

## Processament del Llenguatge Humà 1. Estructura del document



UNIVERSITAT POLITÈCNICA DE CATALUNYA  
BARCELONATECH

Facultat d'Informàtica de Barcelona



# Índice

Estructura del documento

Identificación del idioma

- 1 Estructura del documento
  - Búsqueda de zonas textuales
  - Tokenización
  - Separación de oraciones

- 2 Identificación del idioma

# Índice

Estructura del documento

Búsqueda de zonas textuales

Identificación del idioma

- 1 Estructura del documento
  - Búsqueda de zonas textuales
    - Tokenización
    - Separación de oraciones
- 2 Identificación del idioma

# Tipos de documentos

Estructura del documento

Búsqueda de zonas textuales

Identificación del idioma

- Documentos que contienen texto:
  - Documentos estructurados (p.e., páginas web que solo muestran tablas)
  - Documentos semiestructurados (p.e., páginas web que contienen piezas de texto plano, figuras y tablas)
  - Documentos de texto plano (p.e., ficheros de texto, emails, tweets, transcripciones orales)

Acceder a las zonas textuales contenidas en páginas web puede ser necesario para su posterior uso.

# Analizadores XML

- Transforman un documento XML/HTML/XHTML en un árbol de objetos estándares.
- Proporcionan una interfaz para gestionar esos árboles
- Las zonas textuales pueden ser extraídas del árbol usando el intefaz

```
<?xml version="1.0"?>
<doc type="novel" title="The green apple" >
<chapter id="1" >
<p>There are lots of trees in Amsteel Hill. I remember
going there and spend all the morning climbing those
trees, trying to get as many apples as possible.</p>
<p> James always wanted to come with me but he
was too young to get climbing.</p>
...
</doc>
```

Con *ElementTree.py*

```
for c in root:
    lp=c.findall('p')
    for p in lp:
        print p.text
```

# Índice

Estructura del documento

Tokenización

Identificación del idioma

- 1 Estructura del documento
  - Búsqueda de zonas textuales
  - Tokenización
  - Separación de oraciones
- 2 Identificación del idioma

# Objetivo de la tokenización

- Objetivo: fragmentar texto plano en unidades básicas.
- Uso: RI, clasificación de documentos, división en oraciones, identificación del idioma, normalización de texto  
...
- Diferentes unidades básicas dependiendo de la tarea,
  - Tokenizaciones *naive* : separar por espacios y puntuaciones que ocurran después de cadenas alfanumericas
  - Tokenizaciones complejas: nombres, clíticos, abreviaciones, **colocaciones**...

# Objetivo de la tokenización

- Objetivo: fragmentar texto plano en unidades básicas.
- Uso: RI, clasificación de documentos, división en oraciones, identificación del idioma, normalización de texto  
...
- Diferentes unidades básicas dependiendo de la tarea,
  - Tokenizaciones *naive* : separar por espacios y puntuaciones que ocurran después de cadenas alfanumericas
  - Tokenizaciones complejas: nombres, clíticos, abreviaciones, **colocaciones**...

Definiciones:

***N-grama de palabras***: secuencia de palabras que ocurren en texto

***Colocación***: secuencia de palabras que frecuentemente ocurren en texto. P.e.: "break a leg", "tomar un respiro"



# Ejemplos de tokenización

Espacios	punt. sufixa	Abr.	Clíticos	Coloc.	normalizado
Of course	Of course	Of course	Of course	Of_course	Of_course
I'll	I'll	I'll	I 'll	I 'll	I will
go to	go to	go to	go to	go to	go to
U.P.C.	U.P.C	U.P.C	U.P.C	U.P.C	Universitat...
"Daily,	Daily	Daily	Daily	Daily	Daily
Mr.	Mr	Mr.	Mr.	Mr.	Mister
John Smith..."	John Smith	John Smith	John Smith	John Smith	John_Smith
	...	...	...	...	...
	"	"	"	"	"

Estructura del documento

Tokenización

Identificación del idioma

# Ejemplos de tokenización

Espacios	punt. sufixa	Abr.	Clíticos	Coloc.	normalizado
Of course	Of course	Of course	Of course	Of_course	Of_course
I'll	I'll	I'll	I 'll	I 'll	I will
go to	go to	go to	go to	go to	go to
U.P.C.	U.P.C	U.P.C	U.P.C	U.P.C	Universitat...
"Daily,	Daily	Daily	Daily	Daily	Daily
Mr.	Mr	Mr.	Mr.	Mr.	Mister
John Smith..."	John Smith	John Smith	John Smith	John Smith	John_Smith
	...	...	...	...	...
	"	"	"	"	"

Estructura del documento

Tokenización

Identificación del idioma

## Ejemplos de tokenización

Espacios	punt. sufixa	Abr.	Clíticos	Coloc.	normalizado
Of course	Of course	Of course	Of course	Of_course	Of_course
I'll	I'll	I'll	I 'll	I 'll	I will
go to	go to	go to	go to	go to	go to
U.P.C.	U.P.C	U.P.C	U.P.C	U.P.C	Universitat...
"Daily,	Daily	Daily	Daily	Daily	Daily
Mr.	Mr	Mr.	Mr.	Mr.	Mister
John Smith..."	John Smith	John Smith	John Smith	John Smith	John_Smith
	...	...	...	...	...
	"	"	"	"	"

Problemas: ¿Texto no estándar? ¿Chino? ¿Japonés?

Estructura del documento

Tokenización

Identificación del idioma

# Índice

Estructura del documento

Separación de oraciones

Identificación del idioma

- 1 Estructura del documento
  - Búsqueda de zonas textuales
  - Tokenización
  - Separación de oraciones
- 2 Identificación del idioma

# Objetivo de la división del texto en oraciones

Estructura del documento

Separación de oraciones

Identificación del idioma

- Objetivo: reconocer las fronteras de las oraciones en texto plano.
- Dependiente del idioma  
P.e.: Aleman: "Mein 2. Semester kommt bald zu Ende."  
P.e.: ¿Chino tradicional?
- Depende del dominio  
P.e.: "It is expressed as  $(x=1)?$  T.add('-') : T.add(x)."
- Métodos:
  - Reglas hechas a mano
  - Aprendizaje automático supervisado
  - Métodos no supervisados
- Input:
  - Tokenización *Naïve*, dependiendo del método particular.
  - Por comodidad, aquí usaremos *espacios y puntuación sufija*

" I'll go to U.P.C. "Daily, Mr. John Smith..." "

→ " I 'll go to U.P.C . " Daily , Mr . John Smith ... " "

# Problemas de la separación de oraciones

Estructura del documento

Separación de oraciones

Identificación del idioma

## ■ Abreviaciones y acrónimos

P.e.: "I will meet with Mr. Smith to talk about it."

P.e.: "Lisa run 25 km. She ended up in N.Y."

Cómo las detectamos?

## ■ Elipsis

P.e.: "There're different methods (A, B, ...) but ..."

## ■ Citas

P.e.: "'Stop!' he shouted."

## ■ Números ordinales (p.e., alemán)

## ■ Casos especiales:

Ex: "We have some variables. x stands for the weight,"

# Método: Reglas hechas a mano

Estructura del documento

Separación de oraciones

Identificación del idioma

- Clases de abreviaciones  
(mes, unidad métrica, título, dirección, ...)

P.e.: `TITLE='(Mr | Mrs | Dr ... )'`

- Expresiones regulares

P.e.: `/ $TITLE (\.) /` →  $t \notin s\_boundary$

P.e.: `/ [A-Z] (\.) /` →  $t \notin s\_boundary$

P.e.: `/ ([?!]{2,}) /` →  $t \in s\_boundary$

P.e.: `/ (\.\.\.) [A-Z]/` →  $t \in s\_boundary$

P.e.: `/ ([?!.] [A-Z]/` →  $t \in s\_boundary$

Problema:

- Adaptación a nuevos idiomas altamente costosa

# Método: AA supervisado

- Paradigma más usado (ME, SVM, Perceptron, ...-métodos discriminativos-)
- Requiere un corpus anotado manualmente. Normalmente,  $e^+, e^- = [',', '!', '?']$  y algunos tokens anteriores y posteriores.
- Cada  $e$  es representado como un conjunto de atributos que depende del método, idioma y dominio (normalmente atributos binarios).

Problema:

- Requiere grandes cantidades de ejemplos (entre decenas y centenas de miles)

Estructura del documento

Separación de oraciones

Identificación del idioma



## Método: AA supervisado

Ejemplos de atributos usados en el estado del arte:

tok-1\_X: 1º token antes del '.' es X

tok-2\_X: 2º token antes del '.' es X

tok+1\_X: 1º token después del '.' es X

len\_tok-1\_X: longitud del 1º token antes del '.' es X

len\_tok-2\_X: longitud del 2º token antes del '.' es X

len\_tok+1\_X: longitud del 1º token después del '.' es X

[up|lo|cap|num]\_tok-1: el 1º token antes del '.' es Upper, Lower, CAP, Number

[up|lo|cap|num]\_tok-2: igual para el 2º token antes del '.'

[up|lo|cap|num]\_tok+1: igual para el 1º token después del '.'

class\_tok-1\_X: la clase de abreviación del 1º token antes del '.' es X

...

# Método: AA supervisado

Ejemplo de anotación manual y extracción de atributos binarios

I 'll go to U.P.C. " Daily , Mr John Smith ... "

$e^+$	tok-1_U.P.C	$e^-$	tok-1_Mr
	len_tok-1_3		len_tok-1_2
	CAP_tok-1		up_tok-1
	tok-2_to		tok-2,
	len_tok-2_2		len_tok-2_1
	lo_tok-2		class_tok-1_title
	tok+1_"		tok+1_John
	len_tok+1_1		len_tok+1_4
			up_tok+1

Estructura del documento

Separación de oraciones

Identificación del idioma

# Método: AA no supervisado

Estructura del documento

Separación de oraciones

Identificación del idioma

- Requiere un corpus no anotado de gran volumen
  - Facilmente adaptable a nuevos idiomas
- Foco principal: abreviaciones y elipsis
- Basado en heurísticas y estadísticas de uso en el corpus:
  - 1 ¿Qué tokens son abreviaciones?
  - 2 ¿Cuándo se considera que un '.' es final de oración?
- P.e.: Punkt [Kiss and Strunk, 2006]

# Método: AA no supervisado: Punkt

## 1 Punkt: ¿Consideramos el token $t$ como una abreviación?

Usa los siguientes heurísticos:

- $t' = \langle t, . \rangle$  debería ser una colocación
- $t$  debería ser corto
- $t$  podría incluir puntuación (acrónimo)
- $t$  no debería ser una palabra común anterior a '.' la mayoría de veces (p.e., verbos en turco)

# Método: AA no supervisado: Punkt

## 1 Punkt: ¿Consideramos el token $t$ como una abreviación?

Usa los siguientes heurísticos:

- $t' = \langle t, . \rangle$  debería ser una colocación
- $t$  debería ser corto
- $t$  podría incluir puntuación (acrónimo)
- $t$  no debería ser una palabra común anterior a '.' la mayoría de veces (p.e., verbos en turco)

## 2 Punkt: ¿Consideramos el '.' de la abreviación como final de oración?

Uno de los siguientes heurísticos debe cumplirse:

- $t'' = siguiente(t')$  suele ser principio de oración (las obtenidas en [1])
- $t''$  empieza con mayúscula, ocurre al menos una vez con minúscula en el corpus pero nunca con mayúscula dentro de oraciones (las obtenidas en [1])

# Ejercicio

Explica porqué Punkt falla (rojo) o no (azul) con los siguientes textos:

- " "Good night!", said Laura. "
- " Abbrev. is a common abbreviation of abbreviation. "
- " We are meeting with our mr. You are late! "
- " We are meeting with our Mr. However, we'll finish soon."

Demo de separadores de oraciones:

<http://text-processing.com/demo/tokenize/>

# Índice

Estructura del documento

Identificación del idioma

- 1 Estructura del documento
  - Búsqueda de zonas textuales
  - Tokenización
  - Separación de oraciones

- 2 Identificación del idioma

# Objetivo de la identificación del idioma

Estructura del documento

Identificación del idioma

- Puede verse como un problema de clasificación.
- Dado un documento,  $d$ , y un conjunto de idiomas,  $L = \{l_1, \dots, l_k\}$ , asignar  $l_i$  a  $d$ .
- Método:
  - $\hat{d}$  = representación( $d$ )
  - $M(\hat{d}) \rightarrow l_i$
- El modelo  $M$  puede aprenderse a partir de corpus de aprendizaje  $T = \{T_i\}_{1 \dots k}$  donde  $T_i = \{d_x | d_x \text{ escrito en } l_i\}$ :
  - Métodos de AA supervisado
  - Modelos de lenguaje estadísticos

Survey: <https://arxiv.org/pdf/1804.08186.pdf>



# Modelos de lenguaje para la identificación del idioma

$$M = \{P^{l_i}\}_{l_i \in L}$$

$P^{l_i}(\hat{d})$ : probabilidad de  $\hat{d}$  de pertenecer a  $l_i$

$$l_i = \operatorname{argmax}_{l \in L} (P^l(\hat{d}))$$

$P^{l_i}(\hat{d}) \approx P^{T_i}(\hat{d})$ : probabilidad de  $\hat{d}$  observando los datos en  $T_i$

Estructura del documento

Identificación del idioma

# Modelos de lenguaje para la identificación del idioma

$$M = \{P^{l_i}\}_{l_i \in L}$$

$P^{l_i}(\hat{d})$ : probabilidad de  $\hat{d}$  de pertenecer a  $l_i$

$$l_i = \operatorname{argmax}_{l \in L} (P^l(\hat{d}))$$

$P^{l_i}(\hat{d}) \approx P^{T_i}(\hat{d})$ : probabilidad de  $\hat{d}$  observando los datos en  $T_i$

- 1 ¿Cómo representamos  $\hat{d}$ ?
- 2 ¿Cómo computamos  $P^{T_i}(\hat{d})$ ?

# Modelos de lenguaje para la identificación del idioma

$$M = \{P^{l_i}\}_{l_i \in L}$$

$P^{l_i}(\hat{d})$ : probabilidad de  $\hat{d}$  de pertenecer a  $l_i$

$$l_i = \operatorname{argmax}_{l \in L} (P^l(\hat{d}))$$

$P^{l_i}(\hat{d}) \approx P^{T_i}(\hat{d})$ : probabilidad de  $\hat{d}$  observando los datos en  $T_i$

- 1 ¿Cómo representamos  $\hat{d}$ ?
- 2 ¿Cómo computamos  $P^{T_i}(\hat{d})$ ?

Depende del tipo de modelo particular.

Modelo más usado: [modelo de lenguaje unigrama](#)

# Modelo de lenguaje unigrama para la identificación del idioma

## 1 ¿Cómo representamos $\hat{d}$ ?

$\hat{d} = e_1, \dots, e_s$  es una secuencia de unigramas  $e_i$

- Palabras (después de tokenización *Naïve*) o
- $n$ -gramas de caracteres (no se requiere tokenización)
  - $n$  fija (la más usada) or
  - $n$  variable (mejora la precisión, empeora la eficiencia)

# Modelo de lenguaje unigrama para la identificación del idioma

## 1 ¿Cómo representamos $\hat{d}$ ?

$\hat{d} = e_1, \dots, e_s$  es una secuencia de unigramas  $e_j$

- Palabras (después de tokenización *Naïve*) o
- $n$ -gramas de caracteres (no se requiere tokenización)
  - $n$  fija (la más usada) or
  - $n$  variable (mejora la precisión, empeora la eficiencia)

## 2 ¿Cómo computamos $P^{T_i}(\hat{d})$ ?

Hipótesis: independencia entre unigramas,  $e_j$ .

$$P^T(\hat{d}) = P^T(e_1, \dots, e_s) = \prod_{j=1}^s P^T(e_j)$$

$$\log P^T(\hat{d}) = \sum_{j=1}^s \log P^T(e_j)$$

Posibles estimadores de  $P^T(e_j)$ :

- Estimador de máxima verosimilitud (Maximum Likelihood Estimator - MLE)
- Técnicas de suavizado (smoothing)

# Unigram language models for language identification

## Maximum Likelihood Estimator

$$P^T(e_j) \approx P_{MLE}^T(e_j) = \frac{c_T(e_j)}{N_T}$$

$c_T(x)$ : #ocurrencias de  $x$  en el corpus de entrenamiento  $T$

$N_T$ : #ocurrencias de unigramas en el corpus de entrenamiento  $T$

# Unigram language models for language identification

## Maximum Likelihood Estimator

$$P^T(e_j) \approx P_{MLE}^T(e_j) = \frac{c_T(e_j)}{N_T}$$

$c_T(x)$ : #ocurrencias de  $x$  en el corpus de entrenamiento  $T$

$N_T$ : #ocurrencias de unigramas en el corpus de entrenamiento  $T$

- Problema: dispersión de datos. Unigramas  $e_j$  no observados causan el fallo del modelo. MLE no es adecuado para PLN.

# Unigram language models for language identification

## Maximum Likelihood Estimator

$$P^T(e_j) \approx P_{MLE}^T(e_j) = \frac{c_T(e_j)}{N_T}$$

$c_T(x)$ : #ocurrencias de  $x$  en el corpus de entrenamiento  $T$

$N_T$ : #ocurrencias de unigramas en el corpus de entrenamiento  $T$

- Problema: dispersión de datos. Unigramas  $e_j$  no observados causan el fallo del modelo. MLE no es adecuado para PLN.

- Ejemplo:

$P^{[en]}('The\ doctor\ tell\ us\ about\ his\ quadriplegia')$ ?

$c_{[en]}('quadriplegia') = 0 \implies P_{MLE}^{[en]}('quadriplegia') = 0$

$\implies P^{[en]}('The\ doctor\ tell\ us\ about\ his\ quadriplegia') = 0 !!$



# Unigram language models for language identification

Estructura del documento

Identificación del idioma

Técnicas de smoothing:

Reservar masa de probabilidad para  $e_j$  no observados en  $T$ ;

P.e., Ley de Lidstone (LID)

$$P^T(e_j) \approx P_{LID}^T(e_j) = \frac{c_T(e_j) + \lambda}{N_T + \lambda B} \quad \text{usually, } \lambda = 0,5$$

$B$ : #unigramas potencialmente observables

## Ejercicio

Supón que tenemos un identificador de idiomas para inglés y catalán, basado en unigramas de palabras, con las siguientes estadísticas

$w_i$	a	he	mail	sent	to	mordorian
English language model [en]						
$c_{[en]}(w_i)$	17.000	10.000	3.900	850	25.000	0
$N_{[en]}=1.300.000$	$B_{[en]}=22.600$					
Catalan Language model [ca]						
$c_{[ca]}(w_i)$	21.000	11.900	420	910	750	0
$N_{[ca]}=1.100.000$	$B_{[ca]}=36.800$					

- Computa  $P^{[en]}$  and  $P^{[ca]}$  usando MLE y LID para los siguientes textos:
  - "he"
  - "he sent a"
  - "he sent a mail"
  - "he sent a mail to a mordorian"
- ¿Qué idioma es identificado por cada estimador para cada texto?
- Explica los efectos de la longitud del texto