> ## Data Preprocessing and Deep Learning Model Development

[ ] ↳ *4 cells hidden*

⌄ ## Cargo Navigation [Computer Vision] & Notification System [Natural Language Processing-LLM]

```
import pandas as pd
import numpy as np
import tensorflow as tf
from tensorflow.keras.preprocessing import image as keras_image
from sklearn.preprocessing import LabelEncoder
from nltk.tokenize import word_tokenize
from langchain.prompts import PromptTemplate
import re
import os
import matplotlib.pyplot as plt
from IPython.display import display, HTML
import warnings
import nltk

warnings.filterwarnings("ignore")

nltk_data_path = nltk.data.find('tokenizers/punkt')
if not os.path.isfile(nltk_data_path):
    nltk.download('punkt')

model = tf.keras.models.load_model('/content/drive/MyDrive/Data Science/Cargo Cl

df = pd.read_csv('/content/drive/MyDrive/Data Science/Cargo Classification/train

label_encoder = LabelEncoder()
label_encoder.fit(df['category'])

message = "This is Ship ID: 2823080, Cargo Type: 1. We need entry to the Singapo

def preprocess_image(filepath):
    img = keras_image.load_img(filepath, target_size=(64, 64))
    img_array = keras_image.img_to_array(img)
    img_array = np.expand_dims(img_array, axis=0)
    img_array = img_array / 255.0
    return img_array
```

```python
def extract_info_from_message(message):
    tokens = word_tokenize(message)
    message_str = ' '.join(tokens)

    ship_id_match = re.search(r'Ship ID\s*:\s*(\d+)', message_str)
    cargo_type_match = re.search(r'Cargo Type\s*:\s*(\d+)', message_str)

    ship_id = ship_id_match.group(1).strip() if ship_id_match else None
    cargo_type = cargo_type_match.group(1).strip() if cargo_type_match else None

    if not ship_id or not cargo_type:
        raise ValueError(f"Ship ID or Cargo Type not found in message. Extracted

    return ship_id, cargo_type

def get_predicted_cargo_type(image_path):
    img_array = preprocess_image(image_path)
    prediction = model.predict(img_array)
    predicted_class = np.argmax(prediction, axis=1)[0]
    cargo_type = label_encoder.inverse_transform([predicted_class])[0]
    return cargo_type, prediction

def generate_prompt(ship_id, cargo_type):
    prompt_template = PromptTemplate(
        input_variables=["ship_id", "cargo_type"],
        template="The Ship ID is {ship_id} and the Cargo Type is {cargo_type}. R
    )
    prompt = prompt_template.format(ship_id=ship_id, cargo_type=cargo_type)
    return prompt

def get_llm_response(prompt):
    print(f"\nPrompt for LLM:\n{prompt}\n")
    return "Access Granted"

def main():
    ship_id, cargo_type = extract_info_from_message(message)

    image_filename = f"{ship_id}.jpg"
    image_path = os.path.join('/content/drive/MyDrive/Data Science/Cargo Classif

    if not os.path.isfile(image_path):
        raise ValueError(f"Image file not found: {image_path}")

    predicted_cargo_type, prediction = get_predicted_cargo_type(image_path)

    display(HTML(f"""
    <div style="font-size: 18px; color: #333333; margin-bottom: 20px; background
        <h2 style="color: #007bff;">Ship Message</h2>
        <p><strong>Received Message:</strong> <span style="color: #007bff;">{mes
```

```
<p><strong>Received Message:</strong> <span style= color: #007bff, >{mes
</div>
"""))

display(HTML(f"""
<div style="font-size: 18px; color: #333333; margin-bottom: 20px; background
    <h2 style="color: #007bff;">Cargo Type Extraction and Prediction</h2>
    <h3 style="color: #28a745;">Ship ID: 2823080 Found in Database</h3>
    <p><strong>Extracted Cargo Type from Message:</strong> <span style="colo
    <p><strong>Predicted Cargo Type:</strong> <span style="color: #28a745;">
</div>
"""))

img = keras_image.load_img(image_path)
plt.figure(figsize=(8, 6))
plt.imshow(img)
plt.title(f"Image: {image_filename}", fontsize=16, color='#007bff')
plt.axis('off')
plt.show()

prompt = generate_prompt(ship_id, cargo_type)
response = "Access Granted" if str(cargo_type) == str(predicted_cargo_type)

display(HTML(f"""
<div style="font-size: 18px; color: #333333; margin-bottom: 20px; background
    <h2 style="color: #007bff;">LLM Response</h2>
    <p><strong>Cargo Type from Message:</strong> <span style="color: #28a745
    <p><strong>Predicted Cargo Type:</strong> <span style="color: #28a745;">
    <p><strong>LLM Response:</strong> <span style="color: {'#28a745' if resp
</div>
"""))

main()
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
1/1 ━━━━━━━━━━━━━━━━━━━━━ 0s 74ms/step
```

# Ship Message

**Received Message:** This is Ship ID: 2823080, Cargo Type: 1. We need entry to the Singapore's Tuas Port

# Cargo Type Extraction and Prediction

## Ship ID: 2823080 Found in Database

**Extracted Cargo Type from Message:** 1

**Predicted Cargo Type:** 1

Image: 2823080.jpg



# LLM Response

**Cargo Type from Message:** 1

**Predicted Cargo Type:** 1

**LLM Response:** Access Granted