



universidad
de león



Escuela de Ingenierías Industrial, Informática y Aeroespacial

GRADO EN INGENIERÍA INFORMÁTICA

Prácticas Externas

-

Informe mensual II

Desarrollo de técnicas de inteligencia artificial para luchar
contra el COVID-19

Autor: María García Girón

19 de noviembre, 2021



1. Tutor de la entidad

El tutor asignado por la empresa para este proyecto es Óscar García Olalla que mantiene el cargo de científico de datos e ingeniero de I+D en Xeridia.

El correo electrónico que se nos ha proporcionado para mantener contacto con él es: oscar.olalla@xeridia.com

2. Actividades realizadas hasta el momento

Tal y como se redactó en el primer informe, hasta ahora se ha realizado una formación especializada en la materia básica que hemos debido aprender con la plataforma Kaggle. Por orden se realizaron los siguientes cursos, que nos han permitido obtener una visión más centrada de las tareas que vamos a tener que realizar durante el proyecto.

Cursos: Python, introducción a Machine Learning, curso intermedio de Machine Learning, Pandas, visualización de datos, limpieza de datos, introducción a Deep Learning y computer vision.

Una vez se terminó la fase de aprendizaje se nos encargó la tarea de realizar nuestra primera competición tal como mencione en el primer informe. Esta competición se basaba en predecir a través de un algoritmo de aprendizaje supervisado si una persona podría sobrevivir o no al desastre del Titanic.

En mi trabajo realice dos cuadernos con dos modelos de aprendizaje diferentes que voy a adjuntar abajo:

<https://www.kaggle.com/maragarcagirn/getting-started-with-titanic-mar-a>

<https://www.kaggle.com/maragarcagirn/getting-started-with-titanic-mar-a-models>

Estas últimas semanas hemos seguido con la misma metodología de trabajo individual. Se realizó la competición de Kaggle para predecir el precio de casas. Esta competición se basa en un problema de regresión en el que se intenta en base a unas características estimar la etiqueta de salida del valor de una casa.

Este es el trabajo que he realizado durante esa semana:

<https://www.kaggle.com/maragarcagirn/house-pricing-competition-mar-a>

Posteriormente en la siguiente semana se nos sugirió que realizamos la última de las competiciones de Kaggle. Esta última competición se realizó utilizando como datos de entrada imágenes, gracias a ello conocimos nuevas técnicas de tratado y procesamiento de datos.

El dataset de entrada llamado MNIST es una gran base de datos de dígitos manuscritos que se utiliza comúnmente para la capacitación de diversos sistemas de procesamiento de imágenes. En este cuaderno se realizó el trabajo en tres partes principales: la preparación de datos, el modelado de la red convolucional y la predicción de resultados.

En mi trabajo realice este cuaderno que voy a adjuntar abajo:

<https://www.kaggle.com/maragarcagirn/digit-recognizer-i-mar-a/notebook>

Por último, estas semanas también se nos ha pedido que decidiésemos la temática de nuestro trabajo de fin de grado, si deseábamos realizarlo junto a la empresa. Para ello tuve una reunión con el tutor en la que comentamos diferentes opciones y temáticas mencionando los beneficios e inconvenientes de cada una de ellas.



Finalmente se decidió realizar el TFG sobre la detección de emociones en rostros y para asentar las ideas se realizó un FactSheet sobre las diferentes cuestiones a tratar.

* Se adjunta el trabajo al final del documento.

3. Próximos pasos

Como ya se ha terminado la última fase de formación, en este momento y se va a comenzar con el proyecto final de detección de técnicas que permitan frenar la enfermedad del SARS-CoV-2.

Se nos ha dividido en dos grupos para iniciar el proyecto. En mi grupo nos encontramos Diego González, Carlos Díez y yo. Se va a realizar una reunión presencial en la empresa en la que estaremos todos los miembros del grupo el día 24 a las 12:00. A partir de ese momento se sentarán las bases de las tareas a realizar.

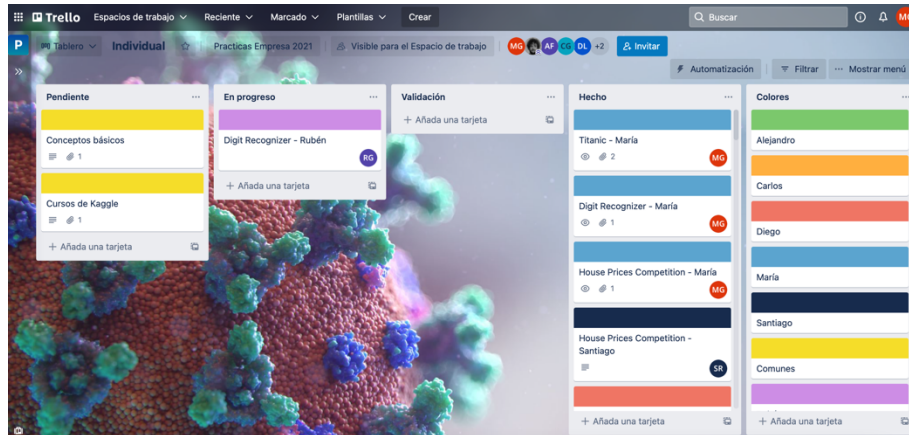
4. Planificación del trabajo

Para la planificación de nuestro proyecto seguimos utilizando la plataforma Trello. Trello es una herramienta en línea que sirve para gestionar proyectos y tareas tanto personales como grupales.

Cada miembro del grupo hemos ido añadiendo nuestras tarjetas a un tablero grupal una vez terminábamos nuestras tareas. Es una herramienta muy práctica y amplia, con funciones que permiten la comunicación entre tus compañeros de trabajo además de servir para la organización del equipo y mantener un mismo ritmo.



Esta es la visión de nuestro tablero actualmente:

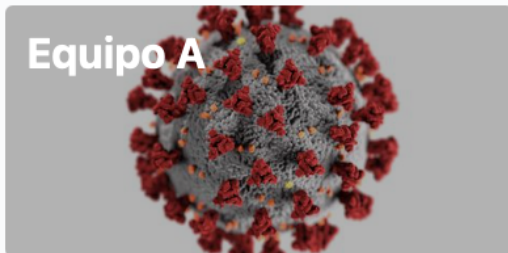


También se han creado dos nuevos tableros para cada grupo ahora que vamos a comenzar con el proyecto final. En ellos iremos añadiendo tanto tareas individuales como ya tareas grupales.

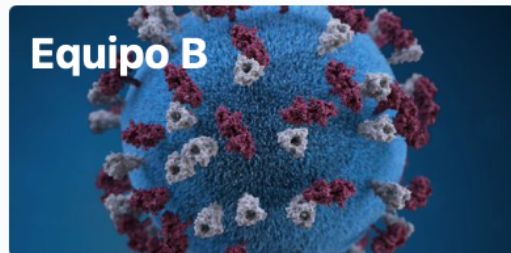


Tableros del Espacio de trabajo

Equipo A



Equipo B



Detección de expresiones faciales

Hoy en día, el aprendizaje profundo es una técnica que se lleva a cabo en muchas aplicaciones y en estudios relacionados con la visión por computadora. En este proyecto se tratan ambos campos para solventar el problema de la detección de emociones en seres humanos.



Resumen del proyecto

Este estudio tiene como objetivo construir un sistema de reconocimiento de expresión facial basado en un modelo de red neural convolucional (CNN), para clasificar automáticamente las expresiones presentadas en la base de datos de FER2013 y AffectNet.

Tecnologías a usar

Para llevar a cabo este proyecto se utilizará el entorno de desarrollo Jupyter Notebook en lenguaje Python. Se necesitarán bibliotecas específicas de Deep Learning como Tensorflow y Keras. Además de librerías para procesamiento de datos como Numpy, Scipy, Matplotlib y Pandas. Se tiene pensado usar si fuera necesario también la biblioteca para visión artificial OpenCV.



** Es posible tener que contratar algún servicio para aumentar el procesamiento*

<https://www.ovhcloud.com/es-es/public-cloud/gpu/>

Datasets

Para entrenar los modelos se podrán utilizar estos dos conjuntos de datos:

AffectNet es un gran conjunto de datos de expresiones faciales con alrededor de 0.4 millones de imágenes etiquetadas manualmente con la presencia de ocho expresiones faciales (neutral, feliz, enfado, tristeza, miedo, sorpresa, disgusto y desprecio)

Fer2013 que contiene aproximadamente 30.000 imágenes de diferentes expresiones y sus etiquetas principales se pueden dividir en (neutral, feliz, enfado, tristeza, miedo, sorpresa, y asco)

Objetivo Principal

El principal objetivo es entrenar un modelo que permita clasificar las 7 emociones que están etiquetadas. Existe mucho margen para mejorar con los estudios que se han realizado en el pasado, ya que los mejores resultados son de un 62% aproximadamente en el dataset de AffectNet.

Posibles problemas y soluciones

El dataset puede que no contenga suficientes datos, pero se utilizarían técnicas de aumentado de datos (Data Augmentation). Es necesario también utilizar técnicas de limpieza y procesamiento de datos con imágenes.

Se va a intentar utilizar ambos datasets en el problema, pero si no funcionase el modelo correctamente existen muchos datasets relacionados con lo cual sería posible modificarlo. Es posible que esta temática no consiga mejorar los resultados obtenidos, si ocurre este caso se intentaría buscar algo que permita diferenciar el proyecto de los que ya estén publicados.

Posibles mejoras

Existen numerosas variaciones que se pueden realizar si una vez entrenado no se consiguiera mejorar los resultados actuales:

- Crear una aplicación que a través de la composición de imágenes analizara un vídeo y realizase gráficas de la emoción de la persona que está siendo grabada.
- Podríamos tapar la zona de los ojos y tratar de analizar el efecto que tendría en nuestro modelo.
- El impacto del uso de mascarillas durante el periodo de pandemia en el reconocimiento de expresiones faciales.
- También se podría comprobar el funcionamiento del modelo añadiendo ruido y oclusiones.
- Otra mejora podría consistir en añadir audio. De esta manera además de tratar con técnicas de procesamiento de imágenes, se aprendería a tratar con procesamiento de audio.