Desambigüación del sentido de las palabras (WSD)

Miguel A. Alonso

Departamento de Computación, Facultad de Informática, Universidade da Coruña

Índice

Word Sense Disambiguation (WSD)

- Seleccionar el sentido correcto para una palabra en una frase
- Útil en muchas tareas de PLN:
  - traducción automática

WordNet Sense	Spanish Translation	Roget Category	Target Word in Context
bass^4	lubina	FISH/INSECT	...fish as Pacific salmon and striped **bass** and...
bass^4	lubina	FISH/INSECT	...produce filets of smoked **bass** or sturgeon...
bass^7	bajo	MUSIC	...exciting jazz **bass** player since Ray Brown...
bass^7	bajo	MUSIC	...play **bass** because he doesn't have to solo...

  - clasificación de textos
  - búsqueda de respuestas
  - recuperación de información
  - ...

Enfoques

- Tipos de WSD:
  - **Muestra léxica** (lexical sample): se intenta desambiguar el sentido de unas pocas palabras escogidas previamente
  - **Todas las palabras** (all-words): se trata de desambiguar el sentido de todas las palabras de un texto
- Múltiples enfoques: basados en corpus etiquetados, basados en thesaurus, aprendizaje semi-supervisado, aprendizaje no supervisado, ...
- En el fondo se trata de una tarea de **clasificación** y por ello se pueden aplicar en cada uno de los enfoques los diversos algoritmos de clasificación desarrollados a lo largo de los años

Lectura recomendada

- Daniel Jurafsky and James H. Martin
  capítulo 20 de *Speech and Language Processing. Second Edition*
  Pearson Education, Upper Saddle River, New Jersey, 2009

Evalución

- Los estándares los definen las tareas SEMEVAL (antes SENSEVAL)
- **Intrínseca** o "in vitro" (independiente de la aplicación, preferida)
  - sense accuracy (porcentaje de palabras etiquetadas correctamente)
  - precision y recall en el caso de que se permita al algoritmo de WSD no decidir en algunas palabras
- **Extrínseca**, orientada a la tarea o "in vivo" (dependiente de la aplicación)
- Cotas:
  - **Baseline**: el sentido más frecuente o el algoritmo de Lesk
  - **Ceiling**: acuerdo mutuo entre anotadores (varía del 75% al 90%)
- Más fácil que WSD sobre palabras reales es WSD sobre **pseudo-palabras**. Ejemplo: convertir todas las ocurrencias de "manzana" y "puerta" en "manzana-puerta"; la tarea consiste en adivinar el origen de cada ocurrencia de manzana-puerta.

Aprendizaje supervisado en WSD: Corpus

- Para WSD de muestras léxicas:
  - SENSEVAL-1 (34 palabras desambiguadas)
  - SENSEVAL-2 (73 palabras desambiguadas)
  - SENSEVAL-3 (57 palabras desambiguadas)
- Para WSD de todas las palabras:
  - SemCor (234.000 palabras del Borwon Corpus etiquetadas manualmente con sentidos de WordNet)
  - SENSEVAL-3 (5. 000 palabras del Penn Treebank, de elllas 2.081 palabras con contenido está etiqeutadas manualmente con sentidos de WordNet)
- SENSEVAL derivó a partir de SENSEVAL-3 en SEMEVAL

Aprendizaje supervisado en WSD: Features

- Se extraen del **contexto** de la palabra, normalmente una ventana de N elementos:
  - las palabras en sí
  - los lemas de las palabras
  - las etiquetas (PoS) de las palabras
  - ...
- Dos tipos:
  - Collocational features (incluye la posición relativa en la ventana)
  - Bag-of-words features (sólo se indica si están o no presentes)

Ejemplo de aprendizaje supervisado: Naive Bayes

- Elige el sentido $\hat{s}$ más pobable (de entre el conjunto de sentidos S) en base a un vector de features $\vec{f}$:

$$\hat{s} = \arg\max_{s \in S} P(s \mid \vec{f})$$

$$\hat{s} = \arg\max_{s \in S} \frac{P(\vec{f} \mid s) P(s)}{P(\vec{f})}$$

- Asumiendo independencia (naively):

$$P(\vec{f} \mid s) \approx \prod_{j=1}^{n} P(\vec{f_j} \mid s)$$

y dado que $P(\vec{f})$ es igual para todos los sentidos:

$$\hat{s} = \arg\max_{s \in S} P(s) \prod_{j=1}^{n} P(\vec{f_j} \mid s)$$

- El entrenamiento se hace "contando" (máxima verosimilitud)

Métodos basados en thesaurus

Una familia de algoritmos que elige el sentido cuya definición o glosa comparte más palabras con el contexto de la palabra a desambiguar

Ejemplo: "The bank can guarantee deposits will eventually cover future tuition costs in invest in adjustable-ration mortgage securities"

bank[1]	Gloss:	a financial institution that accepts deposits and channels the money into lending activities
	Examples:	"he cashed a check at the bank", "that bank holds the mortgage on my home"
bank[2]	Gloss:	sloping land (especially the slope beside a body of water)
	Examples:	"they pulled the canoe up on the bank", "he sat on the bank of the river and watched the currents"

Se elige el sentido 1 porque comparte las palabras "deposits" y "mortgage" con el contexto

Algoritmo de Lesk simplificado

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*)

Algoritmo Lesk Corpus

- Generalmente las definiciones de los diccionarios son cortas y hay poco solapamiento con el contexto
- **Solución**:
  - si hay un pequeño corpus etiquetado disponible, añadir las palabras de las frases en las que aparece un determinado sentido, en su definición.
  - ponderar cada palabra i de la definición y el contexto en función de su $\text{idf}_i = \log\left(\frac{N}{n_i}\right)$
- Es por tanto un enfoque híbrido basado en thesaurus y corpus

Aprendizaje mínimamente supervisado en WSD

- Cuando no se dispone de grandes corpus etiquetados o grandes thesaurus/diccionarios
- Ambos son recursos costosos, que requieren de mano de obra especializada
- **Solución**: métodos que sólo requieren pequeños recursos etiquetados a mano

- El más conocido de los métodos mínimamente supervisados, o métodos semi-supervisados, es el bootstrapping

Bootstrapping

1. Se parte de un pequeño corpus semilla Λ_0 etiquetado y un gran corpus V_0 no etiquetado
2. Se entrena un clasificar inicial en Λ_i (inicialmente Λ_0)
3. Se seleccionan las instancias F de V_i (inicialmente V_0) más fiables, y se obtienen
   - $V_{i+1} = V_i - F$
   - $\Lambda_{i+1} = \Lambda_i \cup F$
4. Se repiten los pasos 2 u 3 hasta que se alcanza una ratio de errores lo suficientemente pequeña en el corpus de entrenamiento o bien hasta que no se puedan extraer más instancias fiables del corpus no etiquetado

Algoritmo de Yarowsky

- Conjunto inicial de semillas:
  - Heurística 1: **un sentido por colocación** (buscar palabras que co-ocurren y desambiguan cada sentido)
  - Heurística 2: **un sentido por discurso** (las distintas apariciones de una palabra en un texto suelen corresponder al mismo sentido)
- Métrica para extraer buenos ejemplos del corpus no etiquetado
  - **log-lilehood ratio**

$$\left| \log \frac{P(sense_1 \mid f)}{P(sense_2 \mid f)} \right|$$

Aprendizaje no supervisado en WSD

El sentido de una palabra viene defindo por las palabras que aparecen en su contexto

Sea $\vec{w} = (f_1, f_2, \ldots f_n)$ el **vector contexto** de la palabra w (las n palabras que más aparecen en un ventana de tamaño m alrededor de las ocurrencias de la palabra w)

1. Para cada ocurrencia w_i de la palabra w, calcular su vector contexto $\vec{c}$
2. Usar un algoritmo de clustering para agrupar los vectores contexto $\vec{c}$. Cada **cluster** es un sentido de w
3. Calcular el centroide de cada cluster. El **vector centroide** $\vec{s_j}$ es el representante del sentido j de w

Desambiguación del sentido con aprendizaje no supervisado

Para cada ocurrencia t de una palabra w:

1. Calcular el vector contexto $\vec{c}$ para t
2. Recuperar todos los vectores centroide $\vec{s_j}$ de w
3. Asignar t al sentido correspondiente al vector $\vec{s_j}$ más carcano a $\vec{c}$

Para ello se puede utilizar cualquier algoritmo de clustering. Lo más habitual es utilizar clustering aglomerativo

La evaluación suele ser extrínseca ya que no se dispone de un corpus etiquetado

The end