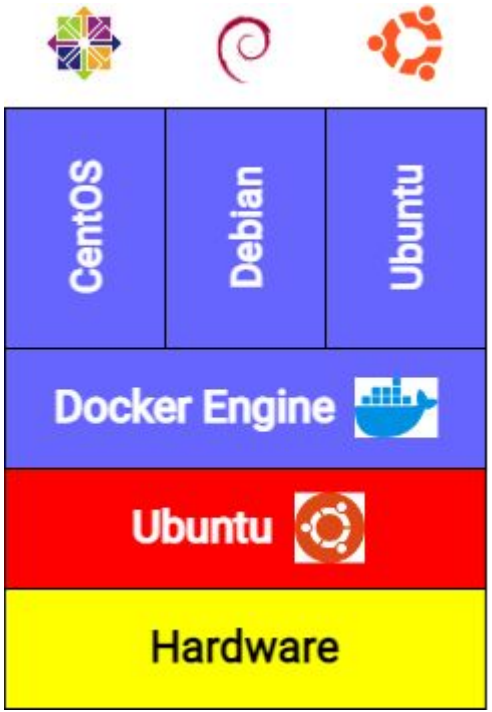
Automating the Selection of Container Orchestrators for Service Deployment

-Suryam Arnav Kalra, 19CS30050
- Kunal Singh, 19CS30025
- Aryan Agarwal, 19CS30005

INTRODUCTION

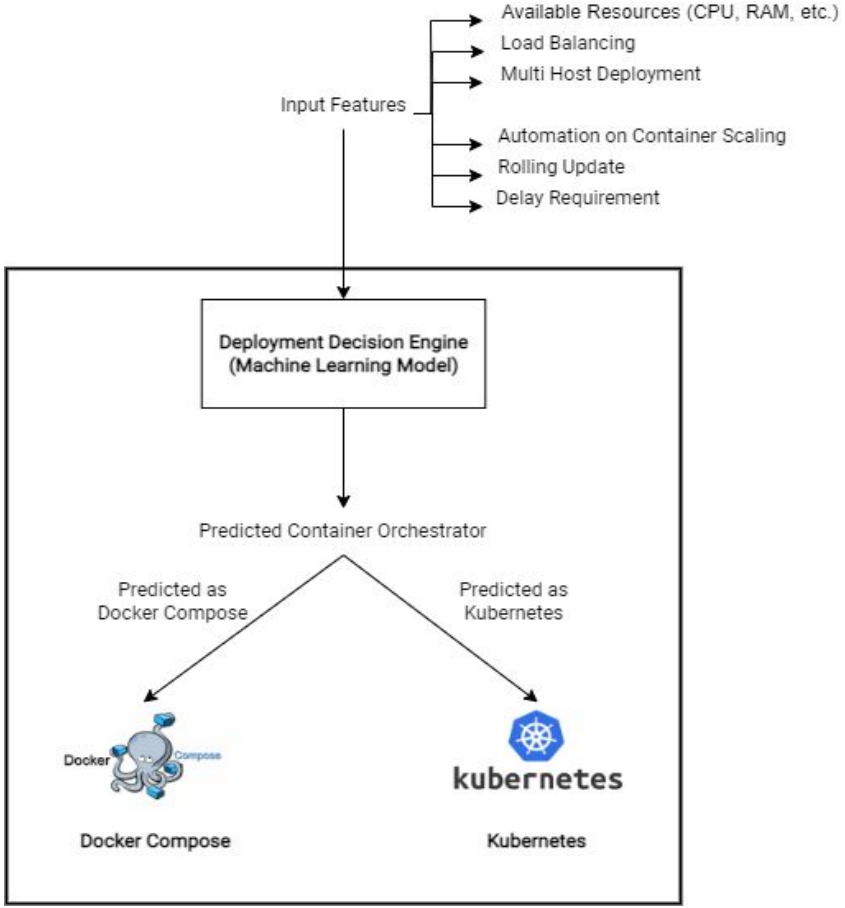
- Cloud services have traditionally been deployed as Virtual Machines (VM) in cloud provider servers
- VMs are slow -> Use containers
- Responsibility of proper resource management of containers
- An architecture to appropriately select the manager based on application needs and user demand
- ML models for effectively and efficiently choosing between Docker compose and Kubernetes



Container Architecture

PROPOSED ARCHITECTURE

- An architecture consisting of 4 layers.
  - First layer takes input from the user the requirements of the application. The features like available resources, memory, load balancing, etc. will be taken as an input and serve as the features for our decision model



- Second layer has our decision model which is a machine learning algorithm (K-Nearest neighbour and Logistic Regression).

- Third layer gives the output out of our decision engine whether to choose Docker compose based container manager or Kubernetes for the specific application of which the features are provided

- Fourth and the last layer is deployment of the application on the container orchestrator that is predicted

IMPLEMENTATION

- Made a full stack web application for our experiments. PHP with docker image and MySQL with docker base image were used
- Setup Time - Time taken by a container manager to finish the deployment

- CPU Usage - %age of CPU use (cpu usage for docker-compose based manager is very less)

- Memory Usage - %age of Memory use (memory usage for docker-compose based manager is very less)

EXPERIMENTAL RESULTS

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

MODEL RESULTS

	KNN	Logistic Regression
Accuracy	0.965	0.957
Precision	0.956	0.936
Recall	0.959	0.961
CV Mean	0.936	0.945

PROPOSED ENHANCEMENT

Proposed two machine learning models:-

- Support vector machines which are some of the simplest and arguably the most elegant methods for classification. Each object you want to classify is represented as a point in an n-dimensional space and the coordinates of this point are usually called features
- ANN(Artificial Neural Networks)

Results from our proposed models

MODEL RESULTS

	KNN	Logistic Regression	SVM	ANN
Accuracy	0.965	0.957	0.995	0.980
Precision	0.956	0.936	0.988	0.971
Recall	0.959	0.961	1.0	0.971
CV Mean	0.936	0.945	0.948	0.973

Enhancement

- First Dataset Enhancement: New full factorial design of experiment dataset. This new dataset incorporates a new feature called container self-healing
- Second Dataset Enhancement: Reliability is also added. They are vital features that users consider when selecting a container orchestrator, we have incorporated these features into our database.
- Architecture Enhancement: We have determined that the container self-healing feature is more pertinent and critical than multi host deployment, and therefore, we have replaced the latter with the former in our architecture

FUTURE WORK

- We could conduct a user survey to better understand the demands of the user based on their applications so that we can create a good custom dataset backed with statistical hypothesis tests to make our ML models learn the real world scenario.

- We could implement our proposed architecture on a cloud platform to complete the application domain of our research.

Thank You!