**PHASE 5**

**IMAGE REGONITION WITH IBM CLOUD USING WATSON STUDIO**

Project overview:

1. Objective: Develop an image recognition system using IBM Cloud

Visual Recognition to automatically classify and analyse images based on their content.

1. Key Features: Image classification, object detection, custom training, content moderation, integration capabilities, and scalability.
2. Technology Stack: IBM Cloud Visual Recognition, programming language (e.g., Python, Node.js), and a possible frontend for user interaction.
3. Phases: Planning, data collection, model training, integration, testing, deployment, monitoring, and maintenance.
4. Success Metrics: Accuracy in image classification, response time, user satisfaction, reduced manual moderation efforts (if applicable), and scalability.
5. Team: Project manager, data scientists, software developers, UX/UI designers, QA testers, and Devops engineers.

Design Thinking:

1. Image Recognition Setup: Set up the IBM Cloud Visual Recognition service and obtain the necessary API keys.
2. User Interface: Design a user-friendly interface for users to upload images and view the AI-generated captions.
3. Image Classification: Implement the image classification process using the IBM Cloud Visual Recognition API.

4 .AI-Generated Captions: Integrate natural language generation to create captions for the recognized images.

1. User Engagement: Design features to allow users to explore, save, and share their AI enhanced images.

Incorporating sentiment analysis to generate caption that capture the emotions and mood of the images:

Incorporating sentiment analysis to generate captions that capture the emotions and mood of images is a valuable application of natural language processing and computer vision. This can be achieved using a combination of image analysis and text generation techniques.

1. **Data Collection**:
   * Gather a dataset of images with associated sentiment labels. You may use pre-labelled datasets or create your own.
2. **Sentiment Analysis**:
   * Utilize a pre-trained sentiment analysis model (e.g., BERT, RoBERTa, VADER, or any other suitable model) to analyze the sentiment of each image.
   * This model should output a sentiment score or label (e.g., positive, negative, neutral) for each image.
3. **Image Processing**:
   * Use computer vision techniques to extract visual features from the images. Convolutional Neural Networks (CNNs) are often used for this purpose.
   * These features can include object recognition, color analysis, and other visual attributes.
4. **Combining Text and Visual Features**:
   * Combine the sentiment analysis results with the visual features extracted from the images to create a comprehensive representation of the image's content.
5. **Caption Generation**:
   * Train a text generation model (e.g., recurrent neural network, LSTM, or GPT-based model) on a dataset of images and their corresponding sentiment labels.
   * The model should learn to generate captions that reflect the sentiment and mood of the image.
6. **Fine-Tuning**:
   * Fine-tune the caption generation model on your specific dataset, which includes the combined sentiment and visual features.
   * This step helps the model adapt to the specific context and sentiment patterns of your data.
7. **Inference**:
   * When you want to generate a caption for a new image, run the image through the sentiment analysis model to get the sentiment label or score.
   * Combine this sentiment information with the visual features of the image.
   * Input this combined representation into the caption generation model to generate a caption that captures the emotions and mood of the image.
8. **Post-Processing**:
   * You can post-process the generated captions to make them more coherent and grammatically correct.
9. **Evaluation**:
   * Evaluate the quality of your generated captions by human judgment or automated metrics such as BLEU, ROUGE, or METEOR.

From IBM Cloud, you need to access Watson and create a Visual Recognition service. From the created service, you have to acquire the API key for performing operations through the API.

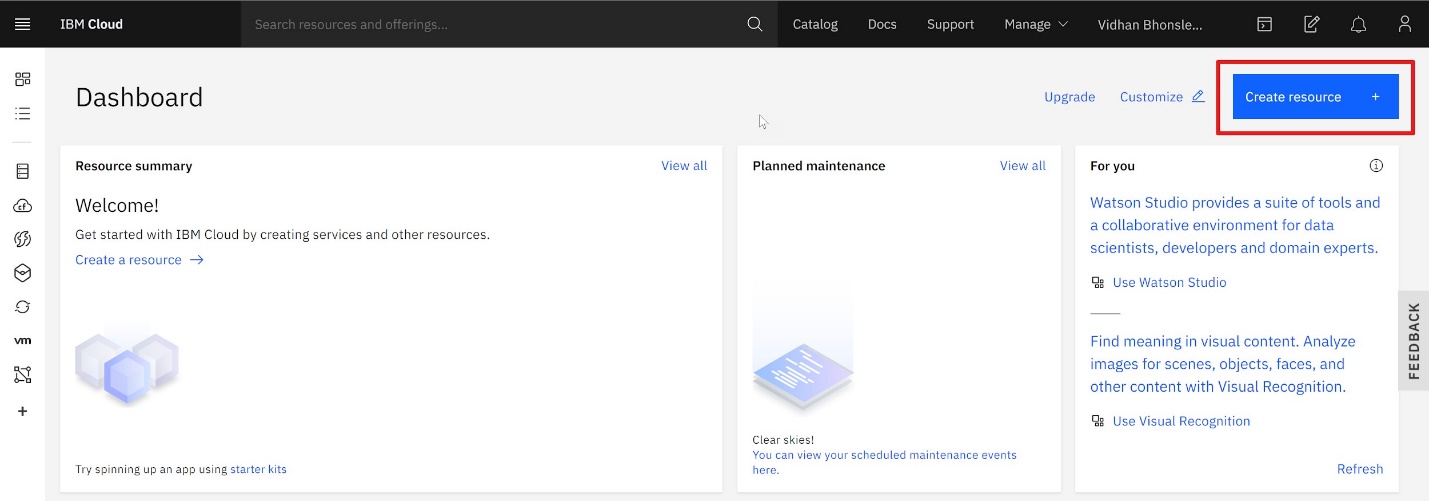
You will also need an API key from the HERE Developer Portal (which we will use in the code). Simply sign up for your free account and generate API key on your account page.

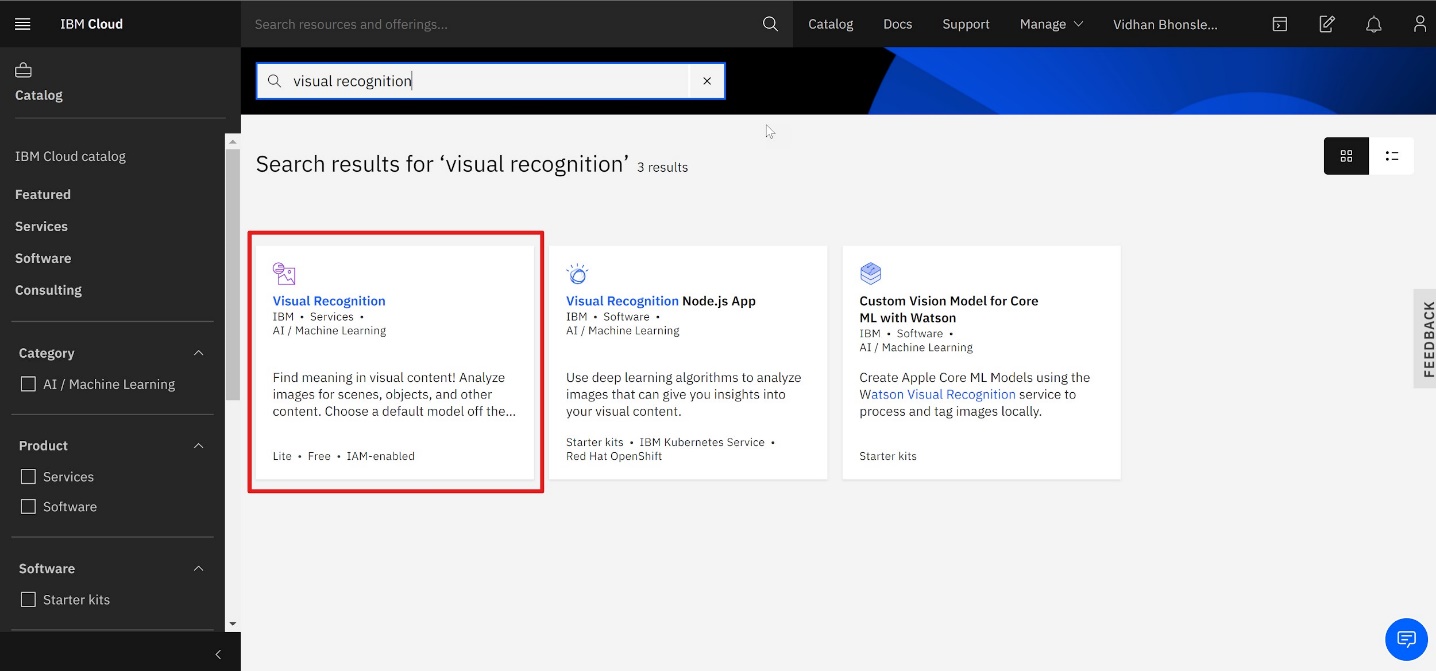
For coding, your machine should have Python installed (I have Python 3.8).

In this part, you will learn to run image recognition code on your machine and extract the required result from the JSON response. (We will cover the location element in next blog post).

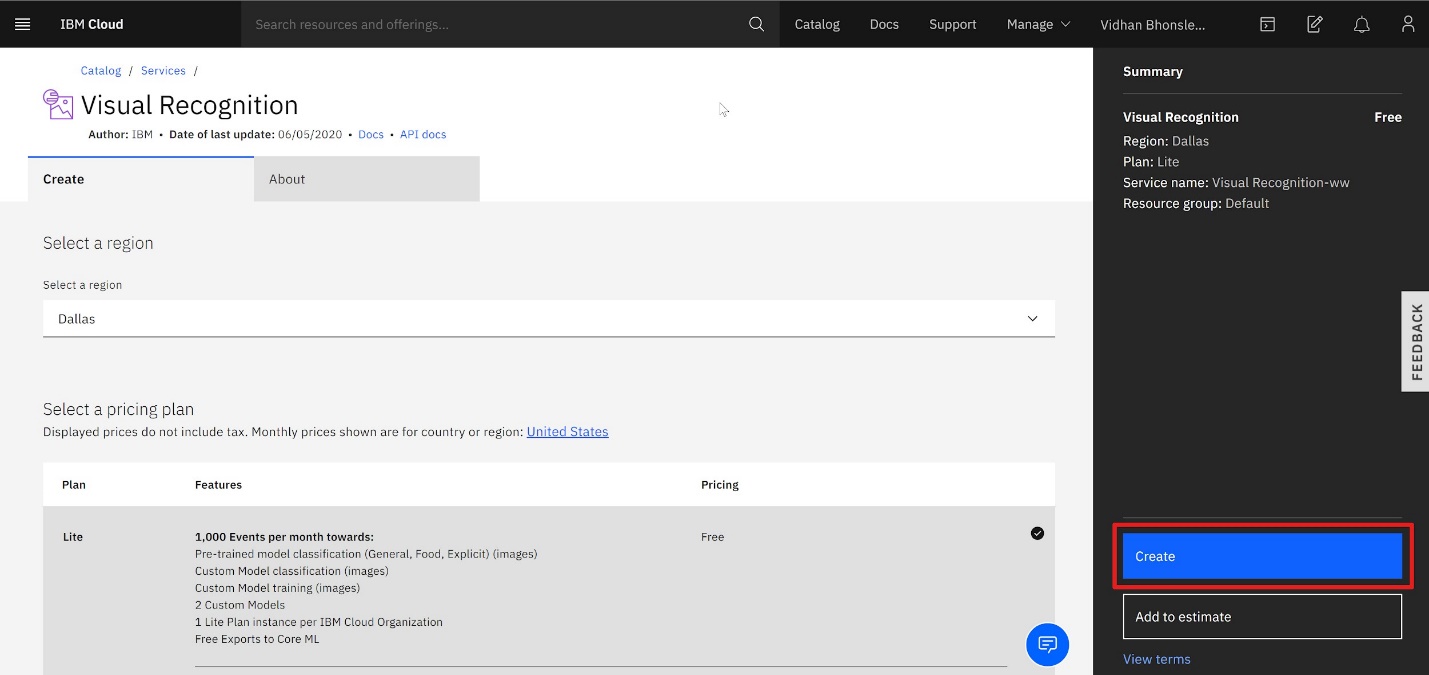
Lets get started then!

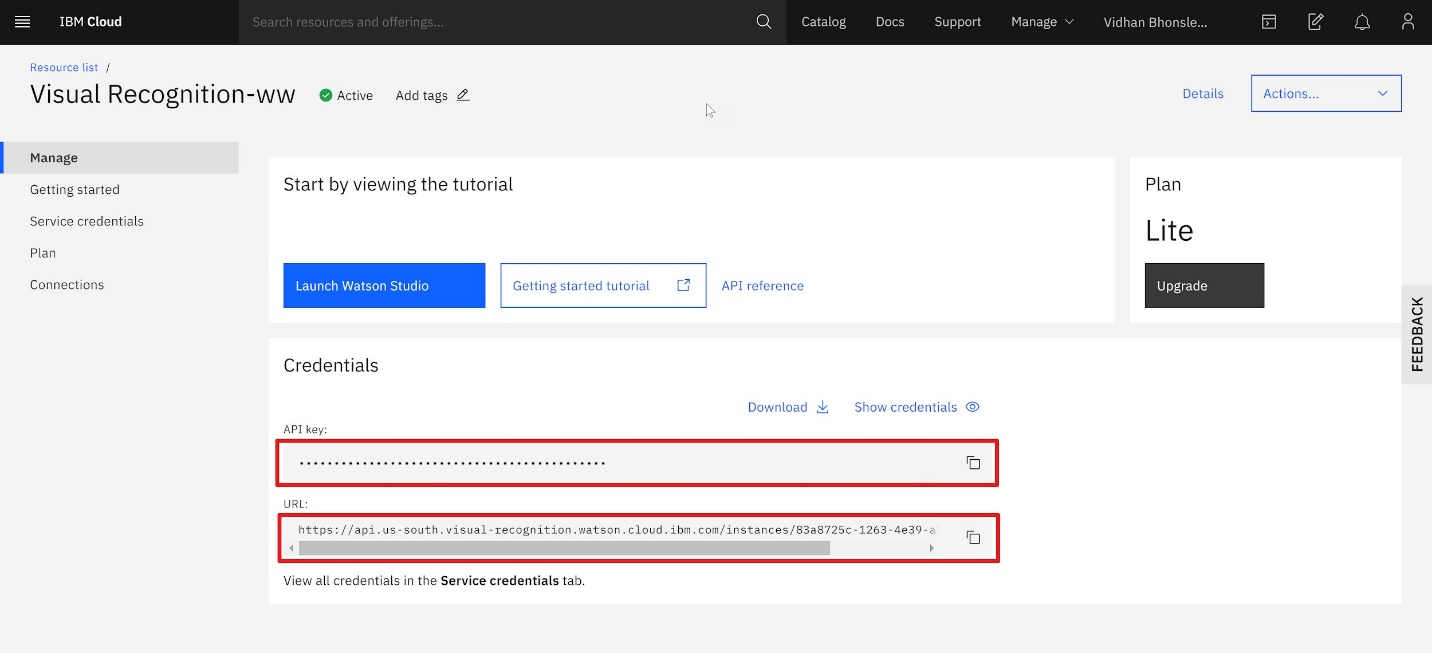
Creating Visual Recognition Service and Acquiring API key

After logging into your IBM Cloud account, you will be on the dashboard page. You need to create a service by clicking on 'Create resource' (as shown in the image below).

This will take you to the IBM Product Catalog page. There, search for 'visual recognition' (as shown below) and select the first option.

The Visual Recognition service uses deep learning algorithms to analyze images for scenes, objects and other content. And it also has a set of built-in models for ease of use.

You will reach a page where you have to select the region and configure resources. It can be edited as per your wish or can be left unaltered (see the

Finally, you will be on the page from where you need to copy the API key and URL. Keep both the values for further coding (represented as IBM API KEY and IBM URL later in the code

Writing Python Code for Visual Recognition

Once you have acquired an API key associated with the Visual Recognition service, the next step is to install Watson Developer Cloud using pip:

pip install --upgrade "ibm-watson>=4.0.1"

The library can be installed using the command prompt, terminal or a Python IDE (PyCharm, Spyder, etc), depending on how you are working with Python.



Conclusion:

The world is moving more and more toward automation and visual recognition is one of the best example of automation. The real question is how to use the output of Machine Learning and AI based model. Till then, happy learning!

Watson's Visual Recognition comes with a number of default classification models. We will use the animals for recognizing images animals.

Image recognition using IBM Cloud involves leveraging IBM's AI and cloud services to analyze and interpret visual content in images. IBM offers tools like Watson Visual Recognition, which utilizes deep learning algorithms to classify and tag images, making it easier to search and analyze visual data. This technology can be applied in various industries, including healthcare, retail, and security, to automate tasks, identify objects, and gain valuable insights from images.

Creating an image recognition system by integrating IBM Cloud Visual Recognition and AI-generated captions is a great idea. Here are the high-level steps to implement this system:

1. Set up IBM Cloud Services:

- Sign up for an IBM Cloud account if you don't have one.

- Create a Visual Recognition service instance from the IBM Cloud catalog.

2. Train or Fine-Tune the Visual Recognition Model:

- Train or fine-tune the Visual Recognition model according to your specific image classification needs. You can use the IBM Cloud Visual Recognition tools to create and customize your model.

3. Upload and Analyze Images:

- Upload images that you want to classify to the Visual Recognition service.

- Use the API to send these images for classification.

4. Receive Classification Results:

- The Visual Recognition service will return results that include labels and confidence scores for each recognized object in the image.

5. Integrate with Natural Language Generation (NLG):

- You can use an NLG tool or library to generate captions based on the recognized objects and their confidence scores. NLG systems like GPT-3 can be helpful for this.

6. Generate Captions:

- Send the recognized labels and confidence scores to the NLG system.

- The NLG system will generate captions for the image based on the provided data.

7. Display or Store Captions:

- You can choose to display the generated captions alongside the images or store them in a database for future use.

8. Iterate and Optimize:

- Continuously train and fine-tune your Visual Recognition model to improve classification accuracy.

- Monitor the quality of the generated captions and make adjustments as needed.

9. Ensure Security and Compliance:

- Be mindful of data privacy and security when working with images and captions, especially if handling sensitive content.

10. Scale and Deploy:

- Depending on your use case, you can deploy the system in a cloud environment for production use.

