SIG788 – ENGINEERING AI SOLUTIONS

TASK- 2

B PRIYANKAA

s224207694

priyayj2016@gmail.com

PART- 1

1.Review the attached paper and describe the main fundamental differences to building applications and platforms for training and building applications based on ML than we have seen prior in application domains. Summarize it into 3 points.

Solution:

This paper is all about a case study focusing on an observation that was made on the teams at Microsoft to understand the challenges and best practices in developing the AI – solutions and to understand how AI is integrated into software and services. The study found three major differences in AI domain and traditional software application domains. Data availability, collection, cleaning, and management were the challenges faced by engineers. To assess the maturity of the ML work, the study also introduced a ML Process Maturity Model. This was a great help for the teams to self-assess their proficiency in ML and gave ideas on the areas of improvement. This study also found that there are certain findings that are specific to certain companies and the ML workflow used. On the whole, the research paper gave some meaningful insights into challenges, practices in ML, and maturity assessment for machine learning.

The main differences to building applications and platforms for training and building applications based on ML than seen prior in applications domain are:

- Data discovery, source, management and, versions: Machine Learning is dependent on data than the traditional software development domain. The efforts taken to complete the process of discovering, sourcing, managing and versioning of data is more complex compared to traditional domain. Consistent and significant efforts are needed to complete the process.

- **Model customization and Model extensibility:** People possessing different set of skills are needed than those already present in software engineering teams to customise the ML model and also to build the extensibility of the ML model. People who have deep knowledge on how to build, evaluate, and tune a ML model based on hyperparameters from the scratch are needed to integrate the ML model into software systems. This is another unique challenge.

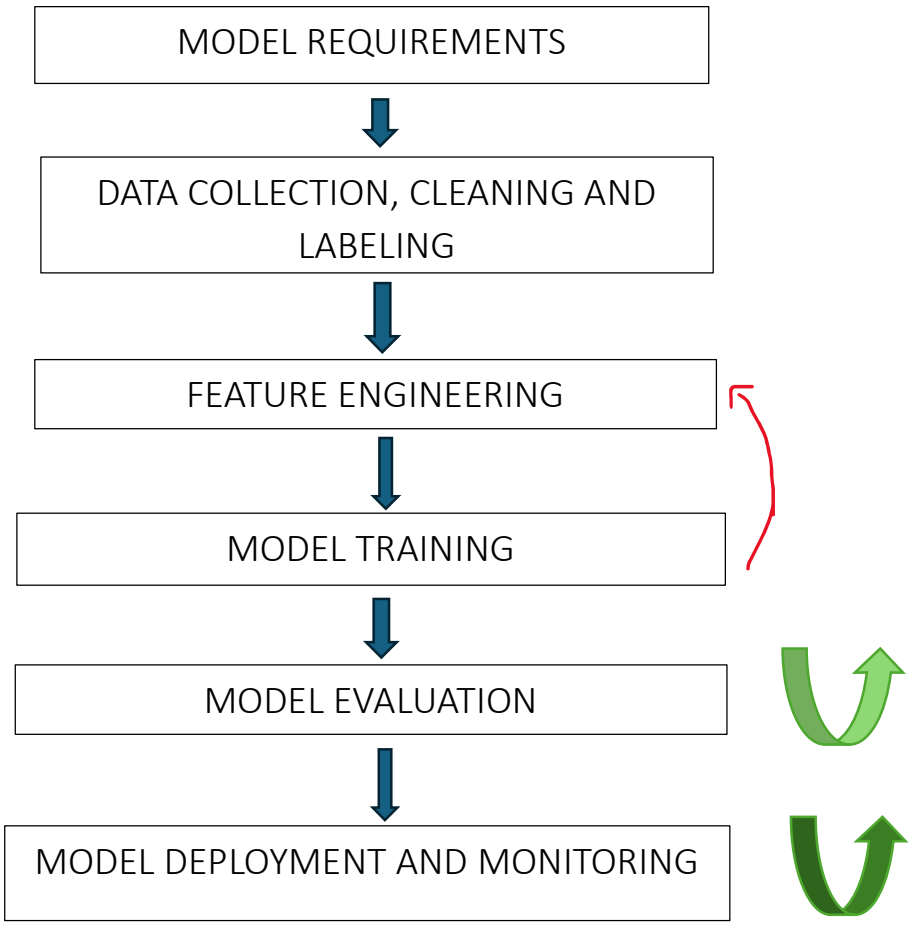
- **Module Boundaries:** Maintaining the strict module boundaries between machine learning components is more difficult than the traditional software engineering domain. During the training and tuning process, the ML models can become more complex by affecting one another even if the software teams try to build them in an isolated way.

2. What are the main stages of machine learning workflow and explain each stage briefly.

This paper describes of how several software teams at Microsoft work to integrate the ML models into applications and platforms. The main stages of machine learning workflow are:

- Model requirements,
- Data collection
- Data cleaning
- Data labelling
- Feature Engineering
- Model Training
- Model Evaluation
- Model Deployment and Monitoring

There is flowchart below that depicts the main stages of ML workflow in a pictorial way.



There are larger feedback arrows that are present near model evaluation, deployment and monitoring stages which denotes that these sections may go to any of the previous stages. But the smaller red feedback loop present near the model training stage denotes that feedback will be sent only to feature engineering stage.

- Model Requirements: This is the stage where the ML model designers decide the feasible features to implement ML and also decide the appropriate models for the problem at hand.
- Data collection, cleaning, and labelling: This is the next stage where new datasets are collected and integrate the different datasets. Here the missing values are treated, and inappropriate values are removed. The missing values are imputed either by

mean or mode. There are ground truth labels assigned to each record present in the dataset.

- Feature Engineering: This is the step to extract and select the features that are more significant to extract the meaningful insights from the ML model developed. This means to select the features needed for model training.
- Model Training: Based on the parameters set on the model function, the model will be trained and may be tuned at times to clean the data and their labels. This is the model optimization stage.
- Model Evaluation: Models are evaluated based on the performance metrics like accuracy, confusion matrix, and classification report. Here teams test the trained model on the test data and record the outputs.
- Model Deployment: The trained and evaluated model is deployed on the targeted device in the production environment. The deployed model is monitored for the errors that may occur in a real-world.

These stages represent the key components of the Machine Learning workflow.

These stages of the Machine Learning workflow are highly non-linear and contain some larger and smaller feedback loops. If there is large shift in the data distribution in the model and data in the real world then software teams may have to re-run certain steps.

Sometimes teams may have to reconsider the modelling choices made in the first stage when the new problem evolves. Everyday, the engineers must work on iterating the selected models frequently based on hyper-parameters and refinement of the datasets.

If there are multiple ML components present in the workflow, then the workflow may get complex as components may interact with each other.

3. Which domains within the Microsoft team have employed AI, and what machine learning approaches have they utilized?

Many teams in Microsoft have widened the applications of machine learning in various domains. The research team were able to divide the results into two ways among the Microsoft teams,

- ✓ In which one is all about to capture the **applications domains** in which AI is applied.
- ✓ Second is all about the **ML algorithms used** to build those applications.

Applications Domain in which AI is applied:

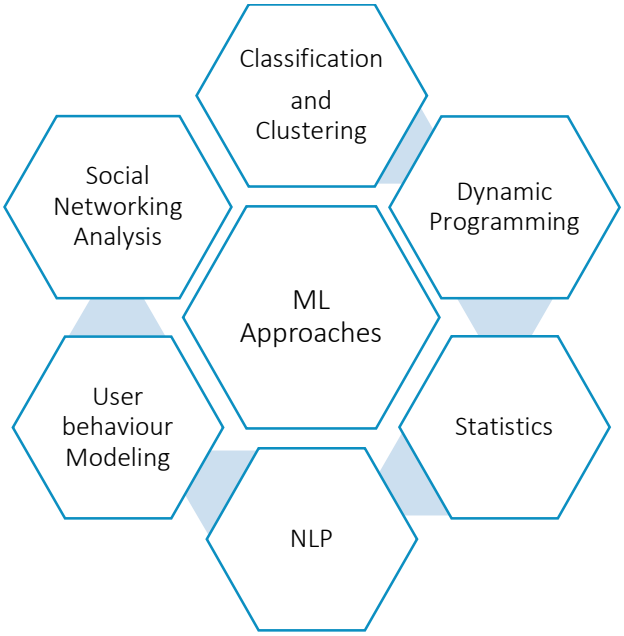
Traditional Areas:

Microsoft teams have employed AI in traditional areas such as search engines (to search on the topics that users mention), advertising (to include ads of the product on the user's web page that they searched for), machine translation, predicting customer purchases, voice recognition and image recognition to secure the important details on phone.

Novel areas:

Microsoft teams have employed AI in novel areas such as identifying the right customer leads (business teams does this to narrow down the people who will benefit from the service that they offer), in MS slides they provide design ideas for the users for presentations, to provide unique drawing features, in healthcare domains to diagnose the disease in the early stage, to improve gameplay, in fintech domains to detect the fraudulent fiscal activity and stop the money from being debited and to monitor the network streams in case of security breaches.

Machine Learning Approaches to build applications:



Classification and Clustering: Based on the features, people or data are clubbed into different groups.

Dynamic Programming: This is needed where there is a need of optimal decision to fix the price, in resource planning and in bidding process. This approach is used to improve decisions needed to make effective desired output.

Statistics: This technique is used to perform tasks like data analysis, hypothesis testing, identifying patterns and predicting the outputs.

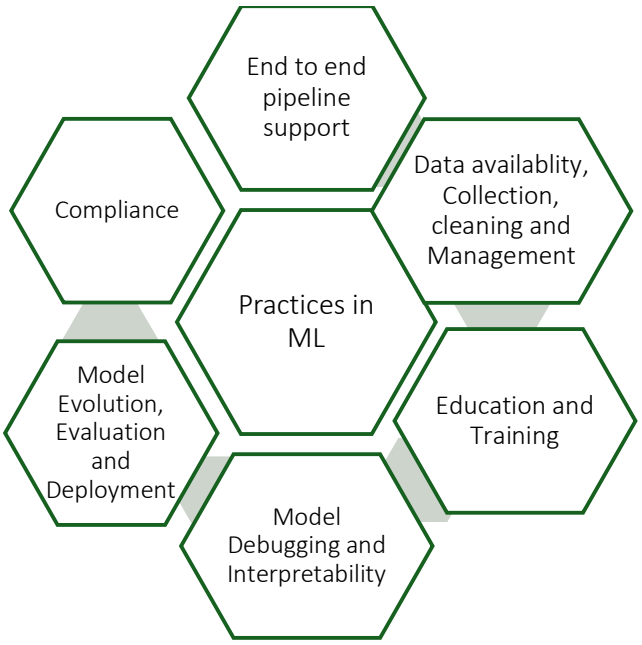
Natural Language Processing (NLP): Many teams work on NLP and develop tools for entity recognition, intent prediction, summarisation and ontology construction.

User-Behaviour Modelling: This technique is used to personalise the user experience based on the user preferences. This actually improves the user engagement of the product or service they offer.

Social Networking Analysis: This is used to analyse the relationships to identify the high valuable users.

4. What are the best practices with ML in software engineering? Provide a summary for each practice.

The best practices with ML in software engineering are:



- ✓ End to end pipeline support: This practice denotes the importance of integrating the ML to traditional software development infrastructure. This step also provides seamless development in the ML stages like data collection, cleaning, labelling, feature engineering, model training, model evaluation, model deployment and monitoring. The main components are:
  - Automation of data pipeline: Building pipelines for the flow of data and for effective processing of the data.
  - Dashboards: These provide good visualisations to effectively monitor the model. Certain metrics are improved in the model to provide value to the user using these dashboards.
  - IDE: Azure ML studio provide environment to build the ML models on large set of data for the data enthusiasts.

- ✓ Data availability, collection, cleaning and maintenance:
  There is a voluminous amount of data present in the internet from various sources like online media: webpages, certain dataset sites (medical datas), Instagram (has pattern of every user), youtube feedbacks and many more. Based on the problem given, datasets from various related sources must be collected, cleaned and management. For the proper ML workflow, the data must be cleaned when there are many missing values. The missing values can be removed but it may result in skewness of the ML model developed or can also be imputed with mean or mode values.
  Automation of data management is needed to preprocess and clean the data effectively. Data versioning must be done to ensure model reproducibility by data sharing.
- ✓ Education and training: There is an importance of education and training the software engineering teams to work effectively with the ML practices.
  - Conferences and forums must be provided for the software engineering teams on machine learning and data science.
  - KT sessions must be provided by ML specialists to the software engineering teams.
  - Online course access can be provided to ensure continuous learning.
- ✓ Model Debugging and Interpretability:
  Identifying and resolving the errors in the model is very important in the model interpretation. Certain visualization techniques are used to interpret the model effectively.
  Debugging experience must be improved based on the errors appear in real world for the model developed. Models must be rectified based on the inaccuracies occurring.

- ✓ Model Evolution, Evaluation, and Deployment:
  Frequent revisions of the model are done based on the parameters tuning and data updates. The system performance has a significant impact. Various teams employed agile techniques to evaluate the models developed by them. Multiple metrics must be included in experiment score cards. Human driven evaluation is performed for sensitive data categories. Tests are automated and then integrating the model building with the software using the common versioning repositories for ML and non-ML codebases. ML and non-ML developed sprints are coupled tightly to ensure the smooth system deployment.
- ✓ Compliance:
  There is a set of principles issued by Microsoft around the uses of AI in the open world to ensure the accountability, fairness, transparency, and ethics.
    - Teams at Microsoft were asked to align their engineering practices with that of behaviour of software and services.
    - Transparency aid and privacy protection are the responsible AI practices for model deployment.
    - Models are audited and compliance is checked to ensure the ethical standards are met.

These are some of the best practices with ML in software engineering.

PART-2 Applications of Fuzzy Technology

Prepare a report and explain detailed information of how Fuzzy technology works in the target domain/ application and discuss the flow of the system using fuzzy technology.

In the report, you need to cover the following details: 1. What is the target domain and application? 2. You need to explain how the fuzzy logic can fit into the target application. You need to discuss the advantage of using a fuzzy system in this domain/ application. 3. You need to provide the flow or pipeline of the system using fuzzy technology. 4. You need to provide the rules that can be extracted from the fuzzy system in the domain / Application.

Solution:

There is so much of uncertainty present in the data and systems that can be dealt with only fuzzy technology. To mathematically model on a vague data, this fuzzy logic is used. When a process changes continuously that cannot be defined either as True or False but there is a need for defining those activities in a Fuzzy manner.

The targeted domain is Automobiles and aircrafts. The application is building an Anti-lock Brake Systems (ABS) based on the fuzzy logic.

Anti-Lock Brake Systems:

ABS is a system that is used on aircrafts and automobiles like cars, trucks, buses, and motorcycle to prevent the skidding of the vehicles. ABS prevents the wheels of the automobiles from locking up when applied brake. Threshold and cadence braking principles are the base principles of ABS that were practiced by the skilful drivers long back.

Working: There are four components in the ABS, they are:

Wheel speed sensors, valves, a pump, and a controller.

- Wheel speed sensors: These speed sensors are used to determine the acceleration of a wheel. These sensors generate a signal using the magnet and electromagnetic coil. When there is a fluctuation generated in magnetic field, a voltage is sensed in the sensor, and it gives signal to controllers.
- Valve: There is a valve present in each of the brakes that are controlled by ABS.
- Pump: This is used to store the hydraulic pressure released by the valve.
- Controller: Based on the signals from the speed sensors, the controller will reduce the brake force and activate the ABS mode.

The ABS is a control unit where it monitors the rotational speed of the wheels. If the wheel is detected to be significantly slower than the speed of the vehicle, then braking force and hydraulic pressure is reduced and the wheel is made to run faster. If the wheel is detected to run faster, then braking force and hydraulic pressure is increased and the vehicle is made to slow down. The wheels are in coherence with the ABS, then it's impossible to lock even in sudden brake conditions.

Fitting Fuzzy logic into ABS:

Fuzzy logic is used in ABS to improve the performance in different driving conditions.

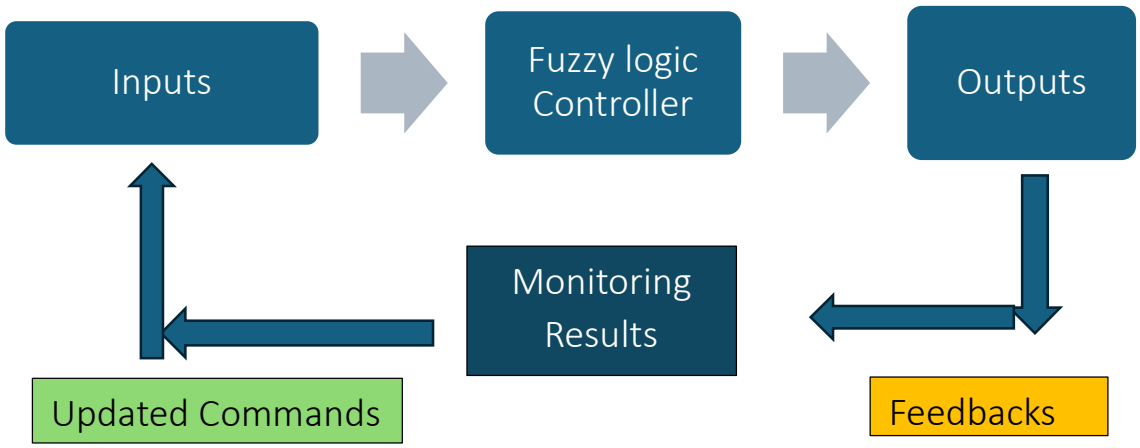
- Inputs from Sensors: Other than the strict thresholds, then real time inputs like wheel speed, vehicle speed, road conditions, wheel slip, and driver's attitude can be considered to develop a proper strategy to adjust the brake pressure.
- Membership Function: Defining this function based on the input variables is the first step in fuzzy logic system. The input value can be termed in linguistic ways such as low, medium, or high.

- Fuzzy Rule Base: Based on the inputs, this base is used to translate the guidelines and knowledge to the fuzzy logic system. This rule base map the inputs to control the output. Here "if-then" statements are used for setting the rules.
- Inference Engine: This engine processes the rules to find the exact control actions based on the inputs. Fuzzy logic operators like AND / OR operators are used for combining the fuzzy rules.
- Defuzzification: The signal from Inference Engine must be converted into commands for the ABS actuators. One common method is weighted average defuzzification.
- Control Actuators: The ABS systems finally use the control signal to adjust the brake pressure at each wheel to prevent wheel lock up by using the valves.

Advantages:

1. As multiple input variables are considered therefore the fuzzy logic system can make better decisions.
2. This approach can optimise the performance of the brakes for over different conditions.
3. The variables have non-linear relationships with each other. Fuzzy logic can handle such non-linearities using the fuzzy rules so that ABS system responds properly.
4. There is high imprecision in the input variables. Those uncertainties are also handled by fuzzy logic.

Fuzzy Logic Pipeline:



Inputs: Inputs like wheel speed, vehicle speed, road conditions, and driver's attitude are to be taken for constructing the fuzzy logic system in the ABS.

Fuzzy logic controller: This controller performs the actions mentioned previously in the rule base, inference engine, and defuzzification steps in order to take effective decisions.

Output Actuators: The ABS systems finally use the control signal to adjust the brake pressure at each wheel to prevent wheel lock up by using the valves.

Feedback is taken from the outputs and monitored and then given as updated commands to the inputs section.

Rules for the Fuzzy logic system:

1. If **wheel slip** is high AND **rate of change of wheel slip** is increasing, THEN brake pressure is decreased.
2. If **wheel slip** is moderate AND **rate of change of wheel slip** is decreasing, THEN brake pressure is maintained.
3. If **wheel slip** is low AND **rate of change of wheel slip** is decreasing, THEN brake pressure is increased.

4. If wheel speed is high AND rate of change of wheel speed is decreasing, THEN brake pressure is decreased.
5. If wheel speed is low AND wheel speed is increasing, THEN brake pressure is increased.
6. If road surface is wet AND rate of change of wheel slip is increasing, THEN brake pressure is decreased abruptly.

REFERENCE

Timothy J. Ross. Fuzzy Logic with Engineering Applications (2010).

Anti-lock Braking Systems, Wikipedia
(https://en.wikipedia.org/wiki/Anti-lock_braking_system)