Algoritmo de Smoothing Kneser-Ney para Modelos de Lenguaje de NGramas

Giovanni Rescia
gir0109@famaf.unc.edu.ar

FaMAFyC

February 20, 2016

Introducción

- Problema: los modelos de n-gramas son modelos probabilísticos; si son implementados sin técnicas de smoothing, responden mal a eventos no observados en tiempo de entrenamiento.
- Solución: las técnicas de smoothing tienen como objetivo atacar ese problema. Las técnicas implementadas hasta el momento son:
  - Addone (Laplace, o Additive con parámetro δ=1)
  - Interpolated (Jelinek-Mercer)
  - Backoff con Discounting (Katz)
- Motivación: estudios demostraron que la técnica de smoothing de Kneser-Ney da los mejores resultados generales.

Idea Intuitiva detrás del algoritmo

Dan Jurafsky provee el siguiente ejemplo en sus videolecturas:
Supongamos que tenemos un corpus y que a nivel de unigrama, 'Francisco' ocurre con alta frecuencia.
Pero si nos movemos a nivel de bigramas, vemos que 'Francisco' solo sigue a 'San', entonces la mayoría de bigramas donde ocurre 'Francisco' es en 'San Francisco'.
Solución: Introducir la idea de contexto.

Definiciones

Sean:

1) $N_{1+}(\bullet w_i) = |\{w_{i-1} : c(w_{i-1}^i) > 0\}|$

2) $N_{1+}(\bullet\bullet) = |\{(w_i, w_{i-1}) : c(w_{i-1}^i) > 0\}| = \sum_{w_i} N_{1+}(\bullet w_i)$

3) $N_{1+}(w_{i-n+1}^{i-1}\bullet) = |\{w_i : c(w_{i-n+1}^i) > 0\}|$

4) $N_{1+}(\bullet w_{i-n+1}^i) = |\{w_{i-n} : c(w_{i-n}^i) > 0\}|$

5) $N_{1+}(\bullet w_{i-n+1}^{i-1}\bullet) = |\{(w_{i-n}, w_i) : c(w_{i-n}^i) > 0\}|$

Notar que 5) también puede escribirse como:

$$N_{1+}(\bullet w_{i-n+1}^{i-1}\bullet) = \sum_{w_i} N_{1+}(\bullet w_{i-n+1}^i)$$

Definiciones (cont.)

1) Es la cantidad de palabras distintas que preceden a w_i observadas en entrenamiento

2) Es la cantidad de bigramas distintos observados en entrenamiento

3) Es la cantidad de palabras distintas que tiene como predecesor al n-grama w_{i-n+1}^{i-1}

4) Es la cantidad de palabras distintas que preceden al n-grama w_{i-n+1}^{i}

5) Es la cantidad de palabras distintas w_{i-n}, w_i tales que el n-grama w_{i-n}^{i} se ha visto en entrenamiento

Algoritmo

Casos:

- El orden del modelo es n $= 1$:

$$P_{KN}(w_i) = \frac{counts(w_i)}{counts()}$$

- El orden del modelo es n > 1:
  - Para el nivel más alto de n-gramas, usar:

$$P_{KN}(w_i|w_{i-n+1}^{i-1}) = \frac{max\{counts(w_{i-n+1}^{i}) - D, 0\}}{counts(w_{i-n+1}^{i-1})}$$

$$+\frac{D}{counts(w_{i-n+1}^{i-1})} * N_{1+}(w_{i-n+1}^{i-1}\bullet) * P_{KN}(w_i|w_{i-n+2}^{i-1})$$

Algoritmo (cont.)

- Para 1 < k < n k-gramas, usar:

$$P_{KN}(w^i|w_{i-n+1}^{i-1}) = \frac{max\{N_{1+}(\bullet w_{i-n+1}^{i}) - D, 0\}}{N_{1+}(\bullet w_{i-n+1}^{i-1}\bullet)}$$

$$+\frac{D}{N_{1+}(\bullet w_{i-n+1}^{i-1}\bullet)} * N_{1+}(w_{i-n+1}^{i-1}\bullet) * P_{KN}(w_i|w_{i-n+2}^{i-1})$$

- Para el nivel más bajo (unigramas), usar:

$$P_{KN}(w_i) = \frac{N_{1+}(\bullet w_i)}{N_{1+}(\bullet\bullet)}$$

Heurísticas

Se utilizó suavizado por AddOne, haciendo los siguientes remplazos:

$$P_{KN}(w_i) = \frac{counts(w_i)}{counts()} \Longrightarrow P_{KN}(w_i) = \frac{counts(w_i) + 1}{counts() + |V|}$$

$$N_{1+}(\bullet w_{i-n+1}^{i-1} \bullet) \Longrightarrow max\{N_{1+}(\bullet w_{i-n+1}^{i-1} \bullet), 1\}$$

$$N_{1+}(w_{i-n+1}^{i} \bullet) \Longrightarrow max\{N_{1+}(w_{i-n+1}^{i} \bullet), 1\}$$

Resultados

- Shakespeare Corpus

	Perplexity			
Smoothing	n=1	n=2	n=3	N=4
NGram (None)	Infinite			
AddOne	833.01	1975.82	13570.08	24318.21
Interpolated	834.65	352.03	331.69	328.33
BackOff	834.65	273.22	254.69	261.08
KneserNey	833.01	261.58	201.12	239.14

Resultados (cont.)

- Brown Corpus

	Perplexity			
Smoothing	n=1	n=2	n=3	N=4
NGram (None)	Infinite			
AddOne	1512.77	5500.39	36192.00	59218.40
Interpolated	1570.48	680.69	660.24	660.91
BackOff	1570.48	490.55	481.41	494.52
KneserNey	1512.77	453.84	414.33	507.97

Resultados (cont.)

- Gutenberg Corpus

	Perplexity			
Smoothing	n=1	n=2	n=3	N=4
NGram (None)	Infinite			
AddOne	1944.95	6391.66	37478.36	55587.10
Interpolated	2155.14	1821.52	1899.86	1914.00
BackOff	2155.14	1541.76	1659.18	1700.70
KneserNey	1944.95	1170.80	917.71	1034.76