Evaluación de Etiquetadores Morfosintácticos para el Español

David Barbas Rebollo

Lingüística Computacional
Noviembre, 2020

1 Introducción

El etiquetado *POS tagging* consiste en asignar una categoría gramatical a cada palabra de un texto. Este trabajo final presenta la evaluación de las prestaciones de distintos etiquetadores morfosintácticos para afrontar este problema.

En el paquete *NLTK* (*Natural Language Toolkit*) están disponibles los distintos etiquetadores y el corpus de español utilizados, *cess-esp*. Se experimenta con diversos parámetros para evaluar las prestaciones del sistema. Entre estos parámetros se encuentran: el tamaño del *training set*, el método de suavizado de palabras desconocidas y el conjunto de categorías morfosintácticas utilizadas. Posteriormente, se analiza la herramienta *Freeling*.

Por último, cabe comentar que se ha empleado el corpus reducido, barajado y utilizando validación cruzada en el entrenamiento en todas las tareas exceptuando las que indiquen lo contrario.

2 Tarea 1

En esta tarea se compara la evaluación de la validación cruzada del corpus *cess-esp* con el conjunto completo de las etiquetas con respecto del conjunto reducido. Para ello, se utiliza el etiquetador basado en *Hidden Markov Models* (*HMM*) del paquete *NLTK*.

Figure 1: Gráfica de los resuldatos al aplicar *HMM* sobre el corpus con el conjunto completo de etiquetas.

Table 1: Tabla de los resuldatos al aplicar *HMM* sobre el corpus con el conjunto completo de etiquetas.

Fold	Accuracy (%)	CI 95% (%)
0	89.24	[86.78, 91.70]
1	89.50	[87.07, 91.94]
2	89.72	[87.30, 92.13]
3	89.62	[87.20, 92.04]
4	89.34	[86.89, 91.79]
5	90.00	[87.62, 92.38]
6	89.87	[87.47, 92.27]
7	89.43	[86.99, 91.87]
8	90.08	[87.70, 92.45]
9	89.94	[87.55, 92.33]

Figure 2: Gráfica de los resuldatos al aplicar *HMM* sobre el corpus con el conjunto reducido de etiquetas.

Table 2: Tabla de los resuldatos al aplicar *HMM* sobre el corpus con el conjunto reducido de etiquetas.

Fold	Accuracy (%)	CI 95% (%)
0	92.29	[90.18, 94.41]
1	92.83	[90.79, 94.88]
2	92.64	[90.57, 94.72]
3	92.32	[90.21, 94.44]
4	92.36	[90.25, 94.47]
5	92.64	[90.56, 94.71]
6	92.72	[90.65, 94.78]
7	92.48	[90.39, 94.57]
8	92.81	[90.76, 94.86]
9	92.73	[90.66, 94.79]

Se puede observar como se mejora la media del *accuracy* de 89.67% a 92.58% al utilizar el conjunto reducido. Esto se debe a que al utilizar un menor número de clases el modelo es capaz de aprender mejor de los datos haciendo una mejor generalización.

3 Tarea 2

El objetivo de la segunda tarea se analiza el *accuracy* del etiquetador basado en *HMM* variando el tamaño del conjunto de entrenamiento. Para ello, se ha dividido todo el corpus reducido en 10 particiones. Se utilizan incrementalmente de la primera a la novena, mientras que se reserva la decima para la evaluación del modelo. Para el entrenamiento no se utiliza validación cruzada.

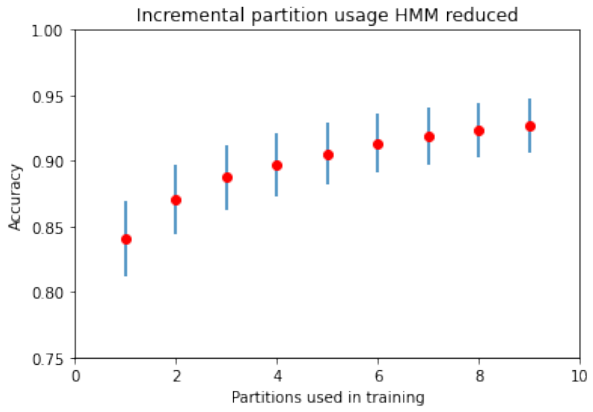

Figure 3: Gráfica de los resuldatos al aplicar *HMM* sobre el corpus con el conjunto reducido de etiquetas e incrementando progresivamente el conjunto de entrenamiento.

Table 3: Tabla de los resuldatos al aplicar *HMM* sobre el corpus con el conjunto reducido de etiquetas e incrementando progresivamente el conjunto de entrenamiento.

Partitions used	Accuracy (%)	CI 95% (%)
0	92.29	[90.18, 94.41]
1	92.83	[90.79, 94.88]
2	92.64	[90.57, 94.72]
3	92.32	[90.21, 94.44]
4	92.36	[90.25, 94.47]
5	92.64	[90.56, 94.71]
6	92.72	[90.65, 94.78]
7	92.48	[90.39, 94.57]
8	92.81	[90.76, 94.86]

Se puede observar una clara progresión de mejora, en la Figura 3 y la Tabla 3 al incrementar el conjunto de entrenamiento.

4 Tarea 3

En la tercera tarea se utiliza el etiquetador *Trigrams'n'Tags* (*TNT*). Normalente el conjunto de entramiento no contiene un vocabulario completo del conjunto de datos. Por ello, se debe poner una solución a este problema. Para solventarlo, se usa un método de suavizado, un etiquetador *AffixTagger* ya que *TNT* no incorpora un método de suavizado para este problema. *AffixTagger* realiza una estimación en base al sufijo de la palabra desconocida. En esta tarea se experimenta si utilizar este suavizado ayuda a mejorar el *accuracy*. Además se experimenta con distintas longitudes de sufijo para determinar la óptima. Cuando la longitud del sufijo es cero, no se usa este suavizado.

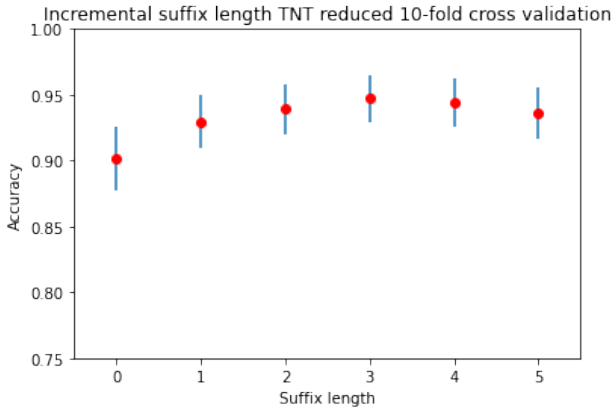

Figure 4: Gráfica de los resuldatos al aplicar *TNT* sobre el corpus con el conjunto reducido de etiquetas y variando la longitud del sufijo.

Table 4: Tabla de los resuldatos al aplicar *TNT* sobre el corpus con el conjunto reducido de etiquetas y variando la longitud del sufijo.

Suffix Length	Accuracy (%)	CI 95% (%)
0	90.16	[87.80, 92.53]
1	92.96	[90.92, 94.99]
2	93.91	[92.01, 95.81]
3	94.70	[92.93, 96.48]
4	94.42	[92.60, 96.24]
5	93.57	[91.62, 95.52]

Al finalizar esta tarea, se observa como introducir un método de suavizado para las palabras desconocidas mejora casi 5 puntos porcentuales el *accuracy* del etiquetador. Además, se ve como aumentando la longitud del sufijo se van mejorando las prestaciones del etiquetador. Sin embargo, al sobrepasar la longitud 3 empeoran las prestaciones del etiquetador. Se concluye que la configuración con la longitud 3 para los sufijos es la óptima.

5 Tarea 4

En la cuarta tarea, se proponen 3 nuevos paradigmas de etiquetado disponibles en el paquete *NLTK*. Este trabajo evalua el etiquetador *Brill*, *CRF* y *Perceptron*. Al terminar la experimentación con estos etiquetadores, compararemos sus prestaciones con los etiquetadores *HMM*, *TNT* y *TNT* con suavizado de sufijos con longitud 3.

Cabe comentar que la técnica *Brill* requiere una inicialización previa. Se ha elegido hacerlo mediante los etiquetadores *HMM* y *Unigram*.

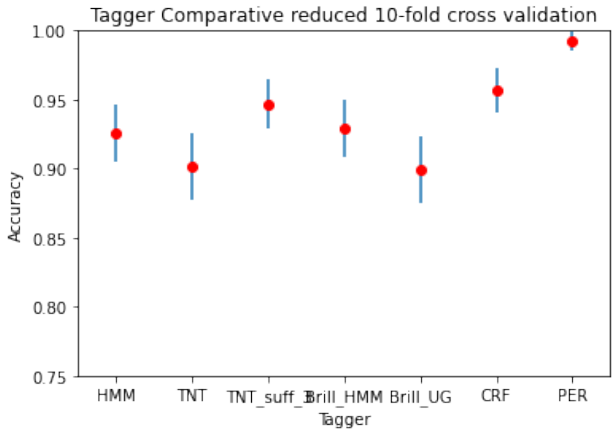

Figure 5: Gráfica comparativa del *accuracy* obtenido por distintos etiquetadores sobre el corpus con el conjunto reducido de etiquetas.

Table 5: Tabla comparativa del *accuracy* obtenido por distintos etiquetadores sobre el corpus con el conjunto reducido de etiquetas.

Tagger	Accuracy (%)	CI 95% (%)
HMM	92.58	[90.50, 94.66]
TNT	90.16	[87.79, 92.53]
TNT_suff_3	94.70	[92.92, 96.48]
Brill_HMM	92.93	[90.90, 94.97]
Brill_UG	89.96	[87.57, 92.34]
CRF	95.67	[94.05, 97.28]
PER	99.28	[98.60, 99.95]

Después de evaluar las distintas técnicas, concluimos que sólo los etiquetadores *CRF* y *perceptron* son capaces de obtener un *accuracy* superior a los resultados obtenidos anteriormente. En concreto el *perceptron* obtiene las mejores prestaciones con un *acurracy* de 99.28%.

6 Tarea 5

En la última y quinta tarea, se realiza una evaluación de la herramienta *Freeling* y ha sido evaluada en *Linux*. *Freeling* es una biblioteca C++ que

proporciona funcionalidades de análisis de idiomas (análisis morfológico, detección de entidades con nombre, etiquetadoPoS, análisis, desambiguación de sentidos de una palabra, etiquetado semántico, etc.) para una variedad de idiomas.

Cabe destacar que la instalación ha sido bastante fácil ya que hemos podido realizarla a través de un archivo *.deb*. Además, su utilización también ha sido bastante sencilla. Aún así la documentación es nefasta, esta muy mal organizada. Asimismo, puede dar el caso que sus servidores esten caidos. Ha sido muy costoso encontrar los sencillos pasos para realizar esta tarea. también es posible utilizar la herramienta en *C++*, *Java* o *Python* pero debido a que configurar las dependencias era demasiado tedioso, se ha optado por trabajar en la terminal.

Para la realización del etiquetado morfosintáctico del archivo proporcionado $Alicia\_utf8.txt$ se ha ejecutado el siguiente comando en la consola.

./analyze -f es.cfg < Alicia_utf8.txt > output_Alicia_utf8.txt

7 Preguntas planteadas

7.1 ¿Por qué al reducir el número de etiquetas del corpus se obtienen mejores resultados?

Cuando se reduce el número de etiquetas, el etiquetador es capaz de entrenar con más muestras de cada categoría, lo que mejora su aprendizaje de las categorías. Además, en test etiquetador tendrá menos posibilidad de fallar ya que tiene menos opciones de etiquetado.

7.2 ¿Qué efecto tiene aumentar el conjunto de entrenamiento?

Cuando se aumenta el número de muestras de entrenamiento, el modelo a entrenar es capaz de realizar un mejor aprendizaje. Esto se debe a que tiene más ejemplos de donde aprender las características de cada clase. Por ello, si no sobregeneraliza las categorías, aumentará su tasa de acierto.