1.2 MODELOS ESTADÍSTICOS

MODELOS ESTADÍSTICOS

- **Modelos de n-gramas**
- **Métodos de suavizado**
  - Plano y añade uno
  - Interpolación Lineal
  - Interpolación No lineal
  - Back-off
- **Modelos basados en categorías**
  - Definición y tipos de clases
  - Agrupamiento automático
- **Modelos dinámicos**
  - Cache
  - Triggers

MODELOS ESTADÍSTICOS

- **Modelos de n-gramas**
- **Métodos de suavizado**
  - Plano y añade uno
  - Interpolación Lineal
  - Interpolación No lineal
  - Back-off
- **Modelos basados en categorías**
  - Definición y tipos de clases
  - Agrupamiento automático
- **Modelos dinámicos**
  - Cache
  - Triggers

Modelos de N-gramas

Como hemos visto en la introducción,

sea $W = \{w_1, w_2, \ldots, w_{|W|}\}$ un conjunto de unidades lingüísticas;

sea $A = A_1 A_2 \ldots A_{|A|}$ la secuencia acústica obtenida a partir de la señal vocal con los procesos de adquisición y preproceso;

nos interesa encontrar la secuencia de unidades lingüísticas tal que:

$$\hat{w} = arg\,max_{w \in W^+} P(w)P(A|w)$$

El Modelo Acústico será el encargado del cálculo de $P(A|w)$

El *Modelo de Lenguaje* será el encargado del cálculo de $P(w)$

Se puede descomponer siguiendo la regla de la cadena como:

$$P(w) = \prod_{i=1}^{n} P(w_i|w_1 w_2 \cdots w_{i-1}) =$$

$$P(w_1) \cdot P(w_2|w_1) \cdot P(w_3|w_1 w_2) \cdots P(w_n|w_1 \cdots w_{n-1})$$

Modelos de N-gramas

Algunas dependencias quedan representadas con el modelo propuesto. Ejemplo: supongamos que el resultado de una decodificación acústica asigna probabilidades semejantes a las frases:

$$the \left\{ \begin{array}{c} pig \\ big \end{array} \right\} dog$$

Si $P(pig|the) = P(big|the)$ entonces la elección de una u otra depende de la palabra *dog*.

$$P(the\ pig\ dog) = P(the).P(pig|the).P(dog|the\ pig)$$

$$P(the\ big\ dog) = P(the).P(big|the).P(dog|the\ big)$$

Si p(dog| the big) > p(dog | the pig) el modelo ayuda a decodificar la frase correctamente

Modelos de N-gramas

$$P(w) = \prod_{i=1}^{n} P(w_i | w_1 w_2 \cdots w_{i-1})$$

- PROBLEMA**:** Necesidad de un elevado número de muestras de aprendizaje.
- ALTERNATIVA**:** Definición de clases de equivalencia (**N-gramas**).

  En base a la historia parcial, pertenecen a la misma clase aquellas historias que coinciden en las últimas *N-1* palabras.
- N-GRAMAS**:** Sea *w* una frase del lenguaje a modelar, de longitud *m*. Se hace la aproximación:

$$P(w) = \prod_{i=1}^{m} P(w_i \mid w_1^{i-1}) \approx \prod_{i=1}^{m} P(w_i \mid w_{i-N+1}^{i-1})$$

Texto ejemplo:

$ llegó con tres heridas &
$ la del amor &
$ la de la muerte &
$ la de la vida &
$ con tres heridas viene &
$ la de la vida &
$ la del amor &
$ la de la muerte &
$ con tres heridas yo &
$ la de la vida &
$ la de la muerte &
$ la del amor &

Bigramas

	amor	con	de	del	heridas	la	llegó	muerte	tres	vida	viene	yo	&	
$		2				9	1							12
amor													3	3
con									3					3
de						6								6
del	3													3
heridas											1	1	1	3
la			6	3				3		3				15
llegó		1												1
muerte													3	3
tres					3									3
vida													3	3
viene													1	1
yo													1	1
														57+12

$ llegó con tres heridas &
$ la del amor &
$ la de la muerte &
$ la de la vida &
$ con tres heridas viene &
$ la de la vida &
$ la del amor &
$ la de la muerte &
$ con tres heridas yo &
$ la de la vida &
$ la de la muerte &
$ la del amor &

Bigramas (máxima verosimilitud)

	amor	con	de	del	heridas	la	llegó	muerte	tres	vida	viene	yo	&	
$		2/12				9/12	1/12							12
amor													3/3	3
con									3/3					3
de						6/6								6
del	3/3													3
heridas											1/3	1/3	1/3	3
la			6/15	3/15				3/15		3/15				15
llegó		1/1												1
muerte													3	3
tres					3/3									3
vida													3/3	3
viene													1/1	1
yo													1/1	1
														57+12

$$P(\text{la de la vida}) = P(\text{la} \mid \$) \cdot P(\text{de} \mid \text{la}) \cdot P(\text{la} \mid \text{de}) \cdot P(\text{vida} \mid \text{la}) \cdot P(\& \mid \text{vida})$$
$$= 9/12 \cdot 6/15 \cdot 1 \cdot 3/15 \cdot 1$$

$$P(\text{la de la de la del amor}) \neq 0$$

$$P(\text{con la muerte}) = P(\text{con} \mid \$)\ P(\text{la} \mid \text{con}) \cdot P(\text{muerte} \mid \text{la}) \cdot P(\& \mid \text{muerte}) = 0$$

N-gramas

Número de parámetros excesivamente grande incluso con esta simplificación.

Unigramas	$p(w)$	$\|W\|$
Bigramas	$p(w_2 \mid w_1)$	$\|W\|^2$
Trigramas	$p(w_3 \mid w_1 w_2)$	$\|W\|^3$

Ejemplo: Wall Street Journal: Corpus de 38 millones de palabras

| *W* | = 5.000 palabras, 125.000 millones de trigramas.

- Test 1: 5.000 palabras, 21% trigramas no vistos.
- Test 2: 20.000 palabras, 32% trigramas no vistos.

Aproximando a Shakespeare

Unigram

To him swallowed confess hear both. Which. Of save on trail for are ay device and rote life have
Every enter now severally so, let
Hill he late speaks; or! a more to leg less first you enter
Are where exeunt and sighs have rise excellency took of.. Sleep knave we. near; vile like

Bigram

What means, sir. I confess she? then all sorts, he is trim, captain.
Why dost stand forth thy canopy, forsooth; he is this palpable hit the King Henry. Live king. Follow.
What we, hath got so she that I rest and sent to scold and nature bankrupt, nor the first gentleman?

Trigram

Sweet prince, Falstaff shall die. Harry of Monmouth's grave.
This shall forbid it should be branded, if renown made it empty.
Indeed the duke; and had a very good friend.
Fly, and will rid me these news of price. Therefore the sadness of parting, as they say, 'tis done.

Quadrigram

King Henry.What! I will go seek the traitor Gloucester. Exeunt some of the watch. A great banquet serv'd in;
Will you not tell me who I am?
It cannot be but so.
Indeed the short and the long. Marry, 'tis a noble Lepidus.

Aproximando a Shakespeare

Shakespeare como corpus

- N=884.647 palabras, W=29.066
- Shakespeare produce 300.000 tipos de bigramas de W^2= 844 millones posibles.
  - Por tanto 99,96% de los posibles bigramas no han sido nunca vistos (tienen entrada cero en las tablas)
- Cuatrigramas peor

Ejemplo de N-gramas

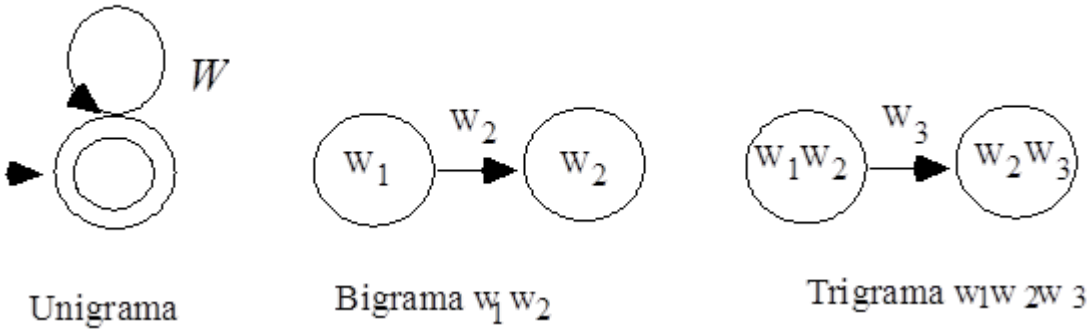
- Probabilidades estimadas por máxima verosimilitud entrenadas a partir de un corpus de textos literarios de la escritora J. Austen y testeadas con otra novela de la misma escritora.
- El conjunto de entrenamiento contiene 617.091 palabras con un vocabulario de 14.585.

In person	*she*		*was*		*inferior*		*to*		*both*		*sisters*	
1-gram	$P(\cdot)$		$P(\ )$		$P(\ )$		$P(\ )$		$P(\cdot)$		$P(\ )$	
1	the	0.034	the	0 034	the	0.034	the	0.034	the	0 034	the	0.034
2	to	0 032	to	0.032	to	0 032	**to**	**0.032**	to	0 032	to	0 032
3	and	0 030	and	0 030	and	0 030			and	0 030	and	0.030
4	of	0 029	of	0.029	of	0 029			of	0.029	of	0.029
.												
8	was	0 015	**was**	**0.015**	was	0.015			was	0 015	was	0 015
.												
13	**she**	**0.011**			she	0 011			she	0.011	she	0.011
.												
254					both	0 0005			**both**	**0.0005**	both	0 0005
.												
435					sisters	0 0003					**sisters**	**0.0003**
.												
1701					**inferior**	**0.00005**						
2-gram	$P(\ \|person)$		$P(\ \|she)$		$P(\ \|was)$		$P(\cdot\|inferior)$		$P(\ \|to)$		$P(\cdot\|both)$	
1	and	0.099	had	0.141	not	0.065	**to**	**0.212**	be	0 111	of	0 066
2	who	0 099	**was**	**0.122**	a	0 052			the	0 057	to	0.041
3	to	0 076			the	0 033			her	0 048	in	0.038
4	in	0.045			to	0.031			have	0.027	and	0.025
.												
23	**she**	**0.009**							Mrs	0 006	she	0.009
.												
41									what	0.004	**sisters**	**0.006**
.												
293									**both**	**0.0004**		
.												
∞					**inferior**	**0**						
3-gram	$P(\ \|In,person)$		$P(\cdot\|person\ she)$		$P(\ \|she\ was)$		$P(\cdot\|was,inf.)$		$P(\ \|inferior\ to)$		$P(\cdot\|to,both)$	
1	UNSEEN		did	0.5	not	0.057	UNSEEN		the	0.286	to	0.222
2			**was**	**0.5**	very	0 038			Maria	0 143	Chapter	0.111
3					in	0 030			cherries	0.143	Hour	0 111
4					to	0.026			her	0.143	Twice	0.111
.												
∞					**inferior**	**0**			**both**	**0**	**sisters**	**0**
4-gram	$P(\ \|u,I,p)$		$P(\ \|I,p,s)$		$P(\ \|p,s,w)$		$P(\ \|s,w,i)$		$P(\cdot\|w,i,t)$		$P(\ \|i,t,b)$	
1	UNSEEN		UNSEEN		in	1 0	UNSEEN		UNSEEN		UNSEEN	
.												
∞					**inferior**	**0**						

Estimación de las probabilidades

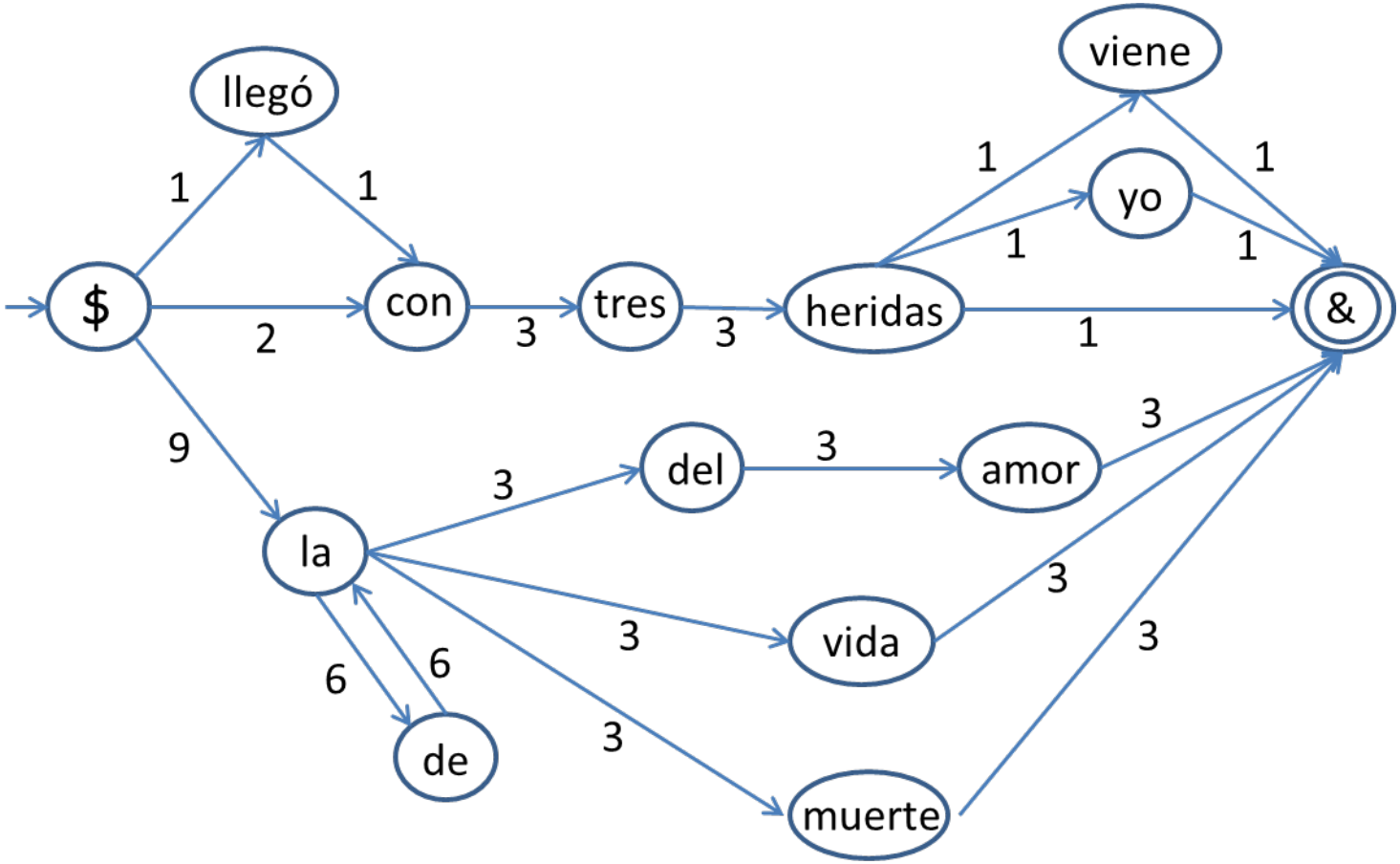
Vamos a suponer que el modelo de N-gramas se ha modelado con un autómata finito (AF).

- En el caso de **unigramas** hay un único estado con transiciones sobre sí mismo con todas las palabras del vocabulario.
- En el caso de **bigramas** hay tantos estados como palabras en el vocabulario, y las transiciones representan los bigramas.
- En el caso de **trigramas** los estados representan secuencias de dos palabras, que representan todas las combinaciones de historias encontradas en el corpus de aprendizaje. Las transiciones representan los trigramas.

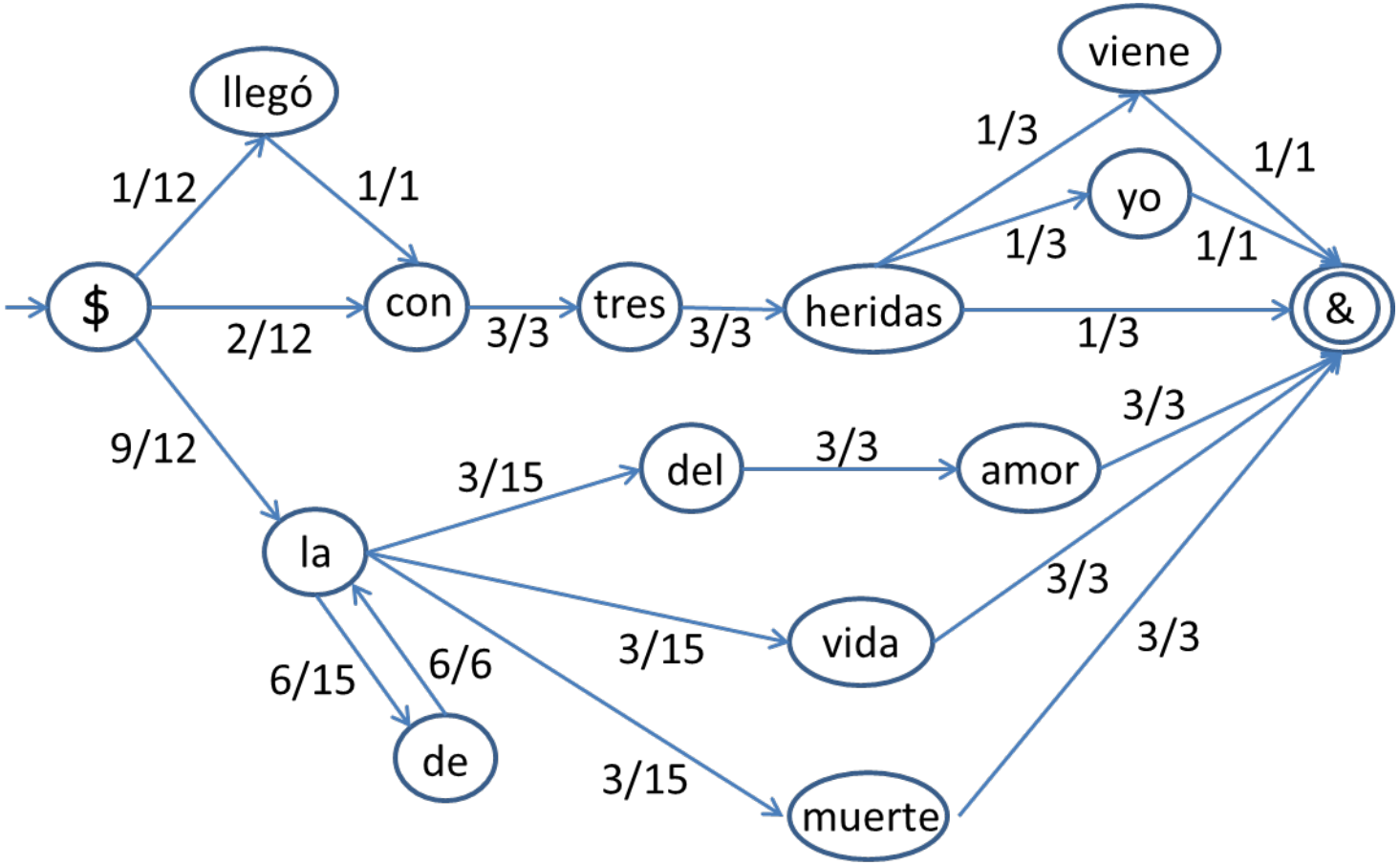


Unigrama Bigrama $w_1 w_2$ Trigrama $w_1 w_2 w_3$

Representación en forma de AF del ejemplo



Representación en forma de AF del ejemplo



Estimación de las probabilidades

Supongamos que se dispone de una muestra de entrenamiento, sobre la que se ha estimado un modelo de N-gramas, representado como un AF.

- Sea q un estado del autómata, y sea c(q) el número total de eventos (N-gramas) observados cuando el modelo se encuentra en el estado q.
- Sea c(w|q) el número de veces que ha sido observada la palabra w estando el modelo en el estado q.
- Sea P(w|q) la probabilidad de observación de la palabra w condicionada al estado q.
- Sea W_q el conjunto de palabras observadas cuando el modelo se encuentra en el estado q.
- Sea W el vocabulario total del lenguaje a modelar.

ESTIMACIÓN POR MAXIMA VEROSIMILITUD: $$P_{ML}(w|q) = \frac{c(w|q)}{c(q)}$$

En un modelo de bigramas: $$P_{ML}(w_i|w_{i-1}) = \frac{c(w_{i-1}w_i)}{c(w_{i-1})}$$

Este criterio asigna una probabilidad cero a los eventos no vistos ===> problemas de cobertura

Estimación de las probabilidades

Sea N_r el número de eventos que han sido observados r veces.

Sea N el número total de eventos observados.

La situación más usual en modelado del lenguaje es:

$$N_1 < N << N_0 < \text{número total de eventos}$$

Ejemplo: Berkeley Restaurant corpus

Muestra las frecuencias de algunos de los bigramas del Berkeley Restaurant corpus (9.332 frases y un vocabulario de 1.446 palabras)

	i	want	to	eat	chinese	food	lunch	spend
i	5	827	0	9	0	0	0	2
want	2	0	608	1	6	6	5	1
to	2	0	4	686	2	0	6	211
eat	0	0	2	0	16	2	42	0
chinese	1	0	0	0	0	82	1	0
food	15	0	15	0	1	4	0	0
lunch	2	0	0	0	0	1	0	0
spend	1	0	1	0	0	0	0	0

Berkeley Restaurant corpus:

- can you tell me about any good cantonese restaurants close by
- mid priced thai food is what i'm looking for
- tell me about chez panisse
- can you give me a listing of the kinds of food that are available

Ejemplo: Berkeley Restaurant corpus

Normalizando por unigramas

i	want	to	eat	chinese	food	lunch	spend
2533	927	2417	746	158	1093	341	278

Muestra las correspondientes probabilidades

	i	want	to	eat	chinese	food	lunch	spend
i	0.002	0.33	0	0.0036	0	0	0	0.00079
want	0.0022	0	0.66	0.0011	0.0065	0.0065	0.0054	0.0011
to	0.00083	0	0.0017	0.28	0.00083	0	0.0025	0.087
eat	0	0	0.0027	0	0.021	0.0027	0.056	0
chinese	0.0063	0	0	0	0	0.52	0.0063	0
food	0.014	0	0.014	0	0.00092	0.0037	0	0
lunch	0.0059	0	0	0	0	0.0029	0	0
spend	0.0036	0	0.0036	0	0	0	0	0

MODELOS ESTADÍSTICOS

- **Modelos de n-gramas**
- **Métodos de suavizado**
  - Plano y añade uno
  - Interpolación Lineal
  - Interpolación No lineal
  - Back-off
- **Modelos basados en categorías**
  - Definición y tipos de clases
  - Agrupamiento automático
- **Modelos dinámicos**
  - Cache
  - Triggers

Métodos de suavizado

La solución que se propone consiste en reservar una cierta cantidad de probabilidad P_u descontada de la probabilidad de los eventos observados en la muestra, para repartirla entre los eventos no observados.

SUAVIZADO PLANO

Distribución uniforme de P_u entre los eventos no vistos.

$$P^S(w \mid q) = \begin{cases} P'_{ML}(w \mid q) & si \;\; c(w \mid q) \neq 0 \\ P_u \dfrac{1}{|W| - |W_q|} & si \;\; c(w \mid q) = 0 \end{cases}$$

donde *s* indica que se trata de un modelo suavizado, y *P'* es una alteración de la estimación de Máxima Verosimilitud que debe cumplir:

$$\sum_{\forall w \in W : c(w \mid q) \neq 0} P'_{ML}(w \mid q) = 1 - P_u$$

Métodos de suavizado

Se han definido diferentes métodos para calcular esta masa de probabilidad y para descontarla de las estimaciones correspondientes a los eventos observados.

$r = f_{MLE}$	$f_{empirical}$	f_{Lap}	f_{del}	f_{GI}	N_r	T_r
0	0.000027	0.000295	0.000037	0.000027	74 671 100 000	2 019 187
1	0.448	0.000589	0.396	0.446	2 018 046	903 206
2	1.25	0.000884	1.24	1.26	449 721	564 153
3	2.24	0.00118	2.23	2.24	188 933	424 015
4	3.23	0.00147	3.22	3.24	105 668	341 099
5	4.21	0.00177	4.22	4.22	68 379	287 776
6	5.23	0.00206	5.20	5.19	48 190	251 951
7	6.21	0.00236	6.21	6.21	35 709	221 693
8	7.21	0.00265	7.18	7.24	27 710	199 779
9	8.26	0.00295	8.18	8.25	22 280	183 971

Table 6.4 Estimated frequencies for the AP data from Church and Gale (1991a). The first five columns show the estimated frequency calculated for a bigram that actually appeared r times in the training data according to different estimators: r is the maximum likelihood estimate, $f_{empirical}$ uses validation on the test set, f_{Lap} is the 'add one' method, f_{del} is deleted interpolation (two-way cross validation, using the training data), and f_{GT} is the Good-Turing estimate. The last two columns give the frequencies of frequencies and how often bigrams of a certain frequency occurred in further text.

Métodos de suavizado

AÑADE UNO (SUAVIZADO DE LAPLACE)

Un método sencillo de suavizado consiste en añadir 1 a TODOS los contadores de la matriz antes de proceder a la normalización para obtener las probabilidades.

	i	want	to	eat	chinese	food	lunch	spend
i	6	828	1	10	1	1	1	3
want	3	1	609	2	7	7	6	2
to	3	1	5	687	3	1	7	212
eat	1	1	3	1	17	3	43	1
chinese	2	1	1	1	1	83	2	1
food	16	1	16	1	2	5	1	1
lunch	3	1	1	1	1	2	1	1
spend	2	1	2	1	1	1	1	1

Métodos de suavizado

AÑADE UNO (SUAVIZADO DE LAPLACE)

$$P^*(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)+1}{C(w_{n-1})+V}$$

	i	want	to	eat	chinese	food	lunch	spend
i	0.0015	0.21	0.00025	0.0025	0.00025	0.00025	0.00025	0.00075
want	0.0013	0.00042	0.26	0.00084	0.0029	0.0029	0.0025	0.00084
to	0.00078	0.00026	0.0013	0.18	0.00078	0.00026	0.0018	0.055
eat	0.00046	0.00046	0.0014	0.00046	0.0078	0.0014	0.02	0.00046
chinese	0.0012	0.00062	0.00062	0.00062	0.00062	0.052	0.0012	0.00062
food	0.0063	0.00039	0.0063	0.00039	0.00079	0.002	0.00039	0.00039
lunch	0.0017	0.00056	0.00056	0.00056	0.00056	0.0011	0.00056	0.00056
spend	0.0012	0.00058	0.0012	0.00058	0.00058	0.00058	0.00058	0.00058

Métodos de suavizado

AÑADE UNO (SUAVIZADO DE LAPLACE)

Si reproducimos los contadores se puede observar una gran diferencia con los contadores iniciales (C(want to) pasa de 608 a 238). Hay demasiada proporción de la masa de probabilidad que se reparte entre los eventos no vistos.

$$c^*(w_{n-1}w_n) = \frac{[C(w_{n-1}w_n)+1] \times C(w_{n-1})}{C(w_{n-1})+V}$$

	i	want	to	eat	chinese	food	lunch	spend
i	3.8	527	0.64	6.4	0.64	0.64	0.64	1.9
want	1.2	0.39	238	0.78	2.7	2.7	2.3	0.78
to	1.9	0.63	3.1	430	1.9	0.63	4.4	133
eat	0.34	0.34	1	0.34	5.8	1	15	0.34
chinese	0.2	0.098	0.098	0.098	0.098	8.2	0.2	0.098
food	6.9	0.43	6.9	0.43	0.86	2.2	0.43	0.43
lunch	0.57	0.19	0.19	0.19	0.19	0.38	0.19	0.19
spend	0.32	0.16	0.32	0.16	0.16	0.16	0.16	0.16

Métodos de suavizado

INTERPOLACIÓN LINEAL

NOTACIÓN:

Sea q un estado del modelo a suavizar.

Sea q* el estado correspondiente del modelo suavizador.

El valor suavizado se obtiene por una combinación lineal de los valores dados por los modelos “a suavizar” y el “suavizador”.

$$P^S(w \mid q) = \lambda_q P(w \mid q) + (1 - \lambda_q) P(w \mid q\,*)$$

con λ_q próximo a 1 cuando $P(w \mid q)$ es fiable y a 0 cuando no lo es.

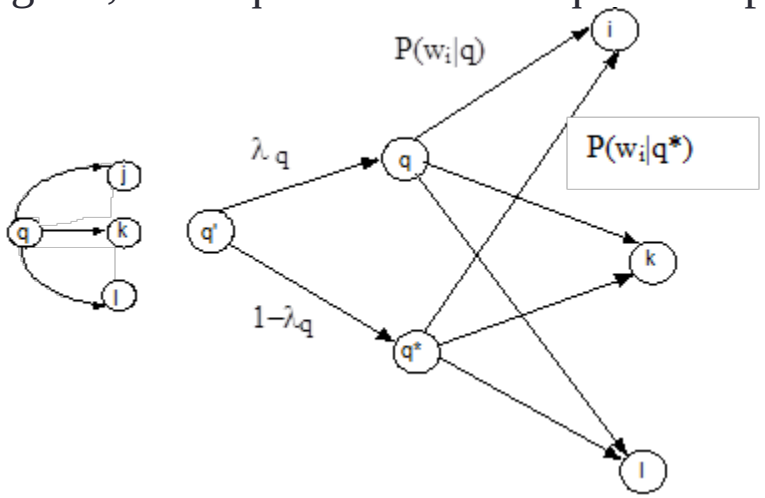
Métodos de suavizado

INTERPOLACIÓN LINEAL

1. En (Jelinek,85) se propone una modificación del Forward-backward que incluye la técnica de leaving-one-out, denominada ***deleted interpolation***, para la estimación de los parámetros λ_q.

Consideremos una fuente de Markov, cuyos parámetros han sido estimados a partir de la secuencia de datos b_1^m. Sea $P(w_i|q) \quad i=1,\cdots,l$ la probabilidad de observación de w_i en el estado *q*, y $P(w_i|q^*) \quad i=1,\cdots,l$ la del modelo suavizador.

La ecuación de la interpolación se interpreta en términos de una fuente de Markov asociada a la fuente original, en la que cada estado q es reemplazado por tres estados: *q*, *q'* y *q**.



Los parámetros de la nueva fuente se estiman por F-B, pero con ciertas variantes. Los datos b_1^m se dividen en *n* bloques de forma que para $t=1,\cdots,n$ las λ_q se estiman a partir del bloque t-ésimo, mientras que las estimaciones de las probabilidades sobre *q* y *q** se basarán en los *n-1* bloques restantes.

Métodos de suavizado

INTERPOLACIÓN LINEAL

2. En (Ney,91) se presenta un método de interpolación lineal más sencillo: para garantizar la no existencia de probabilidades nulas basta con añadir una cierta cantidad a la cuenta de cada evento *qw* y normalizar convenientemente (*floor method*).

$$P^S(qw) = \frac{c(qw) + BP^*(qw)}{B + N}$$

La cantidad total añadida *B* puede ser repartida en función de una distribución $P^*(qw)$.

Se puede expresar como una interpolación lineal:

$$P^S(qw) = (1-\lambda)P(qw) + \lambda P^*(qw) \quad con \quad \lambda = \frac{B}{B+N}$$

En este caso sólo aparece un parámetro de interpolación, por lo que su estimación no requiere del Forward-backward. Si se usa la técnica de leaving-one-out para la estimación del parámetro, se obtiene que: $\lambda \cong \frac{N_1}{N}$

La probabilidad condicional se obtiene como: $P^S(w \mid q) = \frac{P^S(qw)}{P^S(q)}$

Métodos de suavizado

INTERPOLACIÓN NO LINEAL

En una interpolación lineal se calcula una media ponderada de una distribución más específica y una más general, y los pesos respectivos son independientes de los contadores *c(qw)*. Cada contador es reducido en $\lambda c(qw)$.

Como alternativa, se define una función general de descuento *d* que se sustrae de cada contador de frecuencias $c(q|w)$ obteniéndose la siguiente fórmula de suavizado:

$$P^S(w \mid q) = \frac{\max[c(w \mid q) - d(w \mid q), 0]}{c(q)} + Q(d)P(w \mid q^*)$$

El descuento total $Q(d)$ depende de la función de descuento *d*, y se define como:

$$Q(d) = \frac{1}{c(q)} \sum_{\forall w \in W : c(w|q) \neq 0} d(w \mid q)$$

Con el objetivo de que el descuento afecte menos a los eventos bien estimados, se propone un descuento fijo *D* de forma que $0 < D \leq 1$. Esta técnica se conoce como “descuento absoluto”.

$$P^S(w \mid q) = \frac{\max[c(w \mid q) - D, 0]}{c(q)} + \frac{D \mid W_q \mid}{c(q)} P(w \mid q^*)$$

NOTA: si se hace $D=1$ se tratan igual los eventos vistos una vez como los no vistos.

Métodos de suavizado

INTERPOLACIÓN NO LINEAL

EXPERIMENTOS: interpolación lineal versus la no lineal

- Un corpus en inglés con más de un millón de palabras y un vocabulario de unas 50.000 palabras,
- y otro corpus en alemán con unas 950000 palabras y un vocabulario de unas 14.000 palabras.

Los mejores resultados obtenidos para ambas interpolaciones muestran una mejora de un 10 % en la perplejidad del conjunto de prueba de la interpolación no lineal frente a la lineal.

Métodos de suavizado

ESTIMADOR DE GOOD-TURING

La principal idea consiste en hacer uso de los contadores de los eventos vistos una vez para el cálculo de los no observados en la muestra.

$$P_T(qw) = \frac{c^*(qw)}{N} \quad \text{con} \quad c^*(qw) = [c(qw)+1]\frac{N_{c(qw)+1}}{N_{c(qw)}}$$

Según esta fórmula la cantidad de probabilidad asignada a eventos no vistos es: $1 - \sum_{\forall qw: c(qw)>0} P_T(qw) = \frac{N_1}{N}$

Según la fórmula de Turing el descuento para cada evento es:

$$\frac{c(qw)}{N} - \frac{c^*(qw)}{N} = (1 - d(qw))\frac{c(qw)}{N} \quad \text{con} \quad d(qw) = \frac{c^*(qw)}{c(qw)}$$

La estimación se puede expresar como:

$$P_T(qw) = d(qw)\frac{c(qw)}{N}$$

Métodos de suavizado

ESTIMADOR DE GOOD-TURING

Ejemplo de aplicación del estimador de Turing para un corpus de 8 especies de peces en el cual se han observado sólo 6 de dichas especies (entre ellas trout) y otras dos especies (bass y catfish) no han sido observadas. En total han sido observados 18 eventos.

	unseen (bass or catfish)	trout
c	0	1
MLE p	$p = \frac{0}{18} = 0$	$\frac{1}{18}$
c^*		$c^*(\text{trout}) = 2 \times \frac{N_2}{N_1} = 2 \times \frac{1}{3} = .67$
GT p^*_{GT}	$p^*_{\text{GT}}(\text{unseen}) = \frac{N_1}{N} = \frac{3}{18} = .17$	$p^*_{\text{GT}}(\text{trout}) = \frac{.67}{18} = \frac{1}{27} = .037$

La probabilidad para un evento visto una vez (como trout) pasa de .06 (1/18) a .037. El valor P_{GT}(unseen) se debe repartir entre los eventos no vistos.

Métodos de suavizado

ESTIMADOR DE GOOD-TURING

Dos ejemplos de aplicación del estimador Good-Turing: para el corpus Associated Press newswire y el corpus Berkeley Restaurant, donde se calcula el nuevo contador de los bigramas para las primeras 7 frecuencias.

AP Newswire			Berkeley Restaurant		
c (MLE)	N_c	c^* (GT)	c (MLE)	N_c	c^* (GT)
0	74,671,100,000	0.0000270	0	2,081,496	0.002553
1	2,018,046	0.446	1	5315	0.533960
2	449,721	1.26	2	1419	1.357294
3	188,933	2.24	3	642	2.373832
4	105,668	3.24	4	381	4.081365
5	68,379	4.22	5	311	3.781350
6	48,190	5.19	6	196	4.500000

Figure 4.8 Bigram "frequencies of frequencies" and Good-Turing re-estimations for the 22 million AP bigrams from Church and Gale (1991) and from the Berkeley Restaurant corpus of 9332 sentences.

Métodos de suavizado

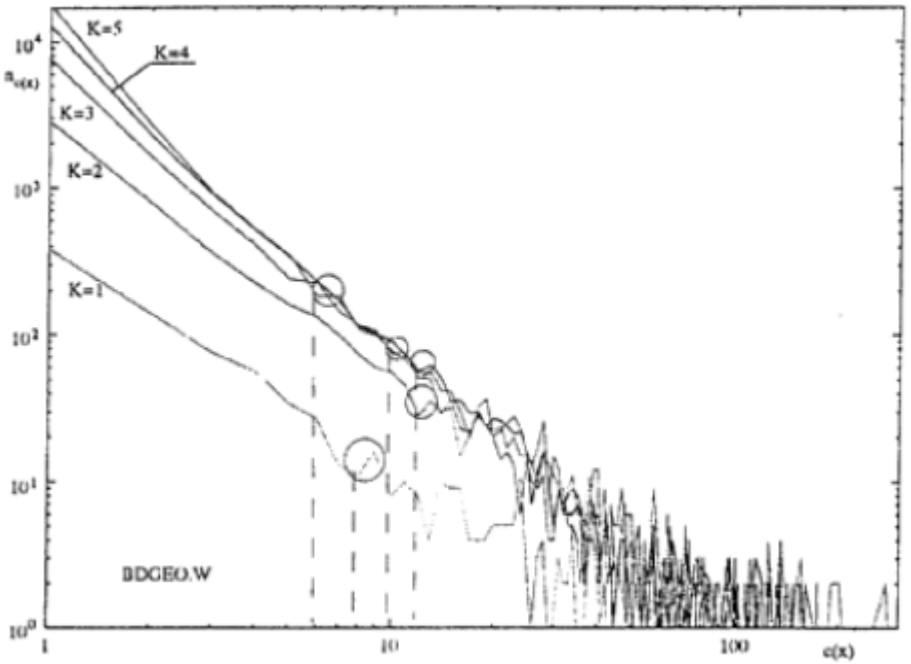
ESTIMADOR DE GOOD-TURING

Una correcta aplicación del método exige que el número de muestras de aprendizaje sea elevado para que la función $N_{c(x)}$ sea monótona decreciente con el aumento de $c(x)$, de forma que el cociente $(N_{c(qw)+1})/(N_{c(qw)})$ tenga siempre un valor menor que la unidad, comportándose como un descuento.

En (Bordel, 96) se presentan unos experimentos para estudiar $N_{c(x)}$ en función de $c(x)$. Se toman secuencias de palabras de longitudes de 1 a 5 obtenidas de 50.000 palabras de BDGEO, y se representa $N_{c(x)}$ en función de $c(x)$.

Para $K=1$ se observa que $N_8>N_7$, por lo que la fórmula de Turing sólo se puede aplicar para contadores entre 1 y 7.

El problema anterior se soluciona tomando un umbral r igual o inferior a 7.



Métodos de suavizado

MÉTODO BACK-OFF

La propuesta de Katz consiste en sustituir el valor $d(qw)$ por:

$$d(qw) = \begin{cases} 1 & para \quad c(qw) > r \\ \dfrac{\left(\dfrac{c^*(qw)}{c(qw)} - \dfrac{(r+1)N_{r+1}}{N_1} \right)}{\left(1 - \dfrac{(r+1)N_{r+1}}{N_1} \right)} & para \quad c(qw) \leq r \end{cases}$$

Con ello se consigue que manteniendo un descuento total de (N_1 / N), cuentas superiores al umbral *r* se estiman por el criterio de Máxima Verosimilitud y las inferiores o iguales a *r* son decrementadas.

Métodos de suavizado

MÉTODO BACK-OFF

Definiremos la probabilidad asignada a los eventos vistos como:

$$P_{T'}(w \mid q) = \begin{cases} P_T(w \mid q) & si \quad c(qw) \leq r \\ P_{ML}(w \mid q) & si \quad c(qw) > r \end{cases}$$

Veamos ahora la forma de *repartir* el descuento total entre los eventos no vistos. Definiremos la probabilidad condicional de la última palabra del N-grama en función de las anteriores:

$$P^S(w \mid q) = \begin{cases} P_{T'}(w \mid q) & si\ c(qw) \neq 0 \\ \alpha(q) P^S(w \mid q^*) & si\ c(qw) = 0 \end{cases}$$

donde $$\alpha(q) = \frac{1 - \sum_{\forall w: c(qw) \neq 0} P^S(w \mid q)}{1 - \sum_{\forall w: c(qw) \neq 0} P^S(w \mid q^*)}$$

Resumiendo tendremos:

$$P^S(w \mid q) = \begin{cases} P_{ML}(w \mid q) & si\ c(qw) > r \\ P_T(w \mid q) & si\ 0 < c(qw) \leq r \\ \alpha(q) P^S(w \mid q^*) & si\ c(qw) = 0 \end{cases}$$

Métodos de suavizado

DESCUENTO WITTEN-BELL

Asumimos que una palabra de frecuencia cero no ha aparecido aún pero va a aparecer.

$\Rightarrow$ Modelamos la probabilidad de los N-gramas de frecuencia cero con la probabilidad de ver un N-grama por primera vez.

Estimamos la masa de probabilidad total de los N-gramas no vistos como:

$$\sum_{i:c_i=0} p_i = \frac{T}{N+T}$$

Donde T es el número de N-gramas distintos observados y N es el número de eventos total.

Para el caso de los bigramas:

$$\sum_{i:c(w_{i-1}w_i)=0} p\left(w_i \middle| w_{i-1}\right) = \frac{T\left(w_{i-1}\right)}{c\left(w_{i-1}\right)+T\left(w_{i-1}\right)}$$

Donde $T(w_{i-1})$ es el número de tipos de bigramas diferentes con historia w_{i-1}, y $c(w_{i-1})$ es el número de bigramas con historia w_{i-1}.

Para los bigramas observados en la muestra se aplica un descuento, de forma que la probabilidad se calcula como:

$$p\left(w_i \middle| w_{i-1}\right) = \frac{c\left(w_{i-1}w_i\right)}{c\left(w_{i-1}\right)+T\left(w_{i-1}\right)}$$

Métodos de suavizado

KNESER-NEY

Usando un suavizado por back-off para el bigrama “on Francisco” devolverá una probabilidad relativamente alta cuando no es observado en entrenamiento, ya que la probabilidad del unigrama “Francisco” es alta porque el bigrama “San Francisco” aparece muchas veces en entrenamiento.

$$P_{\texttt{Katz}}(on\ Francisco) = \begin{cases} \frac{disc(C(on\ Francisco))}{C(on)} & \text{if } C(on\ Francisco) > 0 \\ \alpha(on) \times P_{\texttt{Katz}}(Francisco) & \text{otherwise} \end{cases}$$
$$= \alpha(on) \times P_{\texttt{Katz}}(Francisco)$$

Sin embargo, la palabra “Francisco” ocurre en muy pocos contextos diferentes. La propuesta de Kneser-Ney es utilizar el número de contextos distintos en lugar del número de ocurrencias de la palabra. $|\{v|C(vw_i)>0\}|$ es el número de contextos diferentes para la palabra w_i.

$$P_{\texttt{BKN}}(w_i|w_{i-1}) = \begin{cases} \frac{C(w_{i-1}w_i)-D}{C(w_{i-1})} & \text{if } C(w_{i-1}w_i) > 0 \\ \alpha(w_{i-1}) \frac{|\{v|C(vw_i)>0\}|}{\sum_w |\{v|C(vw)>0\}|} & \text{otherwise} \end{cases}$$

Métodos de suavizado

KNESER-NEY

Experimentalment se ha demostrado que el suavizado de back-off de Katz y el de Kneser-Ney basado en back-off, que bajan a distribuciones de menor orden solamente cuando el contador es cero, no trabajan bién para eventos con contadores bajos (1 o 2).

Los modelos de suavizado basados en interpolación siempre combinan las distribuciones de mayor con las de menor orden, y típicament funcionan mejor.

El suavizado de Knese-Ney en versión interpolada es:

$$P_{\text{IKN}}(w_i|w_{i-1}) = \frac{C(w_{i-1}w_i)-D}{C(w_{i-1})} + \lambda(w_{i-1})\frac{|\{v|C(vw_i)>0\}|}{\sum_w |\{v|C(vw)>0\}|}$$

Chen y Goodman proponen una versión modificada de este suavizado que usa descuentos distintos para los distintos contadores.

MODELOS ESTADÍSTICOS

- **Modelos de n-gramas**
- **Métodos de suavizado**
  - Plano y añade uno
  - Interpolación Lineal
  - Interpolación No lineal
  - Back-off
- **Modelos basados en categorías**
  - Definición y tipos de clases
  - Agrupamiento automático
- **Modelos dinámicos**
  - Cache
  - Triggers

Modelos basados en categorías

DEFINICIÓN

Para reducir el número de parámetros a estimar, y necesitar por tanto menor número de muestras de aprendizaje, se pueden agrupar palabras en **categorías**.

En caso de un bigrama se hace la siguiente aproximación:

$$P(w_i \mid w_{i-1}) \approx P(w_i \mid c_{i-1}) \approx P(c_i \mid c_{i-1}) P(w_i \mid c_i)$$

Donde c_i es la clase (categoría) asociada a la palabra w_i. La probabilidad de la clase depende sólo del predecesor, y la probabilidad de que se observe una palabra, dada una historia, depende sólo de su clase.

- Para la estimación de estas probabilidades es necesario disponer de texto etiquetado, es decir, texto en el que cada palabra ha sido sustituida por la clase a la que pertenece.
- Una estimación del modelo a partir de una muestra etiquetada (con las mismas técnicas vistas para N-gramas en general) dará lugar a eventos no vistos, por lo que se requieren otra vez **técnicas de suavizado**.

Modelos basados en categorías

TIPOS DE CATEGORIAS

1) Por conocimiento lingüístico:

El caso más habitual es el **POS** (Parts of Speech).

Problemas:

- Ambigüedad. El etiquetado (tagging) de esas palabras es un problema abierto.
- Hay diferentes clasificaciones de POS hechas por los lingüistas.
- Estas clasificaciones puede que no sean útiles para un modelo de lenguaje.

2) Por conocimiento del dominio.

Por ejemplo, en ATIS los nombres de aeropuertos, nombres de ciudades, etc., tienen el mismo comportamiento.

Problema: Se requiere un experto. A veces no es fácil definir las clases.

3) Dirigido por los datos.

El conjunto de datos se usa para obtener las clases automáticamente.

Métodos:

- Algoritmo Voraz basado en la Información Mutua (Brown,92).
- Algoritmo basado en técnicas estadísticas (Kneser,93).

Modelos basados en categorías

MÉTODOS DE AGRUPAMIENTO AUTOMÁTICO: (Brown,92)

Sea un modelo de bigramas de clases. Sea W el vocabulario, C el conjunto de clases y π una función que asigna cada palabra $w_i \in W$ a su clase $c_i \in C$.

Se propone maximizar:

$$L(\pi) = \sum_{w} P(w) \log P(w) + \sum_{c_1 c_2} P(c_1 c_2) \log \frac{P(c_2 \mid c_1)}{P(c_2)} =$$

$$= -H(w) + I(c_1, c_2)$$

Donde *H(w)* es la entropía de la distribución de unigramas y $I(c_1,c_2)$ es la información mutua media de las clases adyacentes. Como $L(\pi)$ (la verosimilitud de la partición π) depende de la partición sólo a través de la información mutua, la partición que maximiza $L(\pi)$ es también la que maximiza la información mutua entre clases.

Modelos basados en categorías

MÉTODOS DE AGRUPAMIENTO AUTOMÁTICO: (Brown,92)

ALGORITMO **Merge**:

1. Asignación de una clase distinta a cada palabra, cálculo de la información mutua entre clases adyacentes.
2. Mezclar el par de clases para el cual la pérdida de la información mutua media sea menor.
3. Como la información mutua puede ser mayor cambiando algunas palabras de clase en lugar de mezclar clases, después de algunas mezclas se analiza todo el vocabulario recolocando las palabras de forma que la partición resultante tenga la mayor información mutua media.
4. El proceso para cuando se ha alcanzado un número de clases predefinido.

RESULTADOS

Corpus de inglés escrito de 365.893.263 palabras.

Vocabulario = 260.741 Número de clases = 1000

El grado de captura de aspectos sintácticos y semánticos del inglés es sorprendente si se tiene en cuenta que es un procedimiento automático.

Modelos basados en categorías

MÉTODOS DE AGRUPAMIENTO AUTOMÁTICO: (Brown,92)

Friday Monday Thursday Wednesday Tuesday Saturday Sunday weekends Sundays Saturdays

June March July April January December October November September August

people guys folks fellows CEOs chaps doubters commues unfortunates blokes

down backwards ashore sideways southward northward overboard aloft downwards adrift

water gas coal liquid acid sand carbon steam shale iron

great big vast sudden mere sheer gigantic lifelong scant colossal

man woman boy girl lawyer doctor guy farmer teacher citizen

American Indian European Japanese German African Catholic Israeli Italian Arab

pressure temperature permeability density porosity stress velocity viscosity gravity tension

mother wife father son husband brother daughter sister boss uncle

machine device controller processor CPU printer spindle subsystem compiler plotter

John George James Bob Robert Paul William Jim David Mike

anyone someone anybody somebody

feet miles pounds degrees inches barrels tons acres meters bytes

director chief professor commissioner commander treasurer founder superintendent dean custodian

liberal conservative parliamentary royal progressive Tory provisional separatist federalist PQ

had hadn't hath would've could've should've must've might've

asking telling wondering instructing informing kidding reminding bothering thanking deposing

that tha theat

head body hands eyes voice arm seat eye hair mouth

Modelos basados en categorías

MÉTODOS DE AGRUPAMIENTO AUTOMÁTICO: (Kneser,93)

Sea un modelo de bigramas. El criterio de optimización es el de maximizar la probabilidad del corpus de aprendizaje $w_1 w_2 \ldots w_M$, es decir, minimizar:

$$-ln\left(\prod_{i=1}^{M} P(c_i \mid c_{i-1}) P(w_i \mid c_i)\right)$$

Considerando las frecuencias para el cálculo de las probabilidades:

$$LP = -\ln\left(\prod_{i=1}^{M} \frac{c(c_{i-1}c_i)}{c(c_{i-1})} \frac{c(w_i)}{c(c_i)}\right) = \sum_{i=1}^{M} \left(-\ln(c(c_{i-1}c_i)) + \ln(c(c_{i-1})) - \ln(c(w_i)) + \ln(c(c_i))\right)$$

Reorganizando la suma sobre el vocabulario, y eliminando los factores que no están afectados por el criterio de optimización:

$$LP = -\sum_{c_1 c_2} \left(c(c_1 c_2) ln(c(c_1 c_2))\right) + 2\sum_{c} \left(c(c) ln(c(c))\right)$$

El objetivo es encontrar el agrupamiento que minimice *LP*.

Modelos basados en categorías

MÉTODOS DE AGRUPAMIENTO AUTOMÁTICO: (Kneser,93)

ALGORITMO **ML**:

1. Escoger una función de etiquetado inicial.
2. Iterar hasta un cierto criterio de convergencia

   Para todas las palabras *w*

   Para todas las clases *c'*

   Analizar cómo cambia *LP* si *w* se mueve de *c* a *c'*.

   Mover las palabras *w* a la clase *c'* que devuelva la máxima optimización del criterio.

RESULTADOS

Alemán: corpus con 100.000 palabras (periódicos) vocabulario = 14.000

Inglés: corpus con 1.1 millones de palabras (LOB corpus) vocabulario = 50.000

Se dispone de un etiquetado POS (302 A, 153 I).

Método de suavizado: interpolación no lineal.

3/4 de los corpora para entrenamiento, 1/4 para prueba.

Número de clases óptimo:

Alemán = 120

Inglés = 350

Perplejidad del conjunto de prueba:

	Alemán	Inglés
ML	557	478
ML + bigramas	492	439
ML + bi-POS	408	420
Bigramas	650	541
Bi-POS	485	556

Modelos basados en categorías

MÉTODOS DE AGRUPAMIENTO AUTOMÁTICO: (Kneser,93)

Cluster A: *turned carried opened sent moved caught laid drew pulled lifted threw pushed handed pressed crossed burst thrust slipped swept stretched poured plunged wrapped weighed switched dragged waved ruled rolled rang . . .*

Cluster B: *people men children women boys girls persons students animals teachers officers soldiers stars Americans informants doctors employees Indians Africans prisoners individuals couples servants farmers doors conservatives critics folk artists visitors . . .*

Cluster C: *important serious interesting effective useful popular significant suitable dangerous familiar successful powerful appropriate positive expensive excellent attractive odd complex satisfactory exciting angry vital complicated valuable unexpected outstanding exact improved critical . . .*

Modelos basados en categorías
MÉTODOS DE AGRUPAMIENTO AUTOMÁTICO

COMPARACIÓN EXPERIMENTAL (Moisa, 95):

Se han aplicado los dos métodos anteriores para la obtención de un modelo de bigramas de clases para una tarea específica: consulta sobre horarios de trenes.

Se ha realizado un agrupamiento automático salvo para unas clases predefinidas: ciudades, estaciones, días de la semana, etc. Vocabulario = 751 palabras.

Entrenamiento: 9000 frases de habla espontánea, 60000 palabras

Test: 1358 frases.

Método: *ML*, n. clases óptimo=250,

perpl.=25.8, error reconocimiento (palabras)=17.7%.

Método: *Merge*, n. clases óptimo=300,

perpl.=29.7, error reconocimiento (palabras)=18.4%.

Los resultados son similares para los dos métodos.
En ambos métodos el uso del agrupamiento automático frente a etiquetado manual o ausencia de etiquetado ofrece mejores tasas de reconocimiento.

MODELOS ESTADÍSTICOS

- **Modelos de n-gramas**
- **Métodos de suavizado**
  - Plano y añade uno
  - Interpolación Lineal
  - Interpolación No lineal
  - Back-off
- **Modelos basados en categorías**
  - Definición y tipos de clases
  - Agrupamiento automático
- **Modelos dinámicos**
  - Cache
  - Triggers

Modelos dinámicos

Un modelo dinámico o adaptativo es un modelo que cambia su estimación como resultado de analizar texto del corpus de test. Este tipo de modelo es útil cuando:

- El texto de aprendizaje así como el de test es un gran texto heterogéneo que se compone de segmentos más pequeños homogéneos.
- El modelo de lenguaje ha sido entrenado con datos de un dominio y se pretende utilizar en otro dominio.

Modelos dinámicos

CACHE

La idea básica es que las palabras o secuencias de palabras una vez ocurren en un texto, tienen una mayor probabilidad de volver a ocurrir.

Trabaja como una memoria que usa frecuencias de palabras de un pasado reciente para estimar “probabilidades” a corto plazo, que servirán para actualizar las de los modelos estáticos.

Variantes:

- El componente cache como parte del modelo de bi-POS, de forma que se añade a la probabilidad de una palabra en su categoría.
- El componente cache como suavizador de las probabilidades del unigrama en un modelo de palabras, de forma que se añade a la probabilidad del unigrama.

Modelos dinámicos

CACHE

Sea M la longitud de la memoria cache, de forma que contiene las palabras $w_{n-1} \ldots w_{n-M}$. La probabilidad cache de la palabra w_n es:

$$P^c(w_n) = \sum_{m=1}^{M} a_m \delta(w_n, w_{n-m})$$

con $$\delta(w_n, w_{n-m}) = \begin{cases} 1 & si\ w_{n-m} = w_n \\ 0 & si\ w_{n-m} \neq w_n \end{cases}$$

El parámetro a_m es el peso de la posición *M-m+1* y se cumple que:

$$0 \leq a_m \leq 1 \qquad \sum_m a_m = 1$$

La determinación de estos pesos se puede hacer a partes iguales o bien de forma que las palabras más recientes tengan más peso.

Modelos dinámicos

CACHE

VARIANTE 1: el componente cache como parte de un modelo de bi-POS (Kuhn, 90):

La probabilidad de una palabra en su categoría se estima como:

$$P^c(w_n \mid c_{w_n}) = \beta P(w_n \mid c_{w_n}) + (1-\beta)\sum_{m=1}^{M} a_m^{c_{w_n}} \delta(w_n, w_{n-m})$$

VARIANTE 2: El componente cache como parte del modelo de unigramas (Essen, 91):

$$P^c(w_n) = \beta P(w_n) + (1-\beta)\sum_{m=1}^{M} a_m \delta(w_n, w_{n-m})$$

Los pesos de la interpolación β se estiman usando el conjunto de entrenamiento.

RESULTADOS: Mejoran en general la perplejidad del test set. Esta mejora es mayor en los documentos que presentan más heterogeneidad (Kuhn,90), (Jelinek,91), (Essen, 91).

Modelos dinámicos

TRIGGERS

La idea principal estriba en que además de la contribución del componente cache en la historia del documento de prueba, existe una información importante a considerar: la correlación entre palabras o secuencias de palabras.

Ejemplo: The district attorney's office launched a comprehensive investigation into loans made by several well connected banks.

Un humano puede utilizar la lectura de "district attorney" y "launched" para predecir "investigation", y la lectura de "loans" para anticipar "banks".

El modelo trigger trata de captar de forma sistemática esta información, usando la correlación entre secuencias de palabras derivada de un gran corpus de entrenamiento.

Modelos dinámicos

TRIGGERS

Definición: Si la secuencia de palabras *A* está altamente correlacionada con la secuencia *B*, entonces *(A->B)* se considera un trigger pair. *A* es el elemento desencadenador y *B* el desencadenado.

Determinación de los trigger pairs: Sea *h* la historia del documento ya vista. Sean *A* y *B* secuencias de palabras vistas en la historia. Sea B_0 el evento que representa que la secuencia *B* ocurre inmediatamente después en el documento. Una medida natural de la información proporcionada por *A* sobre B_0 es la información mutua media entre ambas:

$$I(A: B_0) = log\frac{P(B_0 \mid A)}{P(B_0)}$$

donde $P(B_0)$ es la probabilidad de B_0, $P(B_0|A)$ es la probabilidad condicional asignada al evento B_0 por el trigger pair *(A->B)*.

Se define la "utilidad esperada" del trigger pair *(A->B)* para llevar a cabo la selección de la lista de trigger pairs, de la siguiente forma:

$$U(A \rightarrow B) = I(A : B_0)P(B_0 \mid A)$$

Modelos dinámicos

TRIGGERS

Combinación de los componentes estático y dinámico:

- Uso del componente trigger como parte del modelo de pertenencia de una palabra a su clase (como en (Kuhn, 90)).
- Combinación del modelo estático y el dinámico usando un interpolación lineal.
- Uso de métodos de máxima entropía.

RESULTADOS (Lau,93): Con el corpus del WSJ con 24M palabras se estima un modelo estático de trigramas con back-off, y una lista de triggers.

Se realiza una interpolación lineal del trigrama estático con el modelo de trigger con pesos entre 0.02 y 0.06 para el componente trigger.

Se obtienen una mejoras del 10, 28 y 32 % en perplejidad del test set.