Modelos de lenguaje

Jaime Ferrando Huertas

Octubre 2020

1 Introducción

Esta trabajo consiste en la evaluación de las prestaciones de los diferentes modelos de lenguaje de n-gramas y los diferentes métodos de suavizado para su uso como modelos de lenguaje. En esta experimentación se estudiará cómo afectan diversos parámetros a las prestaciones del modelo: el método de suavizado, los métodos de descuento, etc. Para los tres primeros ejercicios hemos trabajado con el corpus **Dihana** y en el cuarto hemos usado el corpus de mayor tamaño **Europarl**.

2 Resultados

2.1 Tarea 1

Para esta tarea hemos usado el corpus Dihana y el suavizado por defecto de la herramientata junto al descuento Good-Turing y esquema de suavizado backoff. Adjuntamos una tabla con la perplejidad de los modelos para N= 1,2,3,4,5.

Table 1: Tarea 1

Corpus	**N**	**Descuento**	**Suavizado**	**Perplejidad ppl1**
Dihana	1	Good Turing	Backoff	183.84
Dihana	2	Good Turing	Backoff	15.71
Dihana	3	Good Turing	Backoff	10.21
Dihana	4	Good Turing	Backoff	9.52
Dihana	5	Good Turing	Backoff	9.59

Podemos ver como la perplejidad disminuye a medida que aumentamos N hasta el valor 4, con N=5 la ventana de n-gramas es demasiado grande y no vemos beneficios en la perplejidad. Usaremos los valores de óptimos N=3, N=4 para las siguientes tareas.

2.2 Tarea 2

Utilizando los valores de N=3,4 óptimos para nuestro corpus obtenidos en la tarea 1 decidimos evaluar distintos métodos de descuento: Good-Turing, Witten-Bell, modified Kneser-Ney y unmodified Kneser-Ney (todos con el método de suavizado por defecto). Adjuntamos una tabla con la perplejidad de cada combinación.

Podemos observar que los resultados no varían tanto como en la anterior tarea, los cuatro métodos de descuento funcionan de manera similar con el unmodified Kneser-Ney reportando mejores resultados para nuestro problema.

Table 2: Tarea 2

Corpus	**N**	**Descuento**	**Suavizado**	**Perplejidad ppl1**
Dihana	3	Good Turing	Backoff	10.21
Dihana	4	Good Turing	Backoff	9.52
Dihana	3	Witten Bell	Backoff	10.44
Dihana	4	Witten Bell	Backoff	9.41
Dihana	3	Modified kneser ney	Backoff	11.19
Dihana	4	Modified kneser ney	Backoff	10.63
Dihana	3	Unmodified kneser ney	Backoff	10.17
Dihana	4	Unmodified kneser ney	Backoff	9.24

2.3 Tarea 3

Para esta tarea queremos estudiar dos métodos de suavizado distintos, interpolación y backoff. Hemos usado los valores óptimos de N=3,4 junto a los descuentos Witten-Bell y Modified Kneser-Ney. Adjuntamos una tabla con la perplejidad de cada combinación.

Table 3: Tarea 3

Corpus	**N**	**Descuento**	**Suavizado**	**Perplejidad ppl1**
Dihana	3	Witten Bell	Backoff	10.44
Dihana	4	Witten Bell	Backoff	9.41
Dihana	3	Witten Bell	Interpolacion	9.73
Dihana	4	Witten Bell	Interpolacion	10.25
Dihana	3	Modified kneser ney	Backoff	11.19
Dihana	4	Modified kneser ney	Backoff	10.63
Dihana	3	Modified kneser ney	Interpolacion	8.56
Dihana	4	Modified kneser ney	Interpolacion	9.21

Podemos ver como el método de suavizado de interpolación nos reporta mejores resultados para tres de los cuatro casos. Vemos un una mejora superior frente a Backoff cuando es usado junto con el descuento de Modified Kneser-Ney que cuando usamos el descuento de Witten-Bell.

2.4 Tarea 4

Para esta tarea hemos usado el corpus de mayor tamaño **Europarl**. El objetivo de esta tarea ha sido evaluar añadir un fichero de vocabulario a nuestro modelo de lenguaje en lugar de tomar como vocabulario el que se estima a partir del conjunto de entrenamiento. Se han probado distintos vocabularios removiendo palabras dependiendo de su frecuencia.

Hemos usado valores para N=3,4 junto a los valores por defecto para el

descuento y suavizado (Good Turing y Backoff). Adjuntamos una tabla con la perplejidad de cada combinación.

Table 4: Tarea 4

Corpus	**N**	**Descuento**	**Suavizado**	**Frecuencia**	**Perplejidad ppl1**
Europarl	3	Good Turing	Backoff	>1	99.42
Europarl	4	Good Turing	Backoff	>1	89.92
Europarl	3	Good Turing	Backoff	>5	96.24
Europarl	4	Good Turing	Backoff	>5	87.02
Europarl	3	Good Turing	Backoff	>9	94.31
Europarl	4	Good Turing	Backoff	>9	85.26

Podemos observar como la perplejidad de nuestro modelo se reduce a medida que eliminamos palabras con menor frecuencia, sin embargo esto no tiene porque ser un punto positivo ya que nuestro modelo de lenguaje perderá la capacidad de reconocer ciertas palabras del vocabulario. Creemos que lo mejor seria hacer un estudio mas exhaustivo de estos modelos para evaluar si la eliminación de estas palabras es realmente beneficioso.

3 Conclusiones

Gracias a este trabajo hemos podido poner en practica bastantes conceptos aprendidos en teoría durante la asignatura, desconocíamos la existencia del software de SRLIM y ha sido un ejercicio beneficioso para afianzar la teoría.

Respecto a los resultados, podríamos decir que para el corpus dihana el mejor modelo lo hemos visto en la tarea 3 con valores N=3 Descuento=Modified kneser ney Suavizado=Interpolación. Creemos que el uso del suavizado por interpolación es el parámetro que reporta mejores beneficios viendo el salto en perplejidad entre la tarea 2 y 3. Por otro lado en el corpus Eurocarl hemos visto como se reducía la perplejidad ha medida que reducíamos el numero de palabras en el vocabulario basándonos en su frecuencia. Como hemos comentado esta es una conclusión que no nos convence demasiado e instamos a realizar mas experimentos con estos modelos para asegurarnos de la eficacia de los mismos.