AI FOR SOFTWARE ENGINEERING: ASSGNMENT 4

1. A human being can take weeks or even months to successfully write a code. Then days to debug it, test it and finally to deploy it. Using AI-driven code generation tools takes minutes or even seconds to generate a code, debug it and test it. This causes developers t save a lot of time when creating applications and websites.
2. Supervised leaning-is trained on labeled data-sets.this tool uses training to identify patterns. It also learns historical data and lastly it can predict if a code has bugs. It's advantages includes but are not limited to high accuracy, easier to interpret, can identify known bug patterns and lastly is easier to validate.

Limitations

Works well with historical data meaning it struggles with new bug types and it requires large labeled data-sets. These can e too time consuming and need a lot of resources.

Unsupervised learning

Unsupervised learning -is learning that finds patterns in data without labells. Usually it learns what normal code looks likem next it will flog anomalities. Unlike supervised learning unsupervised learning can discover new and previously unknown bugs. There is no need for labeled data and it can be used in large projects which makes it useful in early stages.

Limitations

It is harder t explain, it has lower precision compared to supervised learning and can produce false positives-this means it can be inaccurate as not all anomalies are real bugs.

3. Firstly let us identify what a bias is this refers to when an AI system favors certain groups, which is unfair and usually is based an attribute like age, language, race, etc. In AI we have what we call user experience personalization and in this aspect bias can ads to certain groups of people , therefore hindering other groups to see these adds or show different adds to different people for the same item. Content recommendation-this refers to showing certain jobs to males only. Lastly interface design adaptatives -this refers to prioritizing certain users.

Bias mitigation is critical because it ensures fairness and inclusivity. It is important to know that AI should enhance experiences for each and everyone of the users. Hence bias mitigation. It improves model accuracy and ensures compiling with ethical standards.

Section 2
Case study analysis.

In the past software developers had to write code manually, test the code manually, monitor systems around the clock and ensure they react to security threats on time (before any damage).

The introduction of AIOPS improved software deployment efficiency in several ways; let’s discuss:
Intelligent Automation and AI Code suggestion.
In this instance AI suggest possible suitable code whenever a developer is coding.

For example: when creating an HTML code on VS Code, AI assistant suggest possible code a human can use. However this is not a human replacement but an assistance that causes time efficiency and accuracy.

Predictive cost optimization
This means that AI Is able to predict the cost associated with software development. This may improve planning and deployment. Therefore saving cost while optimizing it.

For example: Amazon deploys software using AWS cloud and CI pipeline and it predicts an extra usage of R15000 which can be saved as it results fro test environments that remained active overnight. Amazon remedies this situation and saves R15000.