En la página del autor del algoritmo, existen módulos ya programados.

Descripción b)

Transductores sin lexicones

Aunque los transductores compuestos de lexicón y reglas ortográficas son el modelo estándar para el análisis morfológico, existen mecanismos más sencillos que no necesitan un lexicón para realizar esta y otras tareas relacionadas, como la extracción de la raíz de las palabras (steeming).
El steeming es utilizado frecuentemente en tareas de recuperación de información (IR) ya que aumenta la cobertura de los resultados obtenidos evitando dejar fuera documentos que contienen variaciones morfológicas de los términos buscados.
Uno de los algoritmos de steeming más utilizados es el de Porter [4]. Este algoritmo utiliza una serie de reglas de reescritura dispuestas en cascada que van truncando los sufijos hasta llegar a la raíz. Se basa en que las variaciones morfológicas de las palabras el inglés y muchos otros lenguajes, tienen lugar en el lado derecho de las palabras [5].
Las reglas de reescritura son del tipo:
(m>0) *FULNESS → *FUL
Donde la condición se indica entre paréntesis y a continuación se define la regla de reescritura. En este caso m es una medida que indica el número de secuencias VC que quedarán después de la reescritura donde V es una secuencia de una o más vocales y C una secuencia de una o más consonantes, FULNESS es la terminación actual de la palabra y FUL será la terminación de la palabra si se aplica la regla.
La forma más directa de implementar este algoritmo es la ejecución secuencial de estas reglas de reescritura tal y como lo desarrolló el propio Porter, pero también es posible implementarlo fácilmente mediante un transductor de estados finitos.
Este tipo de algoritmos también tienen sus inconvenientes, ya que al no disponer de un lexicón tienen más probabilidad de seleccionar una raíz equivocada para una palabra ambigua. Por ejemplo el algoritmo de Porter seleccionaría police (policía) como raíz de policy (política). Esto ha llevado a que los algoritmos de steeming actuales tiendan a ser más complejos.