



Introducción a Cognitive Computing

Proyecto

SafeDrive

Ciclo académico: 2022-2

Instructor: Jaime Farfán

Integrantes:

Alex Rodriguez

Hans Burmester

Mauricio Rivera

ÍNDICE

RESUMEN

INTRODUCCIÓN

PROBLEMÁTICA

PROPUESTA DE SOLUCIÓN

METODOLOGÍA

PROCESO

RESULTADOS

TRABAJOS A FUTURO

CONCLUSIONES

ANEXO

RESUMEN:

En el presente se aplican algunos de los conceptos aprendidos a lo largo del curso de Cognitive Computing; dentro de este destacan el uso de base de datos, comprensión y aplicación de máquinas virtuales, y manipulación y creación de contenedores. Como prueba de ello, el proyecto propuesto conecta los servicios brindados por AWS (Amazon Web Services) tanto de instancias para máquinas virtuales (EC2) como en el uso de base de datos (RDS). El proyecto propone implementar un sistema de alerta que permita a los conductores de camiones y otros vehículos (controlados por centrales) mantenerlos despiertos en caso se detecte una aproximación a quedarse dormido. El alcance que tendrá involucra tanto al conductor; manifestándose mediante un aviso sonoro, como a la central, brindándole información detallada del incidente.

INTRODUCCIÓN:

La conducción de vehículos es una tarea logística que hoy en día se realiza constantemente. Su mayor uso es el transporte de materiales desde un punto a otro. Sin embargo, no es una tarea segura ni sencilla de realizar. En primer lugar, la persona encargada de realizar la tarea generalmente recorre las mismas rutas por lo que se vuelve tediosa la repetición de la tarea. En segundo lugar, la posición de descanso en la que se encuentra la persona al conducir puede conllevar a un estado de relajación y en algunos casos de fatiga, dada la cantidad de horas que el conductor debe permanecer en dicha posición. Esto genera cansancio y pesadez en el conductor lo que lo lleva a pobres condiciones laborales y finalmente reduce su efectividad y confiabilidad en el desarrollo de esta.

PROBLEMÁTICA:

El transporte de personas o cargamentos es un trabajo de suma responsabilidad y exigencia dado que el más mínimo descuido y se podrían perder vidas o podrían generar un gran impacto negativo a la compañía. Toda la responsabilidad cae sobre el conductor de bus o camión el cual debe permanecer completamente despierto y atento para realizar dicha labor sin que ocurran accidentes. No obstante, es común que debido a la sobrecarga y exigencia del trabajo, el conductor tienda a quedarse dormido debido al cansancio y al no tener a nadie que lo mantenga despierto todo el tiempo, los riesgos de accidentes se elevan bastante.



PROPUESTA DE SOLUCIÓN:

En este trabajo nosotros planteamos la incorporación de Inteligencia artificial (AI) para que esté pendiente de que el conductor no se quede dormido y, además, reportar, en tiempo real, a la central, la cantidad de veces en los que el conductor cometió una incidencia (estuvo cerca de quedarse dormido) de forma que si superase un número determinado de incidencias, se le clasifica con un estado diferente, indicando que deben tomarse acciones con respecto a este conductor (ayuda, despido, descanso, sesiones de entrenamiento, etc.).

La AI que se usó fue un detector de ojos abiertos y cerrados diseñado por nosotros. El modelo de machine learning utilizado fue Bag of Words (de accuracy de 82%) el cual recopila una serie de features (utilizando SIFT como detector y descriptor) y dependiendo de la cantidad de features detectados clasifica de forma binaria si los ojos de la persona están abiertos o cerrados, de estar cerrados por más de 3 segundos, el algoritmo indica al sistema hacer sonar una alarma, además que genera una incidencia la cual queda registrada en la base de datos de la empresa a nombre del conductor, grabando además datos como la identificación de este, en que lugar ocurrió (coordenadas), cuando (hora), etc.

METODOLOGÍA:

1. Modelo detector

Como primer objetivo, se buscó obtener un modelo lo suficientemente preciso al detectar los ojos cerrados o abiertos de la persona. De esta manera, se podrá indicar el estado del conductor utilizando una cámara y un procesador simple que evalúe los frames entregados por dicho sensor.

Este se logró mediante Google Colab, sin embargo es posible replicar el uso de cualquier entorno de desarrollo integrado que permite conexión al hardware (cámara) y que cuente o sea compatible con suficientes librerías de visión computacional para el desarrollo del detector.

2. Aplicativo web

El aplicativo web propuesto debía integrar un actor principal encargado de visualizar y manejar la información que el proyecto provea: las centrales. Dado que estas podrán ingresar a una cuenta provista, tendrán acceso completo a las funcionalidades del aplicativo las cuales serán:

- Creación, edición y eliminación de conductores
- Visualización de incidencias por conductor, y en totalidad (por central)
- Eliminación de incidencias
- Iniciar y cerrar sesión

3. Base de datos

La base de datos proporcionará la organización de la información de tanto las centrales, los conductores por central y las incidencias de cada central. Dado que tendrán datos asociados entre sí, la tabla de conductores e incidencias serán relacionales. (De la misma forma con la tabla de centrales sin embargo, esta se considera una tabla principal según la central logeada).

- Centrales
 - Código
 - Nombre
 - Teléfono
 - Dirección
 - Clave
 - Suma de incidencias

- Conductores
 - ID Conductor
 - Nombre
 - Apellido
 - Edad
 - Cantidad de incidencias
 - Estado de conductor
 - Placa camión
- Incidencias
 - ID Incidencia
 - Latitud
 - Longitud
 - Nombre
 - Apellido
 - Hora de infracción
 - Placa camión

4. Docker

Se utilizará un contenedor a manera de aligerar la carga de software que contendrá la instancia que guardará y permitirá la ejecución del programa. Dado que es posible instalar y correr el contenedor dentro del servicio de Amazon (EC2) y este es lo suficientemente ligero se podrá utilizar la versión de prueba sin una liquidación adicional.

5. Host

Como se mencionó anteriormente, el objetivo es utilizar el poder de computación en la nube y así no utilizar alguna herramienta física. Esto a su vez, elimina el costo de instalación y compra de hardware como también de alquiler o compra de un espacio físico.

PROCESO:

- Ingresar a la cuenta
- Agregar conductor
- Identificar a conductor mediante ID
- Encender el camión y con ello la cámara y el modelo
- Visualizar incidentes del conductor durante su viaje

RESULTADOS:

Interfaz web:

- Página de inicio

← → C No es seguro | 3.87.71.174/login

SaveDrive Inicio Conductores Incidencias Logout

Iniciar sesión

Empresa dedicada a la seguridad de los transportistas de su empresa.



Con SaveDrive usted puede tener mayor confianza sobre la seguridad de sus transportistas ya que nuestro moderno sistema incorpora una Inteligencia Artificial (IA) la cual advertirá sobre la somnolencia y lanzará una alarma acústica al conductor y se enviará un aviso a la central. Con SaveDrive puede tener un registro sobre las incidencias de cada empleado y de esta forma obtener un mejor criterio para calificar el desempeño.

Nombre

Nombre de la empresa

Contraseña

Contraseña

INICIAR SESIÓN

- Conductores

← → C No es seguro | 3.87.71.174/conductores

SaveDrive Inicio Conductores Incidencias Logout

AGREGAR CONDUCTOR

Incidentes	Nombre	Apellido	Edad	Cantidad_Incidencias	Estatus_Conductor	Placa_Camion	Nombre_Central	
INCIDENCIA ->	Alex	Rodríguez	22	3	0	GAN-693	Prosegur	EDIT DELETE
INCIDENCIA ->	Mauricio	Rivera	20		0	GVU-671	Prosegur	EDIT DELETE

- Incidencias de un determinado conductor

The screenshot shows a web-based application titled "SaveDrive". At the top, there is a navigation bar with links for "Inicio", "Conductores", "Incidencias", and "Logout". Below the navigation bar is a table displaying three rows of incident data. Each row contains the following columns: ID Incidencia, Latitud, Longitud, Nombre, Apellido, Hora de infracción, and Placa de vehículo. To the right of each row is a red "DELETE" button.

ID Incidencia	Latitud	Longitud	Nombre	Apellido	Hora de infracción	Placa de vehículo	
8	-7.0	-146.0	Alex	Rodriguez	7:12	GAN-693	<button>DELETE</button>
9	1.0	103.0	Alex	Rodriguez	1:10	GAN-693	<button>DELETE</button>
10	-41.0	6.0	Alex	Rodriguez	6:31	GAN-693	<button>DELETE</button>

- Incidencias de todos los conductores de la central

This screenshot is identical to the one above, showing the same list of incidents for driver Alex Rodriguez. The table structure and data remain the same, with three rows of incident details and a "DELETE" button for each.

ID Incidencia	Latitud	Longitud	Nombre	Apellido	Hora de infracción	Placa de vehículo	
8	-7.0	-146.0	Alex	Rodriguez	7:12	GAN-693	<button>DELETE</button>
9	1.0	103.0	Alex	Rodriguez	1:10	GAN-693	<button>DELETE</button>
10	-41.0	6.0	Alex	Rodriguez	6:31	GAN-693	<button>DELETE</button>

Conexión con la nube:

- Instancia EC2 con IP: 3.87.71.174

Resumen de instancia de i-029db4d4504957d42 (proy_icc)

ID de la instancia	Dirección IPv4 pública 3.87.71.174 dirección abierta	Direcciones IPv4 privadas 172.31.83.103
Dirección IPv6	-	DNS de IPv4 pública ec2-3-87-71-174.compute-1.amazonaws.com dirección abierta
Tipo de nombre de anfitrión	Nombre de IP: ip-172-31-83-103.ec2.internal	Direcciones IP elásticas -
Responder al nombre DNS de recurso privado IPv4 (A)	Nombre DNS de IP privada (solo IPv4) ip-172-31-83-103.ec2.internal	Hallazgo de AWS Compute Optimizer Suscribirse a AWS Compute Optimizer para recibir recomendaciones.
Dirección IP asignada automáticamente	3.87.71.174 [IP pública]	Más información
Rol de IAM	Tipo de instancia t2.micro	Nombre del grupo de Auto Scaling -
-	ID de VPC vpc-07cb8481151fceef	
-	ID de subred subnet-042c27d386d1b9d9c	

Detalles | Seguridad | Redes | Almacenamiento | Comprobaciones de estado | Monitoreo | Etiquetas

Detalles de la instancia

Plataforma Ubuntu (inferido)	ID de AMI ami-0149b2da6ceec4bb0	Monitoreo desactivado
Detalles de la plataforma	Nombre de AMI	Protección de terminación

- Instancia RDS con puerto de enlace:
dbproyecto.chgqdjgc18b3.us-east-1.rds.amazonaws.com

Resumen

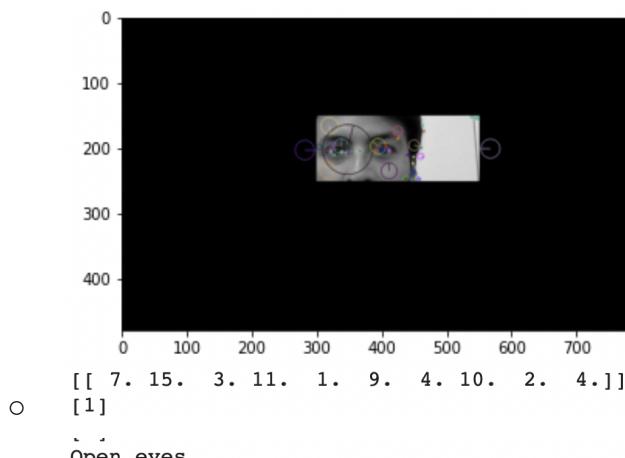
Identificador de base de datos dbproyecto	CPU 1.45%	Estado Disponible	Clase db.m6i.large
Role Instancia	Actividad actual 0.00 sessions	Motor MySQL Community	Región y AZ us-east-1c

Conectividad y seguridad

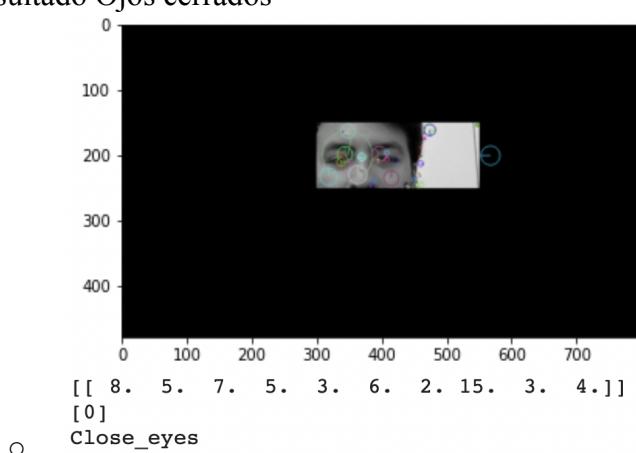
Punto de enlace y puerto dbproyecto.chgqdjgc18b3.us-east-1.rds.amazonaws.com Puerto 3306	Redes Zona de disponibilidad us-east-1c VPC vpc-07cb8481151fceef	Seguridad Grupos de seguridad de la VPC rds-proy (sg-0e4b9ad13334b438a) Activo Accesible públicamente Sí Entidad de certificación
---	--	---

Funcionamiento del programa

- Resultado Ojos abiertos



- Resultado Ojos cerrados



TRABAJOS A FUTURO:

A partir del trabajo realizado creemos que hay tres puntos que creemos se podrían mejorar y de esa manera brindar un mayor valor al proyecto. En primer lugar, utilizar un modelo de machine learning más robusto, complejo y de mayor precisión, para ello recomendamos el uso de un modelo de red neuronal convolucional (CNN). En segundo lugar, incrementar el dataset y hacerlo más variado y adecuado al uso, para el fin del proyecto utilizamos el dataset de una sola persona del grupo en 3 momentos diferentes, lo que creemos es que sería mejor alimentar al modelo con más personas y que estas tengan características de los conductores (ropa, género mayoritario, etc) y con el background común de estos. Por último, creemos que los datos que se utilizan en nuestra interfaz podrían ser mostrados de una forma más atractiva o más fácil de entender a la vista como lo puede ser un gráfico informativo armado con los datos de las tablas por ejemplo.

CONCLUSIONES:

- Se consiguió diseñar un modelo de machine learning el cual sea capaz de identificar si la persona está con los ojos abiertos o cerrados, en la mayoría de los casos. Mediante

una máquina de estados permitimos al sistema saber en qué momento iniciar la alerta y a su vez enviar información a la base de datos.

- Se consiguió almacenar los datos obtenidos por el modelo en una base de datos en la nube de AWS desde la cual se extrajeron los datos mostrados a la interfaz web construida.
- Se consiguió diseñar una interfaz web tomando como base el modelo trabajado en clase pero diferente a este con diversas opciones incorporadas para el objetivo hipotético de la empresa. Cabe mencionar su desarrollo en una plataforma gráfica y local, para luego su despliegue.
- El uso de docker (contenedores) permitió reducir significativamente el peso del programa así como la posibilidad de creación de imágenes del programa desarrollado para su fácil duplicidad.

ANEXO:

<https://github.com/hburmester/proyecto-ICC>