

Curso: Generación de Lenguaje Natural y Aplicaciones
 Escuela de Lingüística Computacional 2010
 26-31 de Julio 2010, Buenos Aires, Argentina

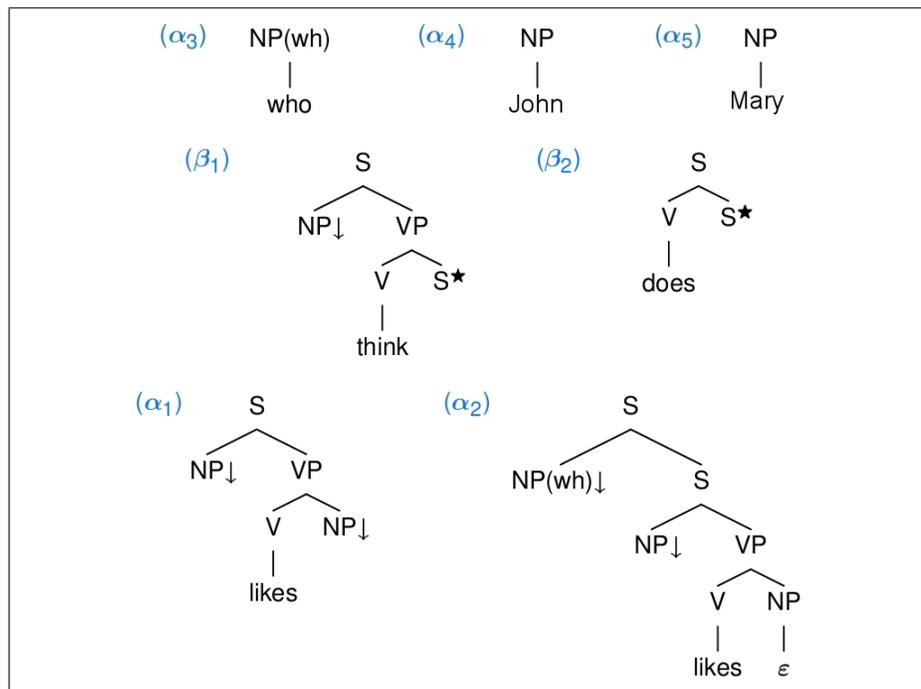
Evaluación Take Home

Entrega: **Lunes 16 de Agosto, 2010**

Pregunta 1:

1. Enumerar las etapas básicas de un sistema estandar de GLN, describiendo cada una de ellas en 2 renglones.
2. Muchos sistemas organizan estas etapas en un pipeline.
 - (a) Describir las **desventajas** de esta arquitectura (max 5 renglones).
 - (b) Mencionar explícitamente dos etapas donde la arquitectura pipeline es particularmente problemática y dar un ejemplo de generación concreto donde la calidad del texto mejoraría al no usar un pipeline.

Pregunta 2: Considerar la siguiente gramática léxica TAG presentada en clase



1. Mostrar que la frase “Who does John think Mary likes” pertenece al lenguaje de la gramática, mostrando los distintos pasos de la construcción del árbol.

2. Dar tres ejemplos de frases generadas por la gramática que no son gramaticalmente correctas¹.
3. Dar una descripción, lo más detallada posible, de las frases generadas por la gramática.
4. Si restringimos las operaciones de sustitución y adjunción de forma que no permitamos sustituciones una vez que una operación de adjunción ha sido aplicada, obtenemos el mismo lenguaje? Justifique por que.

Pregunta 3: El algoritmo incremental de Dale and Reiter para la GRE puede definirse de la siguiente forma:

<p>Input: Un conjunto de individuos con sus propiedades Un individuo target.</p> <p>Output: Un conjunto de propiedades</p> <p>Algoritmo: Empezar con un conjunto vacío Agregar propiedades del objeto target, mientras cada propiedad elimine al menos a un distractor y mientras el conjunto de distractores sea no vacío.</p>

Asumamos que en cada iteración, la propiedad a utilizar se elige en forma ‘greedy’ utilizando aquella con mayor poder discriminatorio.

1. Dar un ejemplo de un conjunto de objetos y propiedades donde para algún objeto target, la heurística greedy no lleva a la descripción más corta.
2. Extender el algoritmo para que tome un orden parcial sobre las propiedades y proponer combinaciones de este orden con el poder discriminatorio de una propiedad. Dar ejemplos.

Pregunta 4:

1. **Monos y Bananas:** un mono esta en una habitación con unas bananas colgando del techo. La habitación tiene una caja a la que el mono puede subirse para alcanzar las bananas. Inicialmente, el mono esta en la posición A, las bananas en la posición B, y la caja en la posición C. El mono y la caja tiene la propiedad ‘altura baja,’ pero si el mono se sube a la caja, su altura pasa a ser ‘alta’ que es la misma altura a la que están las bananas. Las acciones que el mono puede realizar son *Ir* de una posición a otra. *Empujar* un objeto de un lugar a otro. *TreparA* y *BajarseDe* un objeto. Y *Tomar* y *Soltar* un objeto. La acción *Tomar* resulta en el objeto estando en posesión del mono, siempre y cuando el mono y el objeto esten en el mismo lugar y a la misma altura.
 - (a) Especificar el estado inicial descrito, y un estado final en donde el mono tiene las bananas.
 - (b) Especificar las seis acciones mencionadas, usando STRIPS.

¹Cualquier noción razonable de ‘gramaticalidad’ está bien, no vamos a ser estrictos con eso. Pero intenten que los ejemplos sean lo mas diferente posible entre ellos.

2. Considerar el siguiente problema de planning

```

Init(At(C1, SFO) ∧ At(C2, JFK) ∧ At(P1, SFO) ∧ At(P2, JFK)
    ∧ Cargo(C1) ∧ Cargo(C2) ∧ Plane(P1) ∧ Plane(P2)
    ∧ Airport(JFK) ∧ Airport(SFO))
Goal(At(C1, JFK) ∧ At(C2, SFO))
Action(Load(c, p, a),
    PRECOND: At(c, a) ∧ At(p, a) ∧ Cargo(c) ∧ Plane(p) ∧ Airport(a)
    EFFECT: ¬ At(c, a) ∧ In(c, p))
Action(Unload(c, p, a),
    PRECOND: In(c, p) ∧ At(p, a) ∧ Cargo(c) ∧ Plane(p) ∧ Airport(a)
    EFFECT: At(c, a) ∧ ¬ In(c, p))
Action(Fly(p, from, to),
    PRECOND: At(p, from) ∧ Plane(p) ∧ Airport(from) ∧ Airport(to)
    EFFECT: ¬ At(p, from) ∧ At(p, to))

```

Construir los niveles 0, 1 y 2 del grafo de planning (graphplan) indicando mutexes.

3. Demostrar la siguiente propiedad de un grafo de planning

Un literal que no aparece en el último nivel de un grafo de planning (totalmente expandido, i.e., cuando el grafo ‘leveled off’) no puede ser alcanzado desde el estado inicial por ninguna secuencia de acciones.

Pregunta 5: Durante el último día de clase discutimos distintas métricas que pueden utilizarse para evaluar la calidad de un sistema de generación de instrucciones en un ambiente virtual. Supongamos que queremos evaluar (independientemente una de otra) las siguientes características de un sistema

- Eficiencia y eficacia
- Entretenimiento
- Generación de background común entre el sistema y el usuario.

Dar para cada característica un listado ordenado de las métricas ‘Top 5.’ Es decir, mencionar en cada caso, en orden de mejor a peor, 5 métricas que evalúan la característica indicada. Para cada una de las listas dadas, argumentar el orden elegido².

²Pueden incluir métricas no mencionadas en clase, argumentando cómo obtenerlas y por qué son mejores que otras.