Introducción al Procesamiento de Lenguaje Natural

Diciembre de 2014

Solución

Consideraciones generales
i)La prueba es sin material escrito.
ii)Escriba nombre y C.I. en todas las hojas.
iii)Numere todas las hojas.
iv)En la primera hoja, indique el total de hojas.
v)Comience cada ejercicio en una hoja nueva.
vi)Utilice las hojas de un solo lado.
vii)Entregue los ejercicios en orden
viii)El total de puntos es 40

Ejercicio 1

i) ¿Qué es un corpus paralelo? ¿Para qué se utiliza?

Un corpus paralelo es una colección de textos en dos idiomas, donde los textos en un idioma se corresponden con los textos en el otro (son traducciones del mismo texto en dos idiomas). Se utilizan en el contexto de la traducción automática estadística para construir un modelo de traducción entre los dos idiomas. A partir de los textos del corpus se aprende la frecuencia de traducción de pares de palabras o frases.

ii) Defina cuáles son los niveles de alineación que puede tener un corpus paralelo. ¿Cuál de estos niveles de alineación es más difícil de conseguir?

Los corpus paralelos pueden estar alineados en estos niveles:

- *A nivel de documento: se conoce cuáles documentos del idioma A se corresponden con los documentos del idioma B.*
- *A nivel de oración: dentro de cada documento, se conoce cuál oración en el idioma A es traducción de otra oración en otro idioma B.*
- *A nivel de palabra: se conoce además cuáles palabras dentro de una oración en el idioma A se traducen como las palabras de otra oración en el idioma B. Las alineaciones entre palabras pueden ser de muchos-a-muchos.*

En la práctica es muy difícil encontrar un corpus que esté alineado a nivel de palabras. Es más sencillo confeccionar un corpus alineado a nivel de documento y utilizar un algoritmo simple (por ejemplo Gale-Church) para alinearlo a nivel de oración. El proceso para alinear el corpus a nivel de palabra en general se realiza en conjunto con la construcción de un modelo de traducción y es un subproducto del entrenamiento.

iii) Explique brevemente en que consiste el Modelo Vectorial propuesto por Salton y mencione al menos 3 problemas que a su juicio pueden plantear inconvenientes a la hora de aplicarlo con una base de documentos

Modelo Vectorial

Es un modelo algebraico en el que se consideran un conjunto de palabras o términos como elementos característicos de los documentos. Un tema clave es seleccionar aquellos términos útiles que permitan discriminar unos documentos de otros.

No todas las palabras contribuyen con la misma importancia en la caracterización del documento.
También son poco importantes aquellas palabras que por su frecuencia de aparición en toda la colección, pierden su poder de discriminación.

La consultas y los documentos se representan por vectores en un espacio n-dimensional:

Sea {t1,t2,...,tk} el conjunto de términos

Sea {d1,d2,...,dN} el conjunto de documentos

Un documento di se modela como un vector: di = (wi1,wi2,...,wik) donde, a cada término j, en un documento di se le asigna un peso wij

A su vez, también la consulta tiene su vector de términos.

Una vez que los documentos y la consulta tienen sus vectores de términos con los correspondientes pesos, hay que calcular la similitud entre cada documento y la consulta que permita "ranquear" u ordenar aquellos documento que se asemejen más a la consulta.

3 problemas al momento de aplicarlo:

1) No toma en cuenta el orden de las palabras
2) No toma en cuenta la proximidad entre las distintas palabras
3) Los términos empleados en la consulta deberían coincidir exactamente con los términos del documento (no toma en cuenta sinonimia)

iv) ¿Qué es la selección de modelo en el contexto de un método de clasificación? Mencione y explique un método utilizado para realizarlo.

La selección de modelo consiste en elegir los mejores parámetros (dependientes del método) al entrenar un clasificador. Un método típico es la utilización de un corpus held-out: se separa parte del corpus de entrenamiento y se utiliza esta parte para evaluar clasificadores construidos entrenando en el resto del corpus, utilizando diferentes valores de parámetros.

Ejercicio 2

i) Describa brevemente (no más de diez líneas) una de las medidas de similitud entre pares de palabras vistas en el curso.

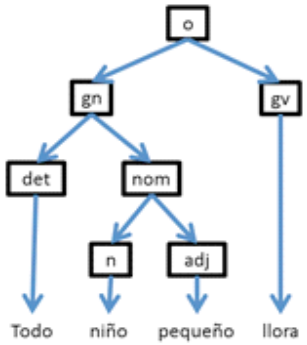
Similitud de Lesk: Para calcular esta similitud se necesita un tesauro que contenga definiciones (glosas) para cada palabra, e idealmente un corpus con muchos ejemplos del uso de las palabras. Parte de la idea de que es más probable que palabras similares compartan más palabras en su definición (o en sus ejemplos de uso). Para calcular la similitud entre w_1 y w_2, se cuenta la cantidad de palabras que se comparten en las glosas (y ejemplos) de w_1 y w_2. Por cada n palabras seguidas compartidas se suma n^2 al conteo de similitud.

ii) Considere la siguiente gramática con anotaciones semánticas:

o → gn gv	o.sem = gn.sem(gv.sem)
gn → nprop	gn.sem = nprop.sem
gn → det nom	gn.sem = det.sem(nom.sem)
nom → n adj	nom.sem = adj.sem(n.sem)
nom → n	nom.sem = n.sem
gv → v	gv.sem = v.sem
nprop → Juana	nprop.sem = λP . P(juana)
nprop → Carlos	nprop.sem = λP . P(carlos)
n → niño	n.sem = λx . niño(x)
v → salta	v.sem = λx . salta(x)
v → llora	v.sem = λx . llora(x)
adj → pequeño	adj.sem = λP .λx . P(x) ∧ pequeño(x)

det → todo	det.sem = λP . λQ . ∀x P(x) → Q(x)
det → un	det.sem = λP . λQ . ∃x P(x) ∧ Q(x)

Dibuje el árbol sintáctico y derive la expresión lógica asociada a la oración: "*Todo niño pequeño llora*".



o.sem =

[sustitución según reglas]

= gn.sem(gv.sem) =

[sustitución según reglas]

= det.sem(nom.sem)(v.sem) =

[sustitución según reglas]

= (λ P . λ Q . ∀ x P(x) →Q(x))(adj.sem(n.sem))(v.sem) =

[sustitución según reglas]

= (λ P . λ Q . ∀x P(x) → Q(x))((λ P.λ x.P(x)∧ pequeño(x))(λ x.niño(x)))(v.sem) =

[cambio de variable]

= (λ P . λ Q . ∀x P(x) → Q(x))((λ P.λ x.P(x)∧ pequeño(x))(λ y.niño(y)))(v.sem) =

[aplicación funcional λP por (λy . niño(y))]

= (λ P . λ Q . ∀x P(x) → Q(x))(λ x.(λ y.niño(y))(x)∧ pequeño(x))(v.sem) =

[aplicación funcional λy por x]

= (λ P . λ Q . ∀ x P(x) →Q(x))(λ x.niño(x)∧ pequeño(x))(v.sem) =

[cambio de variable]

$= (\lambda P . \lambda Q . \forall x\, P(x) \rightarrow Q(x))(\lambda y.niño(y) \wedge pequeño(y))(v.sem) =$

[aplicación funcional λP por (λ y.niño(y)∧ pequeño(y))]

$= (\lambda Q . \forall x(\lambda y.niño(y) \wedge pequeño(y))(x) \rightarrow Q(x))(v.sem) =$

[aplicación funcional λy por x]

$= (\lambda Q . \forall x\, niño(x) \wedge pequeño(x) \rightarrow Q(x))(v.sem) =$

[sustitución según reglas]

$= (\lambda Q . \forall x\, niño(x) \wedge pequeño(x) \rightarrow Q(x))(\lambda x.llora(x)) =$

[cambio de variable]

$= (\lambda Q . \forall x\, niño(x) \wedge pequeño(x) \rightarrow Q(x))(\lambda y.llora(y)) =$

[aplicación funcional λQ por (λy . llora(y))]

$= \forall x\, niño(x) \wedge pequeño(x) \rightarrow (\lambda y.llora(y))(x) =$

[aplicación funcional λy por x]

$= \forall x\, niño(x) \wedge pequeño(x) \rightarrow llora(x)$

Ejercicio 3

Considere la siguiente gramática, similar a la vista en el curso:

O	→	GV | GN GV
GN	→	Det Nom | Nom | GN GP
GV	→	V | V GN | V GN GP
GP	→	Prep GN
Det	→	el | la | los | las
Nom	→	Juan | café | leche | perro | tomo
V	→	tomo | corro
Prep	→	con

1. Aplique el algoritmo CKY para la entrada "*tomo café con leche*", usando la gramática G.

Transformamos la gramática a Forma Normal de Chomsky:

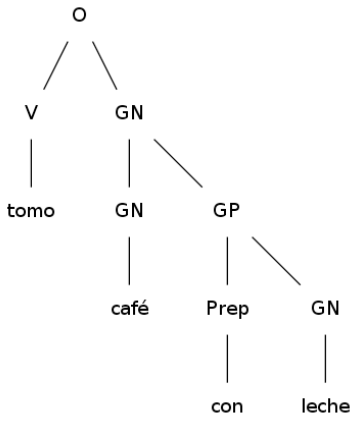
O	→	tomo | corro | V GN | Aux1 GP | GN GV
GN	→	Det Nom | Juan | café | leche | perro | tomo | GN GP
GV	→	tomo | corro | V GN | Aux1 GP
GP	→	Prep GN
Aux1	→	V GN
Det	→	el | la | los | las

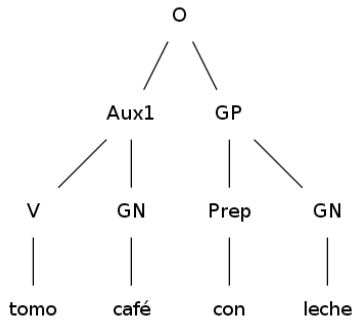
Nom → Juan | café | leche | perro | tomo
V → tomo | corro
Prep → con

Analizamos utilizando CKY

	Tomo (1)	**café (2)**	**con(3)**	**leche(4)**
0	Nom O GN GV V	GV Aux1 O		**O** GV Aux1
1		GN Nom		GN
2			Prep	GP
3				GN NOM

2. Dibuje los árboles sintácticos posibles para la oración. ¿Es posible obtener esta información del resultado del algoritmo? Justifique.





No, no es posible, ya que CKY es, en su formulación original solamente un reconocedor (no devuelve el árbol de parsing. Habría que agregar en cada celda todas las posibles formas de generar cada componente que aparece

3. ¿Cuál de los dos árboles le parece más adecuado para el ejemplo? ¿Cómo podría mejorarse el algoritmo de análisis para seleccionar el mejor?

El primero, ya que "café con leche" es un grupo nominal que incluye un grupo preposicional. El segundo árbol sería más adecuado para casos como "Tomo café con un amigo". Existen diferentes formas posibles de seleccionar: utilizando, por ejemplo información semántica, o asignando probabilidades a las producciones y maximizando la verosimilitud en un corpus de entrenamiento.

Ejercicio 4

Considere los adjetivos en el idioma ficticio Itio:

- Los adjetivos tienen una raíz (el morfema principal), diferentes sufijos, y un posible prefijo. Para este ejercicio, consideraremos solamente las raíces: *kapik* (profundo), *diak* (alto), y *sipia* (ligeramente inclinado).
- Uno de los sufijos (opcional) indica el género: *-x* (masculino), *-o* (neutro). Los adjetivos son femeninos cuando no existen sufijos, y todos los adjetivos admiten los tres géneros.
- Otro sufijo, que lo sigue, que, si está presente, indica plural: *-s* (plural)
- Para indicar la certeza del adjetivo se lo precede de un prefijo opcional: *yii-* (con seguridad), *moo-* (increíble), *pro-* (probablemente)
- Por cuestiones ortográficas, el sufijo correspondiente a plural se convierte en *-es* cuando sigue a una consonante.

a) Describa brevemente los tres componentes principales de un analizador morfológico. Describa cada componente para un hipotético analizador morfológico para los adjetivos en Itio.
b) Construya una expresión regular utilizando el álgebra de Xerox, que relacione un adjetivo con su análisis morfológico. Utilice las siguientes marcas léxicas para la salida: *+Adj* (marca de sustantivo), *+Masc* (masculino), *+Fem* (femenino), *+Neutro* (neutro), *+Singular*, *+Plural, +Seguro, +Incr, +Prob+*. Suponga que las únicas raíces son las mencionadas, pero diseñe la solución de modo que pueda ampliarse fácilmente.

Ejemplos de adjetivos en Itio:

kapikx (profundo) → kapik+Adj+Masc+Singular
yiikapikx (seguramente profundo) → +Seguro+kapik+Adj+Masc+Singular
prokapikx (probablemente profundo) → +Prob+kapik+Adj+Masc+Singular
moodiakes (increíblemente altas) → +Incr+diak+Adj+Fem+Plural+Incr
sipias (ligeramente inclinadas) → sipia+Adj+Fem+Plural

```
define root [{kapik}|{diak}|{sipia}] %+Adj:0;
define sufgenero [%+Fem:0 | %+Neutro:o | %+Masc:x];
define sufplural [%+Plural:s | %+Singular:0];
define prefcerteza [%+Seguro%+:{yii} | %+Incr%+:{moo} | %+Prob%+:{pro}];
define ortografia s -> es || k _ .#.;
define itioadj [(prefcerteza) root sufgenero sufplural] .o. ortografia;
```