Introducción al Procesamiento de Lenguaje Natural

Diciembre 2019 - Soluciones

Ejercicio 1 [8 puntos]

Indique si las siguientes proposiciones son Verdaderas o Falsas. Justifique en cada caso.

a) Mediante el estudio de la morfología derivativa se puede deducir el género y el número de una palabra.

Falso. La Morfología derivativa es el mecanismo "productivo", que combina el morfema principal (raíz) con un afijo y se obtiene una palabra de otra clase o con otro significado. Es el estudio de la morfología flexiva, de donde se puede deducir el género y número (y los demás rasgos).

b) Un tokenizador es un algoritmo que explora texto y devuelve cada palabra encontrada y su categoría gramatical.

Falso. El Tokenizador es un proceso que a partir de una secuencia de caracteres o símbolos, los agrupa en cadenas con un significado.

c) La gramática de dependencias consiste en un formalismo donde se le incorporan atributos a las reglas que indican los distintos rasgos de los constituyentes.

Falso. Las gramáticas categoriales se basan en relaciones biléxicas las palabras y no maneja el concepto de constituyente.

d) El Penn Treebank es un corpus en el cual las oraciones han sido anotadas en base a su estructura sintáctica.

Verdadero. Justamente, el Penn Treebank es un corpus de texto que anota la estructura sintáctica de sus oraciones.

Ejercicio 2 [16 puntos]

Dada las siguientes reglas de una gramática libre de contexto

O → GN GV
GN → Nom | Det Nom | GN GP
GV → V | V GN | GV GP
GP → Prep GN
Det → el | la | los | las
Nom → Maria | Juan | comedor | estar | sala
Prep → del | del | en
V → sale | come | está | salir | estar | comer

a) Aplique el algoritmo de Earley para la oración "***María está en el estar"*** para verificar si puede ser analizada.

María está en el estar

0 1 2 3 4 5

chart[0]

Y → . O [0,0] Dummy
O → . GN GV [0,0] Predict
GN → . Nom [0,0] Predict
GN → . GN GP [0,0] Predict
GN → . Det Nom [0,0] Predict

chart[1]

Nom → María . [0,1] Scanner
GN → Nom . [0,1] Complete
GN → GN . GP [0,1] Complete
O → GN . GV [0,1] Complete
GP → . Prep GN [1,1] Predict
GV → . V [1,1] Predict
GV → . V GN [1,1] Predict
GV → . GV GP [1,1] Predict

chart[2]

V → está . [1,2] Scanner
GV → V . [1,2] Complete
GV → V . GN [1,2] Complete
O → GN GV . [0,2] Complete
GV → GV . GP [1,2] Complete
GN → . Nom [2,2] Predict
GN → . GN GP [2,2] Predict
GN → . Det Nom [2,2] Predict
GP → . Prep GN [2,2] Predict

chart[3]

Prep → en . [2,3] Scanner
GP → Prep . GN [2,3] Complete
GN → . Nom [3,3] Predict
GN → . GN GP [3,3] Predict
GN → . Det Nom [3,3] Predict

chart[4]

Det → el . [3,4] Scanner
GN → Det . Nom [3,4] Complete

chart[5]

Nom → estar . [4,5] Scanner
GN → Det Nom . [3,5] Complete
GP → Prep GN . [2,5] Complete
GV → GV GP . [1,5] Complete
GV → V GN . [1,5] Complete
O → GN GV . [0,5] Complete
Y → O . [0,5] Complete

b) Realice una derivación para otra oración que sea agramatical y explique por qué lo es.

O ⇒ GN GV ⇒ Nom GV ⇒ Juan GV ⇒ Juan V GN ⇒ Juan salir GN ⇒ ***Juan salir estar***

Este ejemplo es agramatical, porque el verbo salir no está conjugado en 1era persona (el sujeto es Juan, y es quien *sale* a algún lugar o de algún lugar y además requeriría luego una preposición - "a dónde sale").

Ejercicio 3 [16 puntos]

a) Describa brevemente (no más de diez líneas) una de las técnicas vistas en el curso para resolver el problema de desambiguación del sentido de las palabras (WSD - Word Sense Disambiguation en inglés).

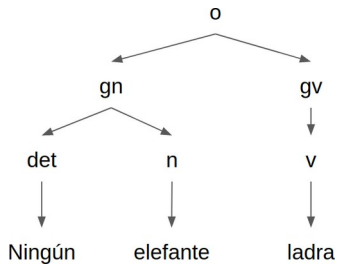
Una de las técnicas es la de *listas de decisión*: se tiene una lista ordenada de reglas a aplicar para decidir qué sentido utilizar para una palabra dado el contexto. Las reglas pueden describirse utilizando features, por ejemplo, features tipo bag of words sobre una ventana de palabras alrededor de la palabra objetivo.
Ejemplo de regla: si la palabra "dinero" pertenece a la ventana → es un banco financiero
Teniendo definido el conjunto de features, las reglas se pueden inferir de manera automática. Una forma de hacer esto es mediante el algoritmo de Yarowsky: Se calcula para cada par <sentido,feature> su poder discriminatorio. Se ordenan todas las features en la lista ubicando más arriba a la de mayor poder discriminatorio.

b) Considere la siguiente gramática con anotaciones semánticas:

o → gn gv	o.sem = gn.sem(gv.sem)
gn → det n	gn.sem = det.sem(n.sem)
gv → v	gv.sem = v.sem
gv → v gn	gv.sem = v.sem(gn.sem)
gv → neg gv	gv.sem = neg.sem(gv.vem)
nprop → Juan	nprop.sem = λ P . P(juan)
n → perro	n.sem = λ x . perro(x)
n → elefante	n.sem = λ x . elefante(x)
v → ladra	v.sem = λ x . ladra(x)
v → come	v.sem = λ P . λ y . P(λ x . come(y, x))
adj → gris	adj.sem = λ P .λ x . P(x) ∧ gris(x)
neg → no	neg.sem = λ P . λ x . ¬ P(x)
det → todo	det.sem = λ P . λ Q . ∀x P(x) → Q(x)
det → ningún	det.sem = λ P . λ Q . ∀x P(x) → ¬Q(x)
det → un	det.sem = λ P . λ Q . ∃x P(x) ∧ Q(x)

Dibuje el árbol sintáctico y derive la expresión lógica asociada a la oración:

*"**Ningún elefante ladra**"*



sustituciones según reglas

o.sem = gn.sem(gv.sem) = det.sem(n.sem)(v.sem) = (λP . λQ . ∀x P(x) → ¬Q(x)) (λx . elefante(x)) (λx . ladra(x)) =

cambio de variable

= (λP . λQ . ∀x P(x) → ¬Q(x)) (λy . elefante(y)) (λx . ladra(x)) =

aplicación funcional

= (λQ . ∀x (λy . elefante(y))(x) → ¬Q(x)) (λx . ladra(x)) =

aplicación funcional

= (λQ . ∀x elefante(x) → ¬Q(x)) (λx . ladra(x)) =

cambio de variable

= (λQ . ∀x elefante(x) → ¬Q(x)) (λy . ladra(y)) =

aplicación funcional

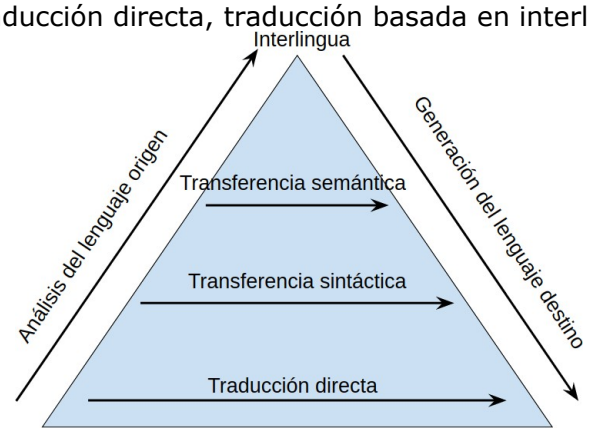
= ∀x elefante(x) → ¬(λy . ladra(y))(x) =

aplicación funcional

= ∀x elefante(x) → ¬ladra(x) =

Ejercicio 4 [20 puntos]

a) Dibuje el Triángulo de Vauquois, indicando qué representan los lados derecho e izquierdo del triángulo y ubicando dentro del triángulo los siguientes métodos de traducción automática: transferencia sintáctica, traducción directa, traducción basada en interlingua.



b) Mencione dos problemas del modelo booleano que son resueltas por el modelo vectorial. Explique brevemente cómo los resuelve.

El Modelo Booleano de RI por ejemplo no maneja el grado de relevancia de un documento y cuán "similar" en contenido sea a lo buscado ni tampoco la frecuencia de aparición de lo términos de búsqueda en los documentos. Ambos aspectos son resueltos por el Modelo Vectorial (Ver teórico)

c) Qué entiende por léxico afectivo y para qué tarea podría utilizarlo. ¿De qué forma?

Léxico afectivo consiste en listas de palabras (positivas o negativas). Típicamente se podrían utilizar en la tarea de análisis de sentimiento en métodos manuales donde se escriban reglas que considere la existencia o no de esos términos en el documento analizado.

d) Explique en qué consiste el método GloVe para construir representaciones vectoriales de palabras.

El método GloVe permite construir un repertorio de vectores de palabras a partir de un corpus. El mismo consiste en la resolución de un problema de mínimos cuadrados planteado a partir de las probabilidades de ocurrencia de las palabras. Las probabilidades de las palabras se calculan a partir de la matriz de la matriz de coocurrencias del corpus.

e) ¿En qué consiste el test de analogías para evaluar la calidad de los vectores de palabras?

El test de analogías se define como el porcentaje de aciertos sobre un conjunto de preguntas de la forma: "a" es a "b" como "c" es a ?. Este test se basa en que los pares de palabras relacionadas (bajo algunos tipos de relaciones) tienden a tener el mismo vector diferencia.