Challenge “Distill Bill”

Advanced Analytics. Smart Decisions

decide

CONFIDENCIALIDAD

TABLA DE CONTENIDOS

1. INTRODUCCIÓN

El reto consiste en ser capaces de extraer algunos campos determinados de información de facturas eléctricas. Estas facturas se presentan en distintos formatos, todas ellas en archivos PDF, pero con un orden y disposición diferente de los campos. Sin embargo, la información de las facturas, aun con variaciones, es esencialmente la misma, y algunos de estos campos comunes deben ser obtenidos de ellas.

El objetivo es ser capaces de crear un método que extraiga la información de las facturas de la forma más genérica posible, no personalizando para ningún tipo de plantilla de factura. De hecho, cuando se evalúe vuestro código, habrá tipos diferentes de plantillas de facturas que no estaban presentes en el dataset de entrenamiento, por lo que hacer algo genérico es fundamental. De cada una de las facturas habrá que obtener los mismos campos, de forma que se convierta esta información desestructurada presente en la factura en una información estructurada, con el mismo formato para todas ellas.

Se pueden desarrollar los métodos que se consideren oportunos para conseguir este objetivo, con los modelos que creáis más convenientes. La única limitación será utilizar Python.

2. DESARROLLO DEL RETO

El reto se desarrollará en varias fases:

1. Recibiréis el dataset de entrenamiento consistente en un número de facturas en formato PDF y sus correspondientes archivos JSON conteniendo la información que se quiere extraer de las facturas. Aquí podréis empezar a programar la extracción de campos de las facturas y comprobar su funcionamiento con los archivos etiquetados correspondientes.
2. Cuando termine el plazo de desarrollo, el día 14 de febrero como máximo, nos enviaréis el código que habéis utilizado para extraer los campos de las facturas, así como las versiones de las librerías usadas, por ejemplo con un archivo “requirements.txt”. No se puede tocar el código bajo ningún concepto a partir de este momento, y nos guardaremos una copia del mismo para comprobar los resultados es en caso de que resultéis entre los primeros clasificados. El código se enviará a jorgeraul.gomez@decidesoluciones.es. Es recomendable cambiar las extensiones de los archivos, de .py o .ipynb a .txt para que no haya problemas con el envío. También se pueden usar herramientas como https://codeshare.io o similares.
3. Recibiréis un dataset de test, con otras facturas diferentes, también en formato PDF, con las que tendréis que probar qué tal funciona vuestro código para extraer la información. Algunas de estas facturas tendrán el mismo formato que algunas de las facturas usadas en el dataset de entrenamiento, mientras que otras tendrán un formato completamente diferente. Vuestro código debería generar ficheros JSON como los que os enviamos en el dataset de entrenamiento para este dataset de test. Obviamente, no se pueden retocar manualmente los archivos JSON generados.

4. Nos enviaréis estos archivos JSON que habéis obtenido, y nosotros usaremos un script para obtener el score que habéis obtenido. Este score se obtendrá mediante una media de una métrica basada en la distancia de Levenshtein de todos los campos de todos los documentos, y se expresará en porcentaje. Concretamente, la métrica es:

$$Score = \frac{\sum_{i=1}^{n}\left(1 - \frac{L(\hat{s}, s)}{len(s)}\right)}{n}$$

siendo *L(a,b)* la distancia de Levenshtein entre los strings a y b, s_i el string del campo i-ésimo, $\hat{s}_i$ el string de vuestra predicción para el campo i-ésimo y len() la función que devuelve la longitud de un string. En el caso de que un campo no se encuentre, bien sea porque no existe el archivo, porque no existe un campo en el archvo JSON, o porque el nombre de la etiqueta o el archivo no son los esperados, se considerará que el string de la predicción es el string vacío "", y como su distancia de Levenshtein es igual a la longitud del string, su contribución al score será de 0.

5. Comprobaremos el código de los ganadores, para comprobar que efectivamente se generan los archivos JSON que nos habéis enviado y os enviaremos los scores. Por este motivo se pide por favor que el código sea legible y esté comentado. El 19 de febrero se comunicará a los participantes que hayan sacado mejor puntuación
6. El 26 de febrero los participantes que hayan sacado una mejor puntuación harán una presentación, explicando cómo lo han hecho.

3. DATASET

El dataset de entrenamiento está formado por 1000 facturas, que van numeradas desde "factura_0.pdf" hasta "factura_999.pdf". Cada una de ellas tiene un archivo JSON correspondiente con el mismo nombre, cambiando sólo la extensión, con los 19 campos que hay que extraer de las facturas. Estos campos son (entre paréntesis viene la etiqueta usada en el archivo JSON):

- Nombre del cliente (nombre_cliente)
- DNI del cliente (dni_cliente)
- Calle del clilente (calle_cliente)
- Código postal del cliente (cp_cliente)
- Población del cliente (población_cliente)
- Provincia del cliente (provincia_cliente)
- Nombre de la empresa comercializadora (nombre_comercializadora)
- CIF de la comercializadora (cif_comercializadora)
- Dirección de la comercializadora (dirección_comercializadora)
- Código postal de la comercializadora (cp_comercializadora)
- Población de la comercializadora (población_comercializadora)
- Provincia de la comercializadora (provincia_comercializadora)
- Número de factura (nümero_factura)
- Inicio del periodo de facturación (inicio_periodo)
- Fin del periodo de facturacón (fin_periodo)
- Importe de la factura (importe_factura)
- Fecha del cargo (fecha_cargo)
- Consumo en el periodo (consumo_periodo)
- Potencia contratada (potencia_contratada)

Los formatos, en caso de fecha, serán “DD.MM.YYYY”, por ejemplo “07.02.2016” y los valores numéricos con decimales estarán separados por una coma, por ejemplo “191,32”. En cualquier caso, si hay alguna duda, se pueden consultar los archivos JSON del dataset de entrenamiento para comprobar cuál es el formato utilizado.

Se puede descargar el dataset de entrenamiento en el siguiente enlace: https://decidesoluciones365-my.sharepoint.com/:u:/g/personal/jorgeraul_gomez_decidesoluciones_es/EeUSyMCJJPJOhtgV1PIcL7YB2hLDKKAvHSKWg_rruB61Yw?e=JY8ndE

El dataset de test está formado por otras 1000 facturas, también numeradas desde “factura_0.pdf” hasta “factura_999.pdf”. Los archivos JSON que generéis deberían seguir la misma nomenclatura, también usada den el dataset de entrenamiento, de forma que los archivos tengan el mismo nombre pero diferente extensión.

4. OTRAS CONSIDERACIONES

- Los archivos PDF contienen texto dentro de ellos. Existen librerías en Python que permiten extraer el texto directamente del archivo PDF. Pero hay que tener cuidado, no todas las librerías extraen el mismo texto ni en el mismo formato.
- No se considerará la diferencia entre letras mayúculas y minúsculas, pero sí se considerarán caracteres diferentes si llevan tilde o si no. Usad la codificación apropiada para tenerlo en cuenta.
- Es posible que en algunas facturas haya datos que faltan, mostrados como “XX”. No obstante, no debería ocurrir en ninguno de los casos de los datos que tenéis que obtener. El resto de información de las facturas es irrelevante.