

Ejercicio 4. WSD

Referencias: <http://www.nltk.org/book/ch02.html> (punto 5)

1) Implementar, usando NLTK y Python, el algoritmo de Lesk simplificado para desambiguar el sentido de las palabras (WSD). La función recibirá una palabra y una frase que la contenga y decidirá el mejor sentido para esa palabra. Las frases serán en inglés y se deberá eliminar de la frase, de la glosa y de los ejemplos de cada sentido las 'stopwords'.

EJEMPLO:

Sentence: "Yesterday I went to the *bank* to withdraw the money and the credit card did not work"

Word: *bank*

```
function SIMPLIFIED LESK(word,sentence) returns best sense of word  
  best-sense <- most frequent sense for word  
  max-overlap <- 0  
  context <- set of words in sentence  
  for each sense in senses of word do  
    signature <- set of words in the gloss and examples of sense  
    overlap <- COMPUTEOVERLAP (signature,context)  
    if overlap > max-overlap then  
      max-overlap <- overlap  
      best-sense <- sense  
end return (best-sense)
```

(figura extraída de la Wikipedia, http://en.wikipedia.org/wiki/Lesk_algorithm)

2) Implementar un algoritmo similar para la desambiguación semántica utilizando Word Embeddings y una distancia de similitud semántica como la distancia coseno (https://en.wikipedia.org/wiki/Cosine_similarity)