**CNN-Based Product Image Recommendation System**

Author: Mohammed Audu

Date: 05th May 2024

**Introduction**

This report details the development of a product image recommendation system utilizing a Convolutional Neural Network (CNN) for feature extraction. The system aims to identify products from images and recommend similar ones based on extracted features.

**High-Level Flow**

1. **Data Preprocessing**:
   * Load image data and corresponding product labels from CNN\_Model\_Train\_Data.csv.
   * Preprocess images by resizing, converting to NumPy arrays, and normalizing pixel values.
2. **Model Development**:
   * Load a pre-trained ResNet50 model (without final classification layers) for transfer learning.
   * Define a CNN architecture using the pre-trained model followed by a GlobalMaxPooling2D layer for feature extraction.
   * Train the model on the preprocessed image data and product labels using an optimizer like Adam and a loss function like categorical crossentropy.
3. **Recommendation System**:
   * Extract features from a new product image using the trained model.
   * Utilize a nearest neighbor algorithm to find similar products based on the extracted features in the feature vector database.
   * Recommend the most similar products to the user.

**Descriptions**

* **CNN Model**: A deep learning architecture consisting of convolutional layers for feature extraction, pooling layers for dimensionality reduction, and fully-connected layers for classification (not used in this implementation).
* **Transfer Learning**: Leveraging pre-trained weights from a model trained on a large dataset for feature extraction in a new task.
* **Feature Extraction**: The process of extracting informative features from images that represent the product's visual characteristics.
* **Nearest Neighbors**: An algorithm that identifies data points closest to a new data point based on a distance metric (e.g., Euclidean distance).
* **Feature Vector Database**: A storage system containing pre-computed features for all products in the system.

**Diagrams**

* A high-level system architecture diagram illustrating the data flow, from image loading and preprocessing to feature extraction, recommendation generation, and user interaction.

**Key Decisions**

* **Pre-trained Model Selection**: Choosing ResNet50 as a pre-trained model based on its effectiveness in image recognition tasks.
* **Feature Extraction Technique**: Utilizing GlobalMaxPooling2D for efficient feature extraction and dimensionality reduction.
* **Nearest Neighbor Algorithm**: Selecting a nearest neighbor approach for its simplicity and efficiency in image recommendation tasks.

**Challenges and Solutions**

* **Data Imbalance**: If the training data has an uneven distribution of product categories, techniques like oversampling or undersampling can be implemented to balance the data.
* **High Feature Dimensionality**: Feature selection methods like Principal Component Analysis (PCA) can be used to reduce feature dimensionality while preserving relevant information.
* **Limited Training Data**: Employing data augmentation techniques like random cropping, flipping, or color jittering can artificially increase the training data size and improve model generalization.