Tarea 5 - Minería de Datos y Procesamiento de Lenguaje Natural

Profesor: Alejandro Figueroa
Ayudante: Nicolás Olivares
Fecha de Entrega: Viernes 12 de octubre de 2013 23:59hrs
Ponderación: 1
Vía de Entrega: Correo electrónico al ayudante (nicolivares@gmail.com)

Enunciado Parte 1

El objetivo de esta tarea es aprender el concepto de "cross-validation". La idea detrás de aprendizaje supervisado es aprender patrones desde ejemplos vistos (etiquetados) del pasado, que nos permitan predecir las clases de ejemplos nuevos, es decir del presente o del futuro.

En el contexto del curso, dado que tenemos una colección de 3,000 consultas etiquetadas manualmente, la idea es ahora ver si podemos aprender un modelo capaz de predecir las etiquetas de las millones de consultas que entran a un buscador.

El primer fundamento que hay que tener en claro es: "*No se puede testear un modelo con los mismos datos con los cuales éste fue creado*". Es decir, si yo entreno o genero un modelo con las 3,000 consultas, para ver cual es su desempeño necesito tener una colección adicional de consultas etiquetadas.

Cross-validation soluciona este problema utilizando los mismos datos para entrenar y testear mediante el siguiente concepto: "*Si oculto un porcentaje de los datos para testear y con los restantes entreno, entonces puedo obtener pronósticos válidos para esos datos de prueba. Si hago ésto sistemáticamente con diferentes porciones de los datos, de manera de probar con todos ellos, podría utilizarlos todos para evaluar el desempeño de un modelo*". Para entender esta idea más claramente, antes debemos introducir un parámetro "n" utilizado por cross-validation. Este parámetro es llamado el número de "folds", o el número de paquetes disjuntos de igual tamaño que se harán con los datos. Normalmente, "n" se configura en 3, 5 ó 10, dependiendo la cantidad de datos. Entre más datos se dispongan, un "n" más pequeño se utiliza. Para esta tarea, utilizaremos n=10.

En decir, las 3,000 consultas se dividen en 10-folds disjuntos de datos. Cada uno de ellos va a tener 300 consultas. Llamemos a cada "fold" F_i con i=1,..,10. Entonces, se generan 10 experimentos utilizándolos de la siguiente forma:

Experimento	Prueba	Entrenamiento
1	F_1	$E_1 = F_2 + F_3 + F_4 + F_5 + F_6 + F_7 + F_8 + F_9 + F_{10}$
2	F_2	$E_2 = F_1 + F_3 + F_4 + F_5 + F_6 + F_7 + F_8 + F_9 + F_{10}$
3	F_3	$E_3 = F_1 + F_2 + F_4 + F_5 + F_6 + F_7 + F_8 + F_9 + F_{10}$
4	F_4	$E_4 = F_1 + F_2 + F_3 + F_5 + F_6 + F_7 + F_8 + F_9 + F_{10}$
5	F_5	$E_5 = F_1 + F_2 + F_3 + F_4 + F_6 + F_7 + F_8 + F_9 + F_{10}$
6	F_6	$E_6 = F_1 + F_2 + F_3 + F_4 + F_5 + F_7 + F_8 + F_9 + F_{10}$
7	F_7	$E_7 = F_1 + F_2 + F_3 + F_4 + F_5 + F_6 + F_8 + F_9 + F_{10}$
8	F_8	$E_8 = F_1 + F_2 + F_3 + F_4 + F_5 + F_6 + F_7 + F_9 + F_{10}$
9	F_9	$E_9 = F_1 + F_2 + F_3 + F_4 + F_5 + F_6 + F_7 + F_8 + F_{10}$
10	F_{10}	$E_{10} = F_1 + F_2 + F_3 + F_4 + F_5 + F_6 + F_7 + F_8 + F_9$

Como se puede apreciar, en cada experimento se deja afuera del entrenamiento el fold que se utilizará para probar, y todo el resto se utiliza para entrenar. De esta manera, podemos utilizar todos los datos para entrenar y testear de manera sistemática. Nótese que cada archivo debe consistir en la representación vectorial de cada consulta. Por su conveniencia, se le recomienda utilizar el siguiente formato:

```
<etiqueta> <feature>:<valor> <feature>:<valor> ... <feature>:<valor> # <comentario>
```

Un ejemplo

```
3 1:1 3:2 9284:1 # abcdef
```

Nótese que las etiquetas "INF", "NAV" y "RES" deben ser mapeadas a un número 1, 2 y 3. Y las id de los features deben comenzar con 1. Las id de los features es el número de la componente de la representación vectorial que ya han usado en las tareas anteriores. Nótese que éstas parten normalmente desde cero, de ser así, deberá sumarle uno. El valor es la frecuencia, en otras palabras, es el valor de la componente respectiva en la representación vectorial, es decir "3:2" denota que la tercera componente tiene una frecuencia de dos. Pero si sus vectores parten de "0", esa componente va a ser la segunda. En resumen, el objetivo es escribir los vectores que ya tienen en un archivo, con un formato establecido.

Cuando se tienen los 10 folds en los archivos en el formato especificado, se realizan los 10 experimentos, es decir se aprenden 10 modelos, y por ende se realizan las predicciones para cada uno de los folds de prueba. Para ésto, utilizaremos Support Vector Machines, en especial, SVM_MULTICLASS, que se encuentra en:

http://www.cs.cornell.edu/People/tj/svm_light/old/svm_multiclass_v2.12.html

Para cada uno de los 10 folders de entrenamiento (cada uno con 2,700 datos) se debe ejecutar:

```
svm_multiclass_learn -c 1.0 E_i modelo_i
```

Y después aplicar el modelo aprendido al folder respectivo de prueba:

```
svm_multiclass_classify F_i modelo_i resultados_i
```

Al final de este proceso se van a tener 10 archivos de resultados con las predicciones de cada folder. Comparando las etiquetas manuales y las asignadas por SVM, calcule accuracy y Mean Reciprocal Rank (MRR). Muestre la matriz de confusión y analice los errores. Ejemplifique sus conclusiones.

Algunas referencias que pueden servir de apoyo:

1) http://en.wikipedia.org/wiki/Cross-validation_(statistics)
2) http://stackoverflow.com/questions/7619700/10-fold-cross-validation