Aldo Hernández
Abraham López
Damián García
Cristian Antonio

ENTREGABLE 1 - PIA

Abstract *En este documento seleccionamos nuestro conjunto de datos (imágenes) para realizar nuestro proyecto integrador de aprendizaje. Este proyecto tendrá la finalidad de crear una red neuronal artificial que clasifique las letras de palabras escritas a mano para su digitalización. El conjunto contiene un total de 413,701 registros divididos en tres subconjuntos: entrenamiento, prueba y validación, estos datos provienen de niños que escribieron a mano su nombre y apellido (si aplica) en algún papel. Finalmente, se detalla el cronograma con las distintas etapas del trabajo mediante un diagrama de Gantt según las fechas de entrega de cada documento semanal.*

Keywords python, dataset, exploración, análisis, pandas

1. Selección del dataset

Para seleccionar nuestro conjunto de datos investigamos diferentes fuentes de información confiables. Comenzamos buscando información gubernamental de México en relación a la salud, concretamente acerca de dengue y COVID-19; sin embargo, descartamos estos conjuntos de datos debido a la poca (o casi nula) aplicación real debido a la cantidad de datos y columnas ya que tenían poco que ver con la predicción de un posible diagnóstico para dichas enfermedades.

Posteriormente, encontramos en la plataforma Kaggle una serie de *datasets* interesantes, pero finalmente nos decantamos por un conjunto de datos que contiene más de 400,000 transcripciones de nombres escritos por niños.

Este *dataset* es adecuado para una tarea de aprendizaje supervisado ya que contiene un gran número de imágenes etiquetadas para llevar a cabo una tarea de reconocimiento óptico de caracteres (Optical Character Recognition, OCR), además, estas imágenes tienen buena calidad y no tienen ruido visual. Por otra parte, podrían haber algunos detalles ya que al tratarse de nombres, podría haber un desbalanceo de clases para ciertas letras no tan comunes en nombres de personas, como podrían ser las letras x, y, z, q o w. También, al ser un conjunto de datos hecho en idioma inglés, no existe ningún registro con la letra ñ. Además, todos los nombres están en mayúsculas y a pesar de que algunos tienen tildes, estas no serán tomadas en cuenta.

Fuera de esos detalles, consideramos que no hay ninguna otra complicación en relación al conjunto de datos, ya que solo se cuenta con dos columnas: la ruta hacia la imagen y el nombre asociado (en mayúsculas), y como a cada imagen le corresponde un nombre pues no hay problemas de datos faltantes.

Finalmente, es importante mencionar que este *dataset* ya está dividido en tres conjuntos importantes para el desarrollo de la red neuronal: entrenamiento, prueba y validación, así que en etapas posteriores ya no nos tendremos que preocupar en volverlas a separar.

2. Definición del problema

Al momento de escribir a mano notas o apuntes puede resultar complicado para el lector interpretar algunas letras o palabras, incluso si es "letra propia". Esto se debe a la infinidad de estilos y variantes que el ser humano puede presentar al escribir a mano, haciendo la tarea de reconocimiento de caracteres complicada tanto para una computadora como en ciertos casos a un ser humano. Es por esto que nos decidimos por este conjunto de datos, ya que queremos crear un modelo capaz de combatir dicha problemática.

Es decir, buscamos reconocer palabras en imágenes escaneadas con texto escrito a mano para digitalizarlo mediante la clasificación de los caracteres según las diferentes maneras de escribir las letras del abecedario inglés.

3. Código inicial

Para el *script* inicial usaremos la librería Pandas para la carga y exploración inicial de los datos.

exploration.py

```python
import pandas as pd

# Carga de los archivos en formato csv
df_train = pd.read_csv('../CSV/written_name_train.csv')
df_test = pd.read_csv('../CSV/written_name_test.csv')
df_validation = pd.read_csv('../CSV/written_name_validation.csv')

# Exploración preliminar: dimensiones
print(f'Cantidad de registros para entrenamiento: {df_train.shape[0]}',
      f'Cantidad de registros para pruebas: {df_test.shape[0]}',
      f'Cantidad de registros para validación: {df_validation.shape[0]}',
      f'Total de registros: {df_train.shape[0] + df_test.shape[0] + df_validation.shape[0]}',
      f'Cantidad de columnas: {df_train.shape[1]}',
      sep='\n')

# Exploración preliminar: primeros registros
print('Estos son los primeros 10 registros del conjunto de entrenamiento:',
      df_train.head(n=10),
      sep='\n')
```

Figure 1. Script inicial completo.

Como se puede observar, primero realizamos la carga de los tres *datasets* que conforman el *dataset* original: entrenamiento, prueba y validación. Seguido de ello investigamos las dimensiones de cada conjunto, como todos tendrán 2 columnas puesto que parten del mismo conjunto, solo nos centraremos en la cantidad de registros:

- Entrenamiento: 330,961 registros
- Prueba: 41,370 registros
- Validación: 41,370 registros
- Total: 413,701 registros

Además, podemos ver los primeros 10 registros del conjunto de entrenamiento en la siguiente tabla:

Table 1
Primeros registros de entrenamiento

#	FILENAME	IDENTITY
0	TRAIN_00001.jpg	BALTHAZAR
1	TRAIN_00002.jpg	SIMON
2	TRAIN_00003.jpg	BENES
3	TRAIN_00004.jpg	LA LOVE
4	TRAIN_00005.jpg	DAPHNE
5	TRAIN_00006.jpg	LUCIE
6	TRAIN_00007.jpg	NASSIM
7	TRAIN_00008.jpg	ASSRAOUI
8	TRAIN_00009.jpg	LAVIAN
9	TRAIN_00010.jpg	MAEVA

Viendo estos primeros datos, nos damos cuenta de que la columna "IDENTITY" es la que contiene las etiquetas con los nombres y apellidos de todos los niños, con su respectivo archivo correspondiente.

4. Planificación

Para el diseño de nuestro cronograma, decidimos utilizar un diagrama de Gantt que va desde el día posterior a la fecha de entrega de este documento hasta el día anterior a la finalización del semestre.

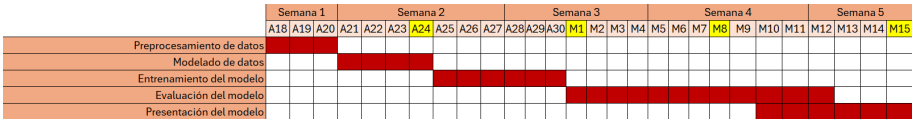

Figure 2. Diagrama de Gantt para la realización del PIA.

Estas principales etapas del proyecto serán realizadas de forma que todos los integrantes colaboremos activamente en cada una de ellas, supervisando cada aspecto antes de hacer la entrega del documento correspondiente.

La mayoría de las fases (en el mejor de los casos) no pueden llevarse a cabo al mismo tiempo debido a que cada una depende de la anterior, por ejemplo la evaluación requiere un buen entrenamiento, que a su vez necesita un modelado de los datos que tiene que ser llevado a cabo después de preprocesar la información; sin embargo, la presentación del proyecto es una excepción porque conlleva la realización de otros materiales no relacionados directamente con el modelo, como gráficas, resultados y documentos de presentación finales. También, se puede notar que la etapa de evaluación del modelo toma unos cuantos días incluso después de la fecha del documento

entregable para dicha etapa, esto es debido a que si al momento de llegar a la fecha de entrega no estamos satisfechos con los resultados del modelo, podríamos implementar cambios (por ejemplo en los hiperparámetros) con la finalidad de conseguir un mejor modelo para la fecha final de entrega del proyecto.

Affiliations

Aldo Hernández
Universidad Autónoma de Nuevo León, San Nicolás de los Garza,
aldo.hernandezt@uanl.edu.mx

Abraham López
Universidad Autónoma de Nuevo León, San Nicolás de los Garza,
abraham.lopezg@uanl.edu.mx

Damián García
Universidad Autónoma de Nuevo León, San Nicolás de los Garza,
gilberto.garciam@uanl.edu.mx

Cristian Antonio
Universidad Autónoma de Nuevo León, San Nicolás de los Garza,
cristian.antoniosnt@uanl.edu.mx

Received: ???
Revised: ???
Accepted: ???