Universidad del País Vasco Euskal Herriko Unibertsitatea

Sistemas de Ayuda a la Decisión

Práctica 2

Índice

1. Objetivos y Descripción de los contenidos

Esta práctica, junto con la práctica 1, tiene como objetivo adquirir las competencias que serán evaluadas en la primera prueba práctica individual que tendrá lugar hacia la cuarta semana. Por lo tanto esta tarea se enmarca dentro de la evaluación continua pero no dentro de los hitos evaluables.

Las **competencias** que el alumno deberá haber adquirido tras realizar la práctica son:

1. Capacidad para normalizar atrbutos de tipo texto y de generar nuevos atributos a partir de estos.
2. Capacidad para generar un primer prototipo básico de un modelo de predicción con la configuración por defecto.
3. Capacidad para evaluar la bonanza del modelo generado empleando:
   - el esquema de evaluacón apropiado (valiadación cruzada k-fold, hold-out)
   - la figuras de mérito apropiadas: Accuracy, precision, recall, f-score...

2. Materiales disponibles

El siguiente material será empleado para el desarrollo de las tareas que se proponen y que tiene como objetivo alcanzar las competencias enumeradas anteriormente.

1. Datos: Se podrán encontrar en eGela los datos de SMS etiquetados como SPAM o no SPAM [1] y datos de e-mails que igualmente tendrán que ser etiquetados como SPAM o no SPAM [2].

2. Tutoriales: Se pone a disposición del alumno en el servidor http://lsi.bp.ehu.es/asignaturas/SAD-2021-2022/practica2/ los siguientes tutoriales:

   - 1-preprocesoDeAtributosLengNatural.mp4 (obj: Preprocesado de atributos textuales).
   - 2-construccionYEvaluacionModelo.mp4 (obj: Creación y evaluación de un modelo de predicción básico).

   Lecturas: https://knowledge.dataiku.com/latest/courses/nlp-visual/text-handling/text-handling-summary.html.

3. Tareas

Para obtener los objetivos buscados en este proyecto se proponen las siguientes **tareas**:

- Visualizar los tutoriales.
- Responder a las preguntas sobre los mismos.
- Repetir los pasos presentados en los tutorial de una manera guiada, de forma autónoma sobre un nuevo conjunto de datos.

3.1. Tarea previa guiada 1: Visualización del vídeo sobre preprocesado de datos textuales.

Visualizar los vídeos : Su referencia se encontrará en eGela.
Leer el contenido de la siguiente página del manual de DSS:
https://knowledge.dataiku.com/latest/courses/nlp-visual/text-handling/text-handling-summary.html
Duración total de los vídeos cortos aprox. **60 min**
Duración del primer vídeo aprox. **30 min** El primer vídeo muestra como realizar un preproceso mínimo de unos mensaje SMS y trabaja los siguientes conceptos:

- Crear un receta para el tratamiento de atributos de tipo texto.

  Concepto: Preprocesado de atributos de tipo texto (lematización, conversión a minúsculas, eliminación de stop words...).

  Tutorial práctico: Preprocesar un atributo de tipo texto.

  Prueba: Preprocesar un atributo.

1. Normalizar	Resp-A. Quedarse con la raiz de las palabras. Por ejemplo, crecer crec
2. Lematizar	Resp-B. Convertir el texto a minúsculas, eliminar las tíldes, y convertir el texto a Unicode
3. Eliminar stop words	Resp-C. Eliminar del texto las palabras de semántica ligera, es decir, palabras con poco contenido semántico que no representan conceptos concretos. Ejemplos pueden ser las preposiciones, artículos, etc

Cuestionario:
Duración aprox. **5 min.**

1. Pregunta: Empareja los términos de la izquierda con sus descripciones a derecha (respuestas encima).
2. Pregunta: Cúal(es) de las siguientes afirmaciones es/son cierta(s)
   - Hashing y Bag Of Words son iguales porque no son más que dos formas de llamar a una vectorización
   - BOW: los tokens (palabras o lemas de palabras) individuales se extraen y se obtiene su frecuencia en el texto a convertir. Por otro lado, cada palabra dentro del conjunto de datos obtiene un identificador único construyendose así un diccionario o vocabulario. Entonces, cualquier texto (por ejemplo un mensaje de SMS) que veamos puede codificarse como un vector de longitud fija (la longitud del vocabulario obtenido a partir del conjunto de datos) donde en cada posición en el vector se almacenaría la frecuencia de cada palabra en el texto (mensaje SMS). Se puede emplear en CountVectorizer.
   - Hashing: Emplear un hash unidireccional de palabras para convertirlos tokens (palabras o lemas) en números enteros a través de algún algoritmo codificador (por ejemplo Murmurhash3). Esto no requiere de vocabulario y puede elegir un vector de longitud fijo de longitud arbitraria. Una desventaja es que el hash es una función unidireccional, por lo que no hay forma de volver a convertir la codificación en una palabra.
3. Pregunta: ¿En qué consiste la simplificación que nos propone DSS?
   - En normalizar el texto pasarlo a minúsculas, eliminar los signos de puntuación, lematizarlo, y eliminar las stop words.
   - En convertirlo en más simple.

4. Pregunta: Responde si la siguiente afirmación es verdadera o falsa: la representación por BOW presenta el problema de generar vectores muy dispersos donde hay muchos ceros.
    - Falso: No hay ceros en en BOWs.
    - Verdadero: El vector tendrá tantos elementos como palabras contenga el vocabulario representativo de todo el conjunto de datos, así que para un determinado texto (especialmente en los SMS) su representación BOW contendrá muchos ceros porque habrá muchas palabras que no aparezcan en el texto (SMS).

Duración del segundo vídeo aprox. **30 min** El segundo vídeo muestra como crear con la ayuda de dataiku un modelo rápido de clasificación y visualizar distintas métricas para medir la bonanza de dicho modelo. En este caso la tarea de predicció consistirá en clasificar mensajes SMS como SPAM o HAM:

- Crear un modelo de predicción que nos propone dataiku por defecto.

    Concepto: tipo de predicción; clasificación binaria, clasificación multiclase, regresión. Conjuntos de entrenamiento y testado. Elección del algoritmo. Métricas de evaluación.

    Tutorial práctico: Generar un modelo de predicción básico.

    Prueba: Generar un modelo modificando el muestreo para la construcción de los conjuntos de entrenamiento y testeo.

Cuestionario:
Duración aprox. **5 min.**

1. Pregunta: Cúal(es) de las siguientes afirmaciones es/son cierta(s)
    - Hashing y Bag Of Words son iguales porque no son más que dos formas de llamar a una vectorización
    - BOW: los tokens (palabras o lemas de palabras) individuales se extraen y se obtiene su frecuencia en el texto a convertir. Por otro lado, cada palabra dentro del conjunto de datos obtiene un identificador único construyendose así un diccionario o vocabulario. Entonces, cualquier texto (por ejemplo un mensaje de SMS) que veamos puede codificarse como un vector de longitud fija (la longitud del vocabulario obtenido a partir del conjunto de datos) donde en cada posición en el vector se almacenaría la frecuencia de cada palabra en el texto (mensaje SMS). Se puede emplear en CountVectorizer.
    - Hashing: Emplear un hash unidireccional de palabras para convertirlos tokens (palabras o lemas) en números enteros a través de algún algoritmo codificador (por ejemplo Murmurhash3). Esto no requiere de vocabulario y puede elegir un vector de longitud fijo de longitud arbitraria. Una desventaja es que el hash es una función unidireccional, por lo que no hay forma de volver a convertir la codificación en una palabra.

2. Pregunta: Responde si la siguiente afirmación es verdadera o falsa. Cuando la distribución de los datos con respecto a la clase que se quiere predecir está muy desbalanceada, la accuracy es una buena métrica a emplear para medir la bonanza del predictor.
   - Verdadero: La accuracy mide el ratio de aciertos con respecto al número total de instancias así que sí es una buena métrica en cualquier caso.
   - Falso. Cuando los datos están muy desbalanceados, asignando a todas la instancias la clase mayoritaria la accuracy será alta pero nuestro sistema realmente no ha aprendido nada.
3. Pregunta: Responde si la siguiente afirmación es verdadera o falsa. Cuando un predictor en una clasificación binarie asigna a todas las instancias una clase determinada, por ejemplo en nuestro caso SPAM.
   - Verdadero: El recall será del 100 % y la precision coincidirá con la accuracy.
   - Falso. El recall coincidirá con la accuracy y la precision será del 100 %.
4. Pregunta: Empareja los términos de la izquierda con sus descripciones a derecha (Respuestas en la siguiente página).

1. Recall	Resp-A. Capacidad de un modelo para encontrar **todos** los casos relevantes dentro de un conjunto de datos. En nuestro ejemplo, la capacidad del sistema de identificar **todos** los SPAMS que hay. Se calcula como el número de verdaderos positivos dividido por el número de verdaderos positivos más el número de falsos negativos.
2. Precision	Resp-B. Media entre la precision y el recall, para medir el equilibrio, es decir encontrar todos y solo los que son
3. FScore	Resp-C. Capacidad de un modelo de clasificación para identificar **solo** los puntos de datos relevantes. En nuestro ejemplo, la capacidad del sistema de identificar **solo** los SPAMS que hay y no más (falsos positivos). Se calcula como número de verdaderos positivos dividido por el número de verdaderos positivos más el número de falsos positivos.

3.2. Tarea autónoma 1: Nueva tarea de clasificación sobre mensajes de eMail.

Con esta tarea autónoma el alumno deberá demostrarse a sí mismo que ha adquirido las competencias esperadas. Para ello se le solicita que cree un nuevo

modelo de predicción. Para ello importará los datos almacenados en el fichero spam-ham-emails.csv, y explorará los datos contenidos en el, y que comprobará la distribución de atributos SPAM-HAM de la muestra. Cabe recordar que la muestra por defecto se genera a partir de los primeros 10000 instancias, así que es interesante emplear la ventana de análisis, para mostrar la información necesaria. Si fuera necesario, el el alumno generará una nueva muestra asegurandose de que esta contenga suficientes representantes de ambas clases. Así y cuando la muestra sea adecuada, podrá generar un modelo de predicción de SPAMs.

Para llevar a cabo esta **subtarea** el alumno deberá bajar del servidor el fichero spam-ham-emails.csv que encontrará en eGela. Este fichero pertenece al conjunto de datos Enrom spam dataset y contiene información acerca de emails que han sido etiquetados como SPAM o no (HAM).
También se solicitará al alumno que exporte las recetas para generar modelos a un notebook de Python. Esta es una tarea *totalmente autónoma* para la cúal el alumno tendrá que investigar libremente.
Duración: Aprox. **30 min**.

1. Pregunta: Rellena la siguiente tabla con los valores que hayas seleccionado.

1. Target
Train/Test
Features
Algorithm
Evaluation

2. Pregunta: Rellena la siguiente tabla con los valores de la matriz de confusión. 3. Visiona el siguiente vídeo https://www.youtube.com/watch?v=TmhzUdPpVPQ.¿Para calcular el area ROC, necesitas conocer la probabilidad que asocia el modelo a cada predicción?¿Por qué?

	SPAM PREDICTION	HAM PREDICTION	TOTALS
REAL SPAM			
REAL HAM			
TOTALS			

Referencias

[1] Dataset de los SPAM: Almeida, T.A., Gómez Hidalgo, J.M., Yamakami, A. Contributions to the Study of SMS Spam Filtering: New Collection and Results. Proceedings of the 2011 ACM Symposium on Document Engineering (DOCENG'11), Mountain View, CA, USA, 2011.

[2] V. Metsis, I. Androutsopoulos and G. Paliouras, "Spam Filtering with Naive Bayes - Which Naive Bayes?". Proceedings of the 3rd Conference on Email and Anti-Spam (CEAS 2006), Mountain View, CA, USA, 2006.

[3] Competencias asociadas de la tarea: Se han tomado como referencia los apuntes de SAD 2020-2021 (fuente: Alicia Pérez) para así coordinar que la práctica y las competencias a obtener sea lo más similar posible a lo solicitado en años anteriores.