Solemne I – 3era Parte: Minería de Datos y Procesamiento de Lenguaje Natural

Nombre: Rodrigo Fuenzalida
Profesor: Alejandro Figueroa

1.- Introducción

Para esta tarea se utilizó el grupo de features 1, ya que fue el que mejor resultados entrego en la actividad anterior.

2.- Oversampling vs. Undersampling

- **Undersampling**: Para generar los datos con oversampling se tomó una porción de los datos con clase tokens y se dejó la cantidad de entidades fija. Esto se hizó para poder equiparar los datos de entidades con los datos pertenecientes a tokens. Se utilizó el 20% de los tokens para poder generar los modelos para este método.
- **Oversampling**: En este caso se copiaron las entidades para aumentar la cantidad de estos datos, con el fin de equiparar la carga entre tokens y entidades.

¿Qué puede concluir de ambos experimentos?

Tomando en cuenta algunas de las categorías de preguntas como ejemplo, (en archivo adjunto (undersampling, oversampling) se pueden encontrar todas las categorías), podemos notar que ambos modelos mejoran la precisión, si bien ambos modelos no son perfectos en el sentido de detectar entidades, cuentan con mejores detecciones y más errores.

```
########################################################################
Categoria Environment:
########################################################################

              precision    recall  f1-score   support

          -1       0.99      1.00      0.99      7839
           1       0.69      0.27      0.39       121

avg / total       0.98      0.99      0.98      7960


Confusion matrix
+-------------+---------------+--------------+
|             |   Entity real |   Token real |
+=============+===============+==============+
| Entity pred |            33 |           88 |
+-------------+---------------+--------------+
| Token pred  |            15 |         7824 |
+-------------+---------------+--------------+
```

Figura 1: Resultados feature 1 tarea anterior.

```
#########################################################################
Categoria Environment:
#########################################################################

              precision    recall  f1-score   support

          -1       0.99      1.00      0.99      7839
           1       0.57      0.31      0.40       121

avg / total       0.98      0.99      0.98      7960


Confusion matrix
+-------------+--------------+--------------+
|             |   Entity real |   Token real |
+=============+==============+==============+
| Entity pred |           37 |           84 |
+-------------+--------------+--------------+
| Token pred  |           28 |         7811 |
+-------------+--------------+--------------+
```

Figura 2: Resultados feature 1 con oversampling.

```
#########################################################################
Categoria Environment:
#########################################################################

              precision    recall  f1-score   support

          -1       1.00      0.98      0.99      7839
           1       0.37      0.79      0.50       121

avg / total       0.99      0.98      0.98      7960


Confusion matrix
+-------------+--------------+--------------+
|             |   Entity real |   Token real |
+=============+==============+==============+
| Entity pred |           95 |           26 |
+-------------+--------------+--------------+
| Token pred  |          162 |         7677 |
+-------------+--------------+--------------+
```

Figura 3: Resultados feature 1 con undersampling.

Para este caso en particular, Figura 2 y 3, se puede apreciar respectivamente los resultados para oversampling y undersampling, siendo este último el que mejor clasifica entidades, si bien, la tasa de error con respecto a la confusión de tokens es alta, clasifica más entidades que los modelos anteriores, esto es debido a que el modelo tiene la misma cantidad de información sobre entidades que el primer modelo sin undersampling ni oversampling, pero posee menor información sobre los tokens. Para el caso de oversampling, si bien no predice tan bien y se parece bastante a los resultados de la Figura 1, nuevamente presenta mayor error al confundir entidades por tokens, como se puede apreciar en las matrices de confusión.

En conclusión, undersampling nos ayuda a tener mayor precisión al momento de querer clasificar entidades, sin embargo, no entre mayor error que los demás modelos, puesto que la

información de cada clase puede llegar a ser tan similar (tendríamos que analizar caso por caso, para llegar a tomar una decisión final), esto altera los resultados. Obviamente, viendo las matrices de confusión, podemos apreciar que la clase “tokens” sigue siendo la clase dominante del conjunto de datos, puesto que estos datos son muchos más que la clase “entity”. Para poder tener una respuesta más segura sobre la clase “entity” habría que ver como se comporta el clasificador con una mejor cantidad de tokens, menor al 20% de las muestras que representan esta clase, para sí lograr apreciar un cambio más determinante dentro de la clasificación.

- Curva ROC para los tres casos.

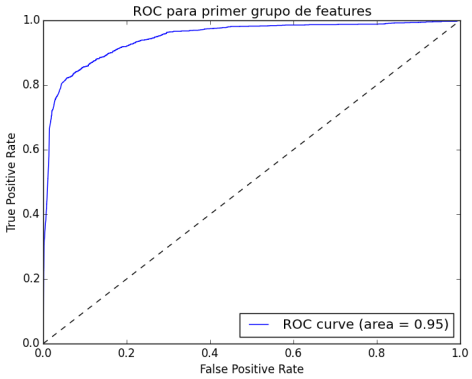


Figura 4: Curva ROC para feature 1 tarea anterior.

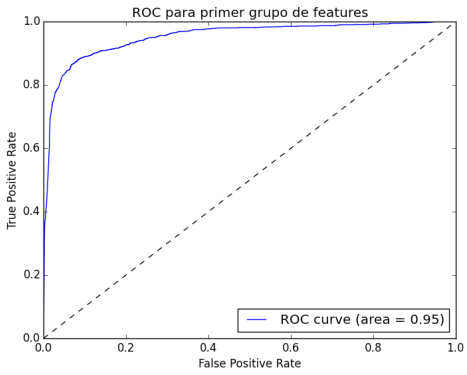


Figura 5: Curva ROC para feature 1 Oversampling.

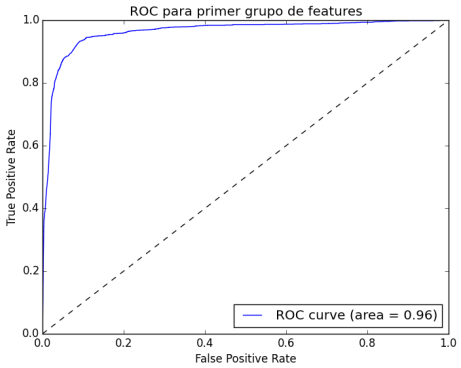
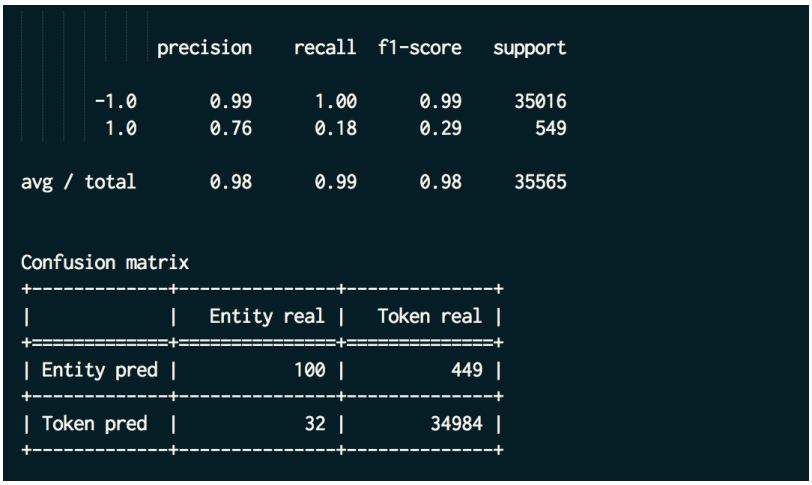


Figura 6: Curva ROC feature 1 Undersampling

3.- Bagging

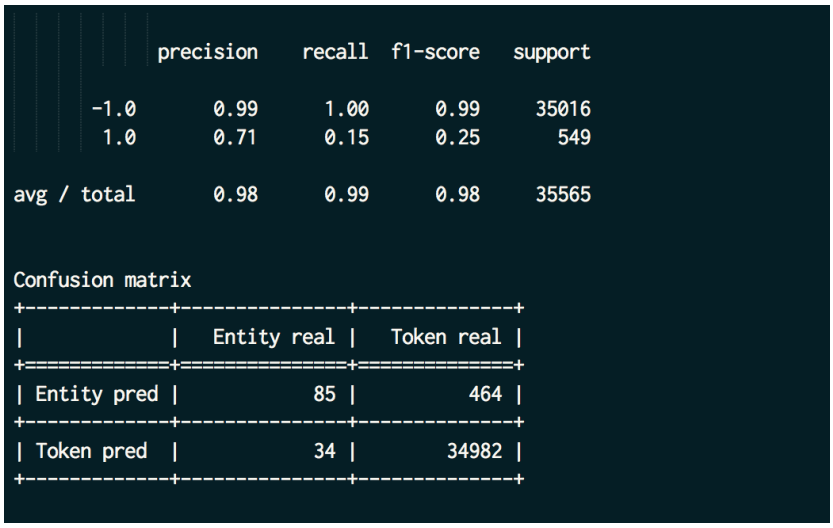
- **Resultados colectivos**:
  - Votos primer clasificador:



```
              precision    recall  f1-score   support

        -1.0       0.99      1.00      0.99     35016
         1.0       0.76      0.18      0.29       549

avg / total       0.98      0.99      0.98     35565


Confusion matrix
+-------------+---------------+--------------+
|             |   Entity real |   Token real |
+=============+===============+==============+
| Entity pred |           100 |          449 |
+-------------+---------------+--------------+
| Token pred  |            32 |        34984 |
+-------------+---------------+--------------+
```

Figura 7: Votos primer clasificador

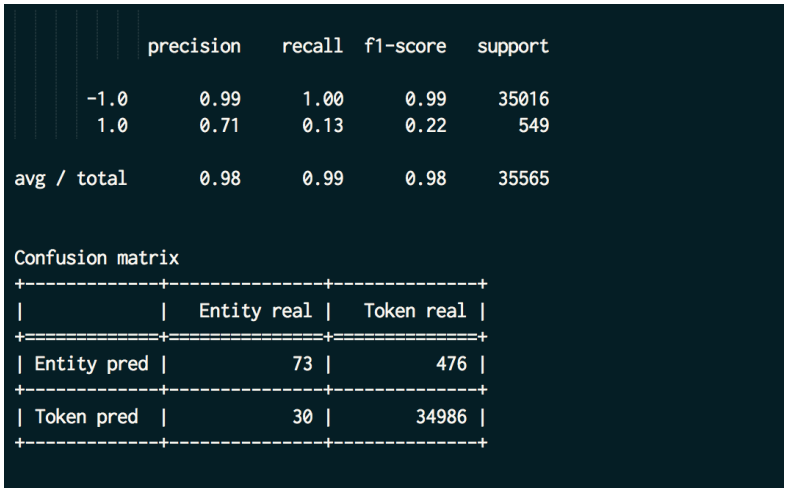
  - Votos primer a quinto clasificador:



```
              precision    recall  f1-score   support

        -1.0       0.99      1.00      0.99     35016
         1.0       0.71      0.15      0.25       549

avg / total       0.98      0.99      0.98     35565


Confusion matrix
+-------------+---------------+--------------+
|             |   Entity real |   Token real |
+=============+===============+==============+
| Entity pred |            85 |          464 |
+-------------+---------------+--------------+
| Token pred  |            34 |        34982 |
+-------------+---------------+--------------+
```

Figura 8: Votos desde primer clasificador a quinto clasificador.

- Votos primer a décimo clasificador:



	precision	recall	f1-score	support
-1.0	0.99	1.00	0.99	35016
1.0	0.71	0.13	0.22	549
avg / total	0.98	0.99	0.98	35565

Confusion matrix

	Entity real	Token real
Entity pred	73	476
Token pred	30	34986

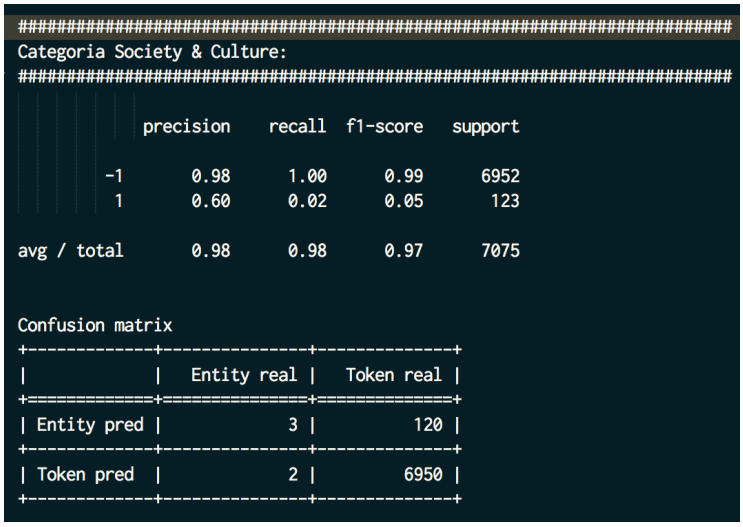
Figura 9: Votos desde quinto clasificador a décimo clasificador.

Se puede ver en las Figuras 7, 8 y 9, el avance de la toma de decisiones por parte de los clasificadores, siendo la Figura 9, el resultado final de la votación. Éste método nos permite tener una clasificación un poco mejor, ya que dividimos el problema en pequeños grupos los cuales tienen distintas opiniones sobre lo que están analizando, lo que nos permite tener una visión más general sobre el problema, ya que podemos atacar la información desde distintos puntos de vistas, y así estos puntos, los juntamos para obtener una clasificación con menos errores. Ahora bien, esto también puede tener muchos errores puesto que si alguno de los clasificadores se entrena con más tokens que el resto, esto puede incurrir en tener un sesgo muy grande de las muestras.

4.- Modelos de aprendizaje

A continuación se presenta un extracto de los resultado, para ver las matrices de confusión de todos las categorías (ver Archivos adjuntos: maxent, bayes).

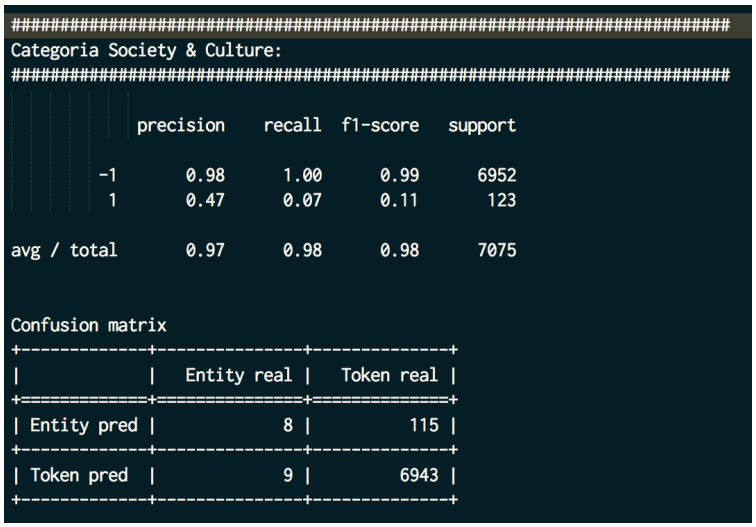
- **Maxent**:



```
########################################################################
Categoria Society & Culture:
########################################################################

              precision    recall  f1-score   support

          -1       0.98      1.00      0.99      6952
           1       0.60      0.02      0.05       123

avg / total       0.98      0.98      0.97      7075


Confusion matrix
+-------------+--------------+--------------+
|             |   Entity real |   Token real |
+=============+==============+==============+
| Entity pred |            3 |          120 |
+-------------+--------------+--------------+
| Token pred  |            2 |         6950 |
+-------------+--------------+--------------+
```

Figura 10: Resultados feature 1 Maxent.

- **Bayes**:



```
########################################################################
Categoria Society & Culture:
########################################################################

              precision    recall  f1-score   support

          -1       0.98      1.00      0.99      6952
           1       0.47      0.07      0.11       123

avg / total       0.97      0.98      0.98      7075


Confusion matrix
+-------------+--------------+--------------+
|             |   Entity real |   Token real |
+=============+==============+==============+
| Entity pred |            8 |          115 |
+-------------+--------------+--------------+
| Token pred  |            9 |         6943 |
+-------------+--------------+--------------+
```

Figura 11: Resultados feature 1 Bayes.

Como se puede ver en las Figuras 7 y 8, se presentan los resultados obtenidos para la clasificación de tokens y entidades para la categoría “Society y Culture”, En este caso podemos ver cómo clasifica los tokens de buena forma, esto lo podemos ver en la precisión que éste tiene como valor, además la precisión para la clase entity es medianamente alta para el caso de maxent, en caso contrario bayes clasifica un poco peor a los entities, ¿A qué se debe este comportamiento?, Básicamente esto se explica principalmente por la forma de los clasificadores. En el caso de SVM

tenemos un clasificador lineal, el cual en tiempo de ejecución busca encontrar el hiper-plano que divide las muestras y el margen o sesgo que éstas poseen. Por otra parte Maxent y bayes ven la probabilidad de aparición de una palabra, en este caso es una palabra, por lo que separan de mejor forma las muestras, dada la estructura de los features.

Como vimos en clases, maxent busca los pesos que mejor interpretan las muestras utilizadas para entrenar, por lo que se ciñe a este comportamiento, lo cual hace que sea un clasificador mucho más asertivo para este caso.

Para el caso de Bayes sucede algo similar, ya que este trabaja con la probabilidad a priori, lo cual nos permite tener un probabilidad aislada por cada palabra, lo cual permite tener menos errores.

- Curvas ROC:

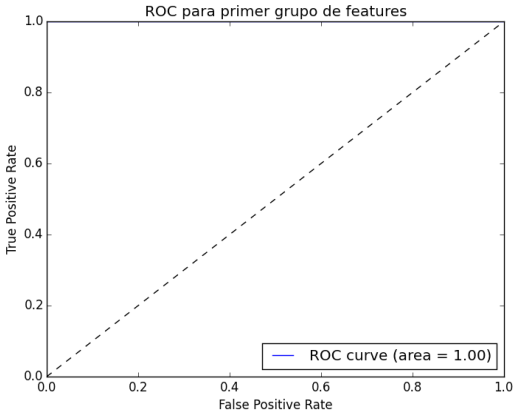


Figura 12: Resultado Maxent.

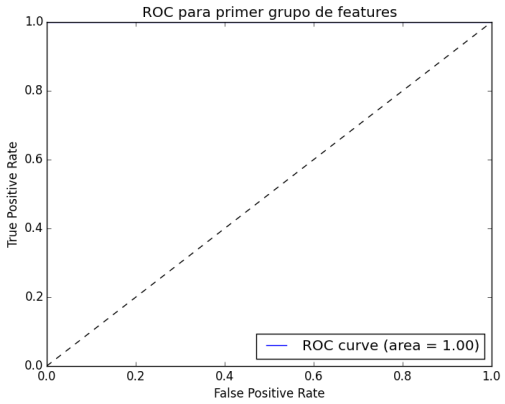


Figura 13: Resultado Bayes.

5.- Modelos de Clasificación Binaria vs. Multi-Clase

```
########################################################################
Categoria Travel:
########################################################################

              precision    recall  f1-score   support

         1.0       0.99      1.00      0.99      4543
         2.0       0.00      0.00      0.00        25
         3.0       0.00      0.00      0.00        40
         4.0       0.00      0.00      0.00         4

avg / total       0.97      0.99      0.98      4612


Confusion matrix
+------------------+-------------+--------------------+----------------+---------------+
|                  |   Token real |   Organization real |   Location real |   Person real |
+==================+=============+====================+================+===============+
| Token pred       |         4543 |                  0 |              0 |             0 |
+------------------+-------------+--------------------+----------------+---------------+
| Organization pred |           25 |                  0 |              0 |             0 |
+------------------+-------------+--------------------+----------------+---------------+
| Location pred    |           40 |                  0 |              0 |             0 |
+------------------+-------------+--------------------+----------------+---------------+
| Person pred      |            4 |                  0 |              0 |             0 |
+------------------+-------------+--------------------+----------------+---------------+
```

Figura 14: Resultados SVM multi-clase.

Para este caso y como se muestra en la Figura 11, se tomó como ejemplo la clase “Travel”, además se observa que los tokens en su mayoría son bien clasificados y que en esta categoría al igual que en el resto de categorías, SVM Multi-clase, tiende a clasificar todo como tokens, por lo que genera bastantes errores de acuerdo a la cantidad de entidades que halla en los textos.

Posiblemente una mejora (no probada), sería utilizando undersampling de los tokens, para así observar el comportamiento del clasificador con respecto a las entidades.