AI Assignment Report

Part 1: Short Answer Questions (30 points)

1. Problem Definition (6 points)

Hypothetical AI Problem: Predicting job application success rates based on resume and job post data.

Objectives:

- Improve job-matching accuracy
- Reduce time spent applying to jobs
- Increase applicant interview success rates

Stakeholders:

- Job seekers
- Recruiting platforms

KPI:

- Job interview conversion rate (percentage of applicants who reach interview stage)

2. Data Collection & Preprocessing (8 points)

Data Sources:

- Job application portals (e.g., Indeed)
- Resume parsing tools

Potential Bias:

- Representation bias due to over-representation of elite institutions or specific regions

Preprocessing Steps:

- Handle missing data (e.g., blank experience entries)
- Normalize job titles (e.g., "Dev" vs "Developer")
- Encode categorical features (e.g., skills, degree)

AI Assignment Report

3. Model Development (8 points)

Chosen Model: Random Forest

Justification: Handles both numerical and categorical data, less prone to overfitting, interpretable.

Data Split Strategy:

- Training: 70%
- Validation: 15%
- Test: 15%

Hyperparameters to Tune:

- max_depth: Controls model complexity
- n_estimators: Number of trees in the forest

4. Evaluation & Deployment (8 points)

Evaluation Metrics:

- Precision: Measures accuracy of positive predictions (important when false positives are costly)
- ROC-AUC: Evaluates classification ability across all thresholds

Concept Drift:

- Data distribution may change over time (e.g., job market trends)
- Monitoring: Use drift detection tools to track changes in input distribution

Technical Challenge:

- Scalability: Processing large-scale job data in real time

🚑 Part 2: Case Study – Hospital Readmission Prediction (40 points)

---

1. Problem Scope (5 points)

Problem:
Hospitals aim to predict whether a patient will be readmitted within 30 days of discharge. This enables proactive interventions, improves care quality, and reduces operational costs.

Objectives:

1. Predict patient readmission risk using AI models
2. Support discharge planning decisions for clinicians
3. Reduce unnecessary hospital readmissions

Stakeholders:

- Doctors & hospital staff
- Patients and their families
- Hospital administrators & insurance providers

---

2. Data Strategy (10 points)

Data Sources:

- **Electronic Health Records (EHRs):** Medical history, vitals, medications, discharge summaries
- **Demographic Data:** Age, gender, location, lifestyle, insurance info

Ethical Concerns:

1. **Patient Privacy:** Mishandling of sensitive medical data could violate patient trust and regulations like HIPAA
2. **Algorithmic Bias:** Models may unfairly predict higher risk for marginalized groups due to historical bias in data

Preprocessing Pipeline:

Step	Description
1. **Data Cleaning**	Handle missing values in records (e.g., BMI, diagnosis)
2. **Feature Engineering**	Derive features like “# of past admissions”, “comorbidity score”, “length of stay”
3. **Encoding & Scaling**	One-hot encode categorical variables (e.g., gender), normalize numerical features like age or lab results

---

3. Model Development (10 points)

Chosen Model:
🧠 **Random Forest Classifier**

Why:

- Works well with tabular EHR data
- Handles non-linear relationships
- Provides feature importance insights (great for healthcare explainability)

Hypothetical Confusion Matrix (Test Set, 100 patients):

	Predicted: No Readmission	Predicted: Readmission
Actual: No	60	10
Actual: Yes	5	25

Metrics:

- **Precision:** 25 / (25 + 10) = **0.714**
- **Recall:** 25 / (25 + 5) = **0.833**

🎯 High recall ensures we catch most at-risk patients. Precision ensures we don't flood doctors with false alarms.

---

4. Deployment (10 points)

Deployment Steps:

1. Integrate with hospital's EHR system via secure APIs
2. Host model on an internal cloud or local server
3. Display readmission risk scores in doctors' dashboards
4. Trigger alerts when high-risk patients are being discharged
5. Log predictions for auditing and retraining

Compliance Strategy (e.g., HIPAA):

- Encrypt all patient data (at rest and in transit)
- Use access control + audit logs
- Get patient consent for data usage if required

---

5. Optimization (5 points)

Preventing Overfitting:
✅ **Cross-validation + Feature Selection**

- Use k-fold cross-validation to generalize performance
- Drop irrelevant features (like ID fields or duplicated metrics)
- Apply regularization if switching to logistic regression or neural nets

🧠 Part 3: Critical Thinking (20 points)

1. Ethics & Bias (10 points)

How might biased training data affect patient outcomes?
If the training data overrepresents certain demographics (e.g., older patients, specific races, or insured individuals), the model could learn patterns that favor those groups while underperforming for others. This can result in:

- Underdiagnosis or overdiagnosis for underrepresented groups
- Unfair resource allocation
- Worsening of existing healthcare inequalities

1 Strategy to Mitigate Bias:
☑ **Balanced Sampling & Fairness Audits**

- Use techniques like SMOTE or stratified sampling to balance the dataset across sensitive features (e.g., race, gender, insurance status)
- Conduct fairness audits by comparing model performance across subgroups (e.g., AUC for males vs females)
- Regularly retrain the model with updated, diversified data

---

2. Trade-offs (10 points)

Interpretability vs Accuracy in Healthcare

- **High Accuracy Models** (e.g., deep neural nets): Better predictive power but hard to explain → risky in healthcare where **transparency** is critical.
- **Interpretable Models** (e.g., decision trees, logistic regression): Easier for doctors to trust but may lack precision.

🧠 *In healthcare, interpretability often wins.* A slightly less accurate model that can be explained is more ethical and safer.

What if the hospital has limited computational resources?

- Avoid heavy models like large deep learning architectures
- Use lightweight models like **Logistic Regression, Decision Trees, or Random Forests** with fewer trees
- Optimize features beforehand (dimensionality reduction, feature selection)
- Deploy using efficient frameworks like **ONNX** or **TensorFlow Lite**

🔁 Part 4: Reflection & Workflow Diagram (10 points)

1. Reflection (5 points)

Most Challenging Part:
📌 *"Data Strategy & Preprocessing"* — It required aligning technical accuracy with ethical and real-world constraints. Balancing the need for clean, structured data while protecting patient privacy was a tough line to walk.

How I Would Improve With More Time/Resources:

- Collect a more diverse dataset from multiple hospitals
- Collaborate with domain experts (doctors, nurses) for better feature engineering
- Explore fairness-aware machine learning libraries like `Fairlearn`
- Deploy a prototype and collect real user feedback to iterate

---

2. Workflow Diagram (5 points)

Start ↓ Problem Definition ↓ Data Collection ↓ Data Preprocessing ↓ Model Selection ↓ Training & Validation ↓ Model Evaluation ↓ Deployment ↓ Monitoring & Optimization ↓ End

✅ This pipeline ensures a systematic, ethical, and reliable approach to building an AI model for healthcare.