**MM MODELS WITH**

**IBM WATSON**

Submitted by.

Peetla Vishnu Vardhan

Au723921104037

Vishnuvardhanpeetla@gmail.com

1.Data Preparation:

Begin by preparing your dataset and cleaning it as necessary. Watson Studio provides tools for data preprocessing and transformation.

2. Model Development :

Use Watson Studio's Jupyter notebooks or other available tools to build and train your machine learning model. You can experiment with various algorithms and techniques.

3. Model Evaluation :

Evaluate the performance of your model using metrics and techniques appropriate for your problem (e.g., accuracy, precision, recall, F1-score for classification models).

4. Model Serialization :

can Once you have a trained and evaluated model, you need to serialize it to a format that be deployed. Common formats include Python's pickle or joblib formats.

5.Creating a Deployment Space:

In Watson Studio, you typically work within a "project." Create a project for your model deployment, and within that project, create a "deployment space" to manage your deployed assets.

6. Deployment Configuration:

the number of instances to run your model Within your deployment space, configure the deployment settings. You'll specify details such as the runtime environment, the hardware configuration, and the number of instances to run your model.

7. Deployment:

Deploy your model to the IBM Cloud. Watson Studio provides a straightforward interface to select your trained model and deploy it to the cloud.

8. Scoring Endpoint :

Once deployed, you'll receive a scoring endpoint (URL) that allows you to make predictions using your model via API calls. You can integrate this endpoint into your applications.

9. Testing :

Test your deployed model to ensure it's working as expected. You can send sample data to the scoring endpoint to verify its functionality.

10. Monitoring and Management :

Watson Studio provides monitoring and management tools to track the performance of your deployed model, manage versions, and make updates when needed.

11. Scaling :

Depending on your usage and requirements, you can scale the deployed model up or down to handle varying workloads.

12. Security and Authentication :

Ensure that your deployed model is secured with appropriate access controls and authentication mechanisms.

13. Documentation and API Integration :

Provide documentation for users or developers who want to integrate your model into their applications. Explain how to make API requests and interpret responses.

14. Maintenance and Updates :

Regularly monitor and maintain your deployed model, and update it as needed to improve its performance or address changing requirements .