1.4 HERRAMIENTAS DE MODELIZACIÓN DEL LENGUAJE

Toolkits y formatos de los datos

En un modelo de lenguaje se suelen representar y calcular las probabilidades en formato log (logprob). Su uso permite manejar números no tan pequeños como en el caso de las probabilidades planas, y los productos se convierten en sumas. Si se necesita obtener probabilidades se aplica la función exponencial sobre el logprob.

$$p_1 \times p_2 \times p_3 \times p_4 = \exp(\log p_1 + \log p_2 + \log p_3 + \log p_4)$$

Los modelos de lenguaje de N-gramas con backoff se suelen almacenar en formato ARPA. Es un fichero ASCII con una pequeña cabecera seguida de una lista de probabilidades de N-gramas diferentes de cero (los trigramas, bigrama y unigramas, por ejemplo). Cada entrada se almacena con su logprob descontada (generalmente se utiliza el logaritmo base 10) y su peso de backoff α. Este peso sólo es necesario si este N-grama es prefijo de un n-grama mayor (por lo que no acompaña a los N-gramas de mayor orden) o N-gramas finales (acabados con la marca de final <s>)

Para un modelo de trigramas el formato es:

unigram:	$\log p^*(w_i)$	w_i	$\log \alpha(w_i)$
bigram:	$\log p^*(w_i \mid w_{i-1})$	$w_{i-1}w_i$	$\log \alpha(w_{i-1}w_i)$
trigram:	$\log p^*(w_i \mid w_{i-2}, w_{i-1})$	$w_{i-2}w_{i-1}w_i$	

Toolkits y formatos de los datos

Mostramos algunos N-gramas seleccionados del corpus BeRP en formato ARPA.

```
\data\
ngram 1=1447
ngram 2=9420
ngram 3=5201

\1-grams:
-0.8679678      </s>
-99             <s>                             -1.068532
-4.743076       chow-fun                        -0.1943932
-4.266155       fries                           -0.5432462
-3.175167       thursday                        -0.7510199
-1.776296       want                            -1.04292
...

\2-grams:
-0.6077676      <s>     i                       -0.6257131
-0.4861297      i       want                    0.0425899
-2.832415       to      drink                   -0.06423882
-0.5469525      to      eat                     -0.008193135
-0.09403705     today   </s>
...

\3-grams:
-2.579416       <s>     i       prefer
-1.148009       <s>     about   fifteen
-0.4120701      to      go      to
-0.3735807      me      a       list
-0.260361       at      jupiter </s>
-0.260361       a       malaysian       restaurant
...
\end\
```

Toolkits y formatos de los datos

Dado uno de esos trigramas, la probabilidad para la secuencia de palabras x,y,z se calcula:

$$P_{\text{katz}}(z|x,y) = \begin{cases} P^*(z|x,y), & \text{if } C(x,y,z) > 0 \\ \alpha(x,y) P_{\text{katz}}(z|y), & \text{else if } C(x,y) > 0 \\ P^*(z), & \text{otherwise.} \end{cases}$$

$$P_{\text{katz}}(z|y) = \begin{cases} P^*(z|y), & \text{if } C(y,z) > 0 \\ \alpha(y) P^*(z), & \text{otherwise.} \end{cases}$$

Hay herramientas disponibles para la construcción de modelos de lenguaje con diferentes métodos de suavizado:

- SRILMtoolkit (Stolke, 2002)
  http://www.speech.sri.com/projects/srilm/download.html
- KENLM (K. Heafield 2011)
  http://kheafield.com/code/kenlm/

Unknown words: vocabulario abierto versus vocabulario cerrado

Si se conocen todas las palabras

- El vocabulario *V* es fijo
- Tarea de vocabulario cerrado

A menudo no se conoce exactamente todo el vocabulario

- **Out Of Vocabulary** = OOV words
- Tarea de vocabulario abierto

Para ello: se crea un token para la palabra desconocida <UNK>

- Se entrenan las probabilidades de <UNK>
  - Se crea un léxico *L*
  - En una fase posterior, cualquier palabra del conjunto de entrenamiento que no esté en *L* se sustituye por <UNK>
  - Se entrenan las probabilidades tomando <UNK> como una palabra más
- En la fase de decodificación
  - Se usan las probabilidades estimadas para <UNK> para toda palabra que no esté en *L*