

CIENCIAS DE LA COMPUTACIÓN

INTELIGENCIA ARTIFICIAL
MACHINE LEARNING
DEEP LEARNING

TÍTULO PRÁCTICA: Machine Learning, Deep Learning y despliegue de Servicios

OBJETIVO:

Desarrolla modelos para realizar la clasificación de patrones mediante machine learning y deep learning
Despliega servicios de machine learning

INSTRUCCIONES:

1. Revisar el contenido teórico del tema
2. Profundizar los conocimientos revisando los libros guías, los enlaces contenidos en el material de aprendizaje y la documentación disponible en fuentes académicas en línea
3. Desarrollar las actividades de la sección “**ACTIVIDADES POR DESARROLLAR**”
4. Subir al AVAC las evidencias de desarrollo de las actividades

ACTIVIDADES POR DESARROLLAR

Revisar las guías de ejemplo de:

<https://genscinet.com/deep-learning-cnn-cifar10/>

<https://genscinet.com/despliegue-servicio-machine-learning-flask-django/>

1. Fase de Preparación: A partir del conjunto de datos “CIFAR 100” realizar un análisis exploratorio de las imágenes: visualizar algunas imágenes e indicar el shape de cada imagen ejemplo: (32,32,3). Presentar un análisis de frecuencia de clases. Se puede realizar un muestreo de al menos un 10% del conjunto de datos, manteniendo la misma distribución de clases. Presentar un análisis de frecuencia de clases del muestreo.

2. Fase de Modelado: Desarrollar una SVM

3. Fase de Modelado: Desarrollar una Red Neuronal Convolutacional (CNN)

Los Archivos denominados “FASE_PREPARACION_CIFAR100 CNN” y “FASE_PREPARACION_CIFAR100 SVM”, realizan la parte 1, 2 y 3, donde se realizar la preparación de datos con cierta cantidad de datos totales del dataset CIFAR100, además también esta el proceso para la creación para el desarrollo de la CNN y SVM.

4. Comparar los modelos en una tabla como la siguiente:

Modelo	Accuracy	Precision	Recall	F1-Score
SVM	0.16	0.15	0.16	0.15
CNN	0.16	0.1734	0.16	0.1538

1. **Fase de Predicción de Nuevos Samples:** Realizar la predicción con un nuevo sample (imagen) con los dos modelos, tanto con la SVM y con la CNN.

Además, se realizó la comprobación del uso de los modelos SVM y CNN los puede visualizar en el archivo “Fase de uso de los modelos CNN y SVM”

2. Desarrollar un **proyecto en Django** para consumir los dos modelos. Para ello, crear dos servicios de Machine Learning:
 - a) Servicio para clasificar una imagen utilizando la SVM.
 - b) Servicio para clasificar una imagen utilizando la CNN. Probar los servicios desde un cliente web.

En el siguiente enlace podrá encontrar el proyecto además agregare unas imágenes de los resultados obtenidos

<https://github.com/ChristianJapon/ProyectoCIFAR.git>

Aquí podemos observar que estamos arrancando el proyecto de manera local

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GIT LENS

```
* History restored
yectoCIFAR> ^C
PS D:\P63\Aprendizaje Automattico\Unidad 2\Proy
yectoCIFAR> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified some issues:

WARNINGS:
polls.Vote: (models.W042) Auto-created
primary
key used when not defining a primary k
ey type
, by default 'django.db.models.AutoFiel
primary key used when not defining a pr
imary key type, by default 'django.db.m
odels.AutoField'.
HINT: Configure the DEFAULT_AUT
O_FIELD setting or the AppConfig.default_auto_field attribute to point to a subclass of AutoField, e.g. 'django.db.models.BigAutoField'.

System check identified 1 issue (0 silenced).
January 03, 2024 - 01:25:10
Django version 5.0, using settings 'ProyectoCIFAR.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Page not found (404)

Directory indexes are not allowed here.

Request Method: GET
Request URL: http://127.0.0.1:8000/
Raised by: django.views.static.serve

Using the URLconf defined in ProyectoCIFAR.urls, Django tried these URL patterns, in this order:
1. ^swagger(?P<format>\.json|\.yaml)\$ [name='schema-json']
2. ^swagger/\$ [name='schema-swagger-ui']
3. ^redoc/\$ [name='schema-redoc']
4. admin/
5. ^nuevasolicitud/\$
6. ^predecir/ [name='predecir']
7. ^predecirIOjson/
8. ^static/(?P<path>.*)\$
9. ^(?P<path>.*)\$
The empty path matched the last one.

You're seeing this error because you have DEBUG = True in your Django settings file. Change that to False, and Django will display a standard 404 page.

Luego pasamos a entrar a nuestra nueva solicitud

Un formato simple que es entendible para el usuario, ademas maneja sistema de errores como tratar de predecir sin agregar una imagen

Predicción de Imágenes

Seleccionar archivo

Sin archivos seleccionados

Predecir

Por lo cual si no agregamos la imagen este nos da un aviso

Predicción de Imágenes

Seleccionar archivo Sin archivos seleccionados



Selecciona un archivo.

Luego de seleccionar la imagen podemos dar a predecir

Predicción de Imágenes

Seleccionar archivo carro1.jpeg

Predecir

Imagen cargada:



Y como podemos observar en la siguiente imagen nos indica que es lo que cada modelo predijo

Predicción de Imágenes

Seleccionar archivo

Sin archivos seleccionados

Predecir

MODELO CNN:

Clase predicha por CNN: pickup_truck

MODELO SVM:

Clase predicha por SVM: chimpanzee

8. Presentar **conclusiones y referencias** en formato APA

CONCLUSIONES


El desarrollo del proyecto en Django proporciona una estructura modular clara que facilita la organización y mantenimiento del código. La arquitectura basada en el patrón Modelo-Vista-Controlador (MVC) de Django permite separar la lógica de la aplicación, mejorando la escalabilidad del sistema. La capacidad de agregar nuevas aplicaciones de manera sencilla y la flexibilidad para extender funcionalidades hacen que Django sea una elección sólida para proyectos que pueden evolucionar con el tiempo.

La implementación de modelos Convolutional Neural Network (CNN) y Support Vector Machine (SVM) en el proyecto ha demostrado ser eficaz para la clasificación de imágenes. La CNN, esta eficacia es la de poder ser usada, claramente se debe conocer perfectamente los modelos CNN y SVM ya que estos fueron entrenados de una manera específica, con un preprocesamiento de imágenes y una entrada de sus datos, es decir como ejemplo en nuestra CNN la imagen tenía que estar redimensionada y escalada lo cual se debe implementar en nuestro proyecto además de agregar unas etiquetas para que estas sean entendidas a nuestro modelo.

Django es un buen camino para realizar este tipo de proyectos pero desde mi punto de vista existen mejores maneras de presentar proyectos con la misma idea y de una manera mas cómoda de realizar para el programador.

REFERENCIAS

1. Aggarwal, C. C. (2018). *Neural Networks and Deep Learning: A Textbook*. Springer.
2. Cristianini, N., & Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines*.
3. Vincent, W. S. (2018). *Django for Beginners*. William S. Vincent.
4. Greenfeld, D. R., & Greenfeld, A. R. (2020). *Two Scoops of Django: Best Practices for Django 3.x*. Two Scoops Press.

	CIENCIAS DE LA COMPUTACIÓN	Docente: Ing. Remigio Hurtado, PhD.
	INTELIGENCIA ARTIFICIAL MACHINE LEARNING DEEP LEARNING	

5. Ameisen, E. (2018). *Building Machine Learning Powered Applications: Going from Idea to Product*. O'Reilly Media.
6. Zheng, A., & Casari, A. (2020). *Deploying Machine Learning Models*. O'Reilly Media.