

Documentación del refrigerador inteligente: “El Imposible”

Justificación de Machine Learning

Se utilizó el servicio de Microsoft Azure para el procesamiento de imágenes, en este caso para identificar las características físicas que definen a cada individuo, creando una anticipación acerca de su edad y género. Debido a cierta ética no se perfilará ni registrará información racial de los individuos. Microsoft Azure fue utilizado ya que su implementación en el proyecto es bastante sencilla, el servicio nos apoya haciendo el procesamiento pesado que la Raspberry Pi no puede. Nosotros enviamos la fotografía a la nube del servicio, el cual después de procesarla, esta regresa una serie de datos que nos permiten crear un perfil acerca de las compras usuales de ciertos grupos de individuos a partir de las características físicas previamente definidas.

Datasets

En este proyecto se usaron 3 distintos datasets, cada uno con un fin distinto, estos fueron los siguientes:

1. **Almacén** – El dataset de almacén contiene toda la información acerca de los productos disponibles dentro del refrigerador. Este tiene una conexión directa con el lector RFID ya que este si registra un nuevo producto lo agrega al inventario disponible, pero si se escanea un producto ya existente en el dataset, este es restado de la base de datos, si la cantidad de este producto llega a ser 0, este es eliminado del dataset.
2. **Temperatura y humedad** – El dataset de la temperatura y humedad se toma a través de un sensor que se encarga de la recopilación de datos, esta información la pasamos a la nube. De los datos previamente mencionados, se toman los últimos datos de cuando la puerta del refrigerador se encuentra abierta, después se toma un promedio de ambos datos y se despliegan en la base de datos.
3. **Usuarios** – El dataset de usuarios mantiene un registro de todas las caras, las cuales se comportan como entradas, que registra, esta información siendo desplegada, mostrando las edades y sexos registrados. El dataset también tiene una función que se conecta directamente con el dataset de almacén, ya que toma en cuenta los productos registrados o tomados por cada usuario que es registrado en la base de datos, creando una relación entre la persona y los productos que tomó.

Guía de configuración, instalación y despliegue

Una vez que se tiene el Raspberry Pi instalado y cableado, para que el programa funcione correctamente se debe de instalar las siguientes librerías, además de crear el servicio de la nube que se implementa en el software.

Librerías

Lo primero que se debe de instalar es el comando pip, para poder utilizar sus distintas funciones en la descarga e instalación de distintas librerías, esto se puede lograr usando el comando 'sudo apt-get -y install python3-pip' en la consola, una vez instalado el pip se deben de instalar las siguientes librerías, utilizando los siguientes comandos en consola:

- **json** – 'pip3 install json'
- **jwt** - 'pip3 install jwt'
- **adafruit_dht** - 'pip3 install adafruit_dht'
- **taho.mqtt.client** - 'pip3 install taho.mqtt.client'

subsecuentemente se debe de copiar el repertorio de mfrc522 utilizando 'git clone' con el link: <https://github.com/mxgxw/MFRC522-python>

Una vez que las librerías necesarias se encuentra descargadas en el Raspberry Pi, es importante clonar el repositorio donde se encuentra el programa, este se puede clonar en la dirección: <https://github.com/tec-csf/reto-iot-en-supermercado-2019-los-chocoflanes>, después se debe de modificar el archivo Almacen.csv encontrado en la carpeta de DB, este debe de contener las ubicaciones de los archivos adecuados de forma correcta.

El siguiente paso consiste en crear las llaves adecuadas para el funcionamiento del software con el programa encontrado en la nube, ya que este es utilizado a lo largo de la computación del programa. Para generar la llave publica y privada que se utilizara con la nube se utiliza el comando 'openssl req -x509 -newkey rsa:2048 -keyout demo_private.pem -nodes -out demo.pub -subj "/CN=unused"' en consola.

Ahora es momento de crear el servicio de la nube y enlazarlo con nuestro programa, para esto utilizaremos google cloud, una vez dentro del servicio se deben de crear 3 tablas, cada una de estas subsecuentes a los datasets previamente definidos, lo siguiente a hacer es crear los métodos pub/sub para el sistema. Después se debe de utilizar la llave publica creada para vincular la respuesta de nuestro programa con la nube y cambiar la configuración de "user to cloud" para que el id del proyecto sea el mismo que se encuentra en nuestro registro

Una vez que todo lo previo sea configurado y preparado el programa esta listo para ser ejecutado y que trabaje de forma correcta, el programa que se utilizara es llamado Refri.py el cual se encuentra en la carpeta de backend del repositorio descargado.