TASK 2P

STUDENT NAME: M. Rachel

STUDENT ID: 224234147

STUDENT MAIL ID: rachelriney1608@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.
    - Firstly, machine learning is all about data. The amount of effort and rigor it takes to discover, source, manage, and version data is inherently more complex and different than doing the same with software code.
    - Secondly, building for customizability and extensibility of models require teams to not only have software engineering skills but almost always require deep enough knowledge of machine learning to build, evaluate, and tune models from scratch.
    - Thirdly, it can be more difficult to maintain strict module boundaries between machine learning components than for software engineering modules.

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

Model Requirement

Data Collection

Data Cleaning

Data Labeling

Feature Engineering

Model Training

Model Evaluation

Model Deployment

Model Monitoring

Model Requirement:

In the model requirements, designers decide which features are feasible to implement with machine learning and which can be useful for a given existing product or for a new one. Importantly, they also decide what types of models are most appropriate for the given problem.

Data Collection:

In Microsoft the software Teams look for and available datasets of internal or open source or collect their own. they might train a partial model using available generic datasets, and then use transfer learning together with more specialized data to train a more specific model.

Data Cleaning:

It involves removing inaccurate or noisy records from the dataset, a common activity to all forms of data science.

Data Labeling:

Most of the supervised learning techniques require labels to be able to induce a model. Other techniques use demonstration data or environment rewards to adjust their policies. It also assigns ground truth labels to each record.

Feature Engineering:

It refers to all activities that are performed to extract and select informative features for machine learning models. For some models like deep learning models this stage is less explicit and often blended.

Model Training:

For Model Training the chosen models using the selected features that are trained and tuned on the clean, collected data and their respective labels.

Model Evaluation:

The Software engineers evaluate the output model on tested using pre-defined metrics. For critical domains, this stage might also involve extensive human evaluation.

Model Deployment:

The code of the model is then deployed on the targeted devices or cloud servers.

Model Monitoring:

It is continuously monitored for possible errors during real-world execution by user feedback upgrading with best models .

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

Domains:

AI is used in areas such as search, advertising, machine translation, predicting customer purchases, voice recognition, and image recognition, but also saw it being used in novel areas, such as identifying customer leads, providing design advice for presentations and word processing documents, providing unique drawing features, healthcare, and improving gameplay.

Approaches:

ML approaches tobuild their applications, from classification, clustering, dy-namic programming, and statistics, to user behavior modeling,social networking analysis, and collaborative filtering.

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

End-to-end pipeline support:

It is important to develop a data pipeline, capable of continuously loading and massaging data, enabling engineers to try out many permutations of AI algorithms with different hyper-parameters without hassle. The pipelines created by these teams are automated, supporting training, deployment, and integration of models with the product they are a part of. In addition, some pipeline engineers indicated that “rich dashboards” showing the value provided to users are useful. due to the experimental and even more iterative nature of ML development, unifying and automating the day-to-day workflow of software engineers reduces overhead and facilitate progress in the field.

Data availability, collection, cleaning, and management:

The many machine learning techniques are centered around learning from large datasets, projects often heavily depends on data availability, quality and management. Labeling datasets is costly and time-consuming, so it is important to make them available for use within the company.

It necessary to blend data management tools with their ML frameworks to avoid the fragmentation of data and model management activities. A fundamental aspect of data management for machine learning is the rapid evolution of data sources.

Education and Training:

The Software engineers with traditional software engineering backgrounds need to learn how to work along side of the ML specialists.

- First, the company hosts a twice-yearly internal conference on machine learning and data science with at least one day devoted to introductions to the basics of technologies, algorithms, and best practices.
- Second,a number of Microsoft teams host weekly open forums onmachine learning and deep learning, enabling practitioners toget together and learn more about AI.
- Finally, mailing lists andonline forums with thousands of participants enable anyoneto ask and answer technical and pragmatic questions aboutAI and machine learning, as well as frequently share recentresults from academic conferences.

Model Debugging and Interpretability:

Debugging activities for components that learn from datanot only focus on programming bugs, but also focus oninherent issues that arise from model errors and uncertainty to use more interpretable models, or to develop visualization techniques that make the models more interpretable.

Model Evolution, Evaluation and Deployment:

ML based software goes through frequent revisions initiated by model changes, parameter tuning, and data updates, the combination of which has a significant impact on system performance. A number of teams have found it important to employ rigorous and agile techniques to evaluate their experiments.

Fast-paced model iterations require more frequent deployment. To ensure that system deployment goes smoothly, several engineers recommend not only to automate the training and deployment pipeline, but also to integrate model building with the rest of the software, use common versioning repositories for both ML and non-ML codebases

Compliance:

These Compliance include fairness, accountability, transparency, and ethics. All teams at Microsoft have been asked to align their engineering practices and the behaviors of fielded software and services in accordance with these principles. Respect for them is a high priority in software engineering and AI and ML processes and practices.

Varied Perceptions:

The team grouped the respondents into three buckets (low, medium, and high), evenly divided by the number of years of experience respondents personally had with AI.

First, we each of the card sorted categories of respondents challenges divided by the AI experience buckets. Two things are worth noticing. First, across the board, Data Availability, Collection, Cleaning, and Management, is ranked as the top challenge by many respondents, no matter their experience level

Second, some of the challenges rise or fall in importance as the respondents' experience with AI differs.

Challenges around tooling, scale, and model evolution, evaluation, and deployment are more important for engineers with a lot of experience with AI.

PART-2: APPLICATION OF FUZZY LOGIC

Smart Energy Management Systems

1. What is the target domain and application?

The target domain of this application is the implementation of fuzzy logic in smart energy management systems. This system is utilized in various contexts, including smart grids, renewable energy integration, and energy-efficient buildings, to optimize energy consumption, production, and distribution.

2. 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.

Fuzzy logic is well-suited for smart energy management systems due to its ability to handle uncertainty and variability in energy supply and demand. By incorporating fuzzy logic into the control algorithms, these systems can make intelligent decisions based on fuzzy rules and real-time sensor data, leading to more efficient and sustainable energy usage.

Advantages of using a fuzzy system in this domain/application include:

- **Robustness to Uncertainty:** Fuzzy logic enables smart energy management systems to adaptively respond to fluctuations in energy supply and changes in demand, ensuring reliable and stable operation.
- **Adaptability:** Fuzzy logic-based control systems can adjust energy production, storage, and consumption based on real-time inputs from sensors, weather forecasts, and user preferences, and cost savings.
- **Optimization:** By fuzzy rules that prioritize energy conservation, load balancing. Fuzzy logic helps optimize the use of available energy resources and minimize waste.

3. Flow or Pipeline of the Smart Energy Management System:

- **Sensor Inputs:** The smart energy management system collects data from various sensors, including energy meters, weather stations to monitor energy consumption, production, and environmental conditions.

- **Fuzzification:**The raw sensor data is fuzzified into linguistic variables using fuzzy membership functions. For ex: energy consumption readings might be fuzzified into linguistic terms such as "low," "medium," and "high," with corresponding membership functions defining the degree of membership to each term.

- **Fuzzy Inference:** The fuzzified sensor inputs are processed by a Fuzzy Inference System (FIS), which consists of linguistic variables, fuzzy rules, and an inference engine. The fuzzy rules capture expert knowledge or heuristics about energy management strategies, such as demand response, load shifting, and energy storage. The inference engine combines these rules to generate control actions based on the current state of the energy system.

- **Defuzzification:**The control actions determined by the fuzzy inference engine are converted back into action commands. This involves aggregating the fuzzy outputs and selecting the most appropriate control action for energy production, storage, and consumption devices.

- **Control Actions:** Finally, the smart energy management system implements the determined control actions to optimize energy usage and maintain grid stability. This includes adjusting the operation of renewable energy sources, energy storage systems and appliances to align with energy demand and availability.

4. Rules Extracted from the Fuzzy System:

The rules extracted from the fuzzy system in the domain of smart energy management are based on expert knowledge and heuristics related to energy conservation and user preferences. These rules map fuzzy input variables, such as energy demand, renewable energy generation, and environmental conditions, to fuzzy output variables representing control actions, such as adjusting power output, charging/discharging batteries, and scheduling appliance operation.

Example fuzzy rules might include:

- If energy demand is high and renewable energy generation is low, then prioritize energy storage and demand response strategies to reduce grid reliance.
- If environmental conditions indicate high solar irradiance and low energy demand, then increase solar power output and store excess energy in batteries for later use.

These rules enable the smart energy management system to make intelligent decisions in real-time, optimizing energy usage and promoting sustainability.