ИТОГОВАЯ АТТЕСТАЦИЯ

Курс: "Архитектор в области искусственного интеллекта"

ОБЩАЯ ИНФОРМАЦИЯ

Форма работы: Индивидуальная
Формат сдачи:

- Репозиторий GitHub с кодом проекта
- PDF-отчет (5-15 страниц)
- Презентация (10-15 слайдов)

Система оценивания: Балльная (100 баллов максимум)
Минимальный проходной балл: 60 баллов

ЦЕЛЬ АТТЕСТАЦИИ

Продемонстрировать комплексное владение навыками архитектора ИИ через создание законченного ML-проекта, включающего все этапы: от сбора требований и проектирования архитектуры данных до разработки, тестирования и развертывания модели машинного обучения.

СТРУКТУРА ЗАДАНИЯ

Задание состоит из обязательных и дополнительных блоков. Каждый блок оценивается отдельно.

ОБЯЗАТЕЛЬНЫЕ БЛОКИ (60 баллов - минимум для зачета)

- Блок 1: Анализ и проектирование (15 баллов)
- Блок 2: Работа с данными (20 баллов)
- Блок 3: Машинное обучение (25 баллов)

ДОПОЛНИТЕЛЬНЫЕ БЛОКИ (40 баллов)

- Блок 4: Архитектура данных (15 баллов)
- Блок 5: Инфраструктура и развертывание (15 баллов)
- Блок 6: Качество и документация (10 баллов)

ВЫБОР ТЕМЫ ПРОЕКТА

Вам необходимо выбрать ОДНУ из трех предметных областей и решить соответствующую задачу машинного обучения:

Вариант А: E-commerce (Интернет-магазин)

Задача: Прогнозирование оттока клиентов (Customer Churn Prediction)

Описание: Построить систему, которая предсказывает вероятность того, что клиент перестанет совершать покупки в течение следующих 3 месяцев.

Датасет: E-commerce dataset или аналогичный

Бизнес-требования:

- Precision не менее 0.70 (важно не беспокоить лояльных клиентов)
- Recall не менее 0.65 (важно выявить большинство потенциальных оттоков)
- Модель должна объяснять свои предсказания (интерпретируемость)
- Время инференса: < 100ms на одного клиента

Вариант Б: Производство (Manufacturing)

Задача: Предиктивное обслуживание оборудования (Predictive Maintenance)

Описание: Построить систему, которая предсказывает вероятность выхода оборудования из строя в ближайшие 7 дней на основе данных с датчиков.

Датасет: NASA Turbofan Engine Degradation или Predictive Maintenance Dataset

Бизнес-требования:

- Recall не менее 0.85 (критично не пропустить поломку)
- False Positive Rate < 0.15 (минимизация ложных тревог)
- Модель должна работать с данными в режиме реального времени
- Время инференса: < 50ms

Вариант В: Банковская сфера (Banking)

Задача: Скоринг кредитоспособности (Credit Scoring)

Описание: Построить систему оценки кредитоспособности клиентов для принятия решения о выдаче кредита.

Датасет: Give Me Some Credit или Home Credit Default Risk

Бизнес-требования:

- AUC-ROC не менее 0.75
- Модель должна быть интерпретируемой (регуляторные требования)
- Справедливость (fairness) - отсутствие дискриминации по защищенным признакам

- Время инференса: < 200ms

ДЕТАЛЬНОЕ ОПИСАНИЕ БЛОКОВ

БЛОК 1: АНАЛИЗ И ПРОЕКТИРОВАНИЕ (15 баллов)

1.1 Сбор и анализ требований (5 баллов)

Применение знаний из темы 1.5

Формализовать бизнес-задачу, определить метрики успеха (бизнес и ML метрики), описать функциональные и нефункциональные требования.

Критерии оценки:

- 5 баллов: Полная формализация с количественными метриками и обоснованием
- 3 балла: Базовое описание требований без детальной проработки
- 1 балл: Поверхностное описание без конкретики

1.2 Проектирование архитектуры решения (5 баллов)

Применение знаний из тем 4.1, 6.1-6.3

Создать архитектурную диаграмму системы, показывающую: источники данных, хранилища данных, компоненты обработки, ML компоненты, API для inference, мониторинг.

1.3 Планирование экспериментов (5 баллов)

Применение знаний из тем 1.3, 1.4

Определить гипотезы для проверки (минимум 3), спланировать A/B тесты для валидации модели, описать методологию сравнения моделей, определить baseline решение.

БЛОК 2: РАБОТА С ДАННЫМИ (20 баллов)

2.1 Сбор и подготовка данных (10 баллов)

Применение знаний из разделов 2 (SQL), 3 (Python), 4 (Архитектура данных)

Обязательные задачи:

1. Загрузка данных (2 балла): Загрузить датасет, сохранить raw данные в PostgreSQL
2. EDA (4 балла): Анализ распределений, корреляций, выбросов, целевой переменной
3. Очистка данных (4 балла): Обработка пропусков, дубликатов, выбросов, feature engineering

2.2 Организация логических слоев данных (10 баллов)

Применение знаний из тем 4.3, 5.1

Обязательная структура в PostgreSQL:

- Raw Layer - исходные данные (2 балла)
- Cleaned Layer - очищенные данные (3 балла)
- Features Layer - подготовленные признаки для ML (5 баллов)

БЛОК 3: МАШИННОЕ ОБУЧЕНИЕ (25 баллов)

3.1 Baseline модель (5 баллов)

Применение знаний из раздела 7

Реализовать простую baseline модель: Логистическая регрессия / Линейная регрессия / Decision Tree. Train/Test split, оценка на базовых метриках, анализ ошибок.

3.2 Эксперименты с алгоритмами (10 баллов)

Применение знаний из тем 7.4-7.8

Реализовать и сравнить минимум 3 различных алгоритма: Random Forest, Gradient Boosting, SVM, Neural Networks. Гиперпараметрическая оптимизация, кросс-валидация, сравнение моделей, анализ важности признаков.

3.3 Финальная модель и оценка (10 баллов)

Обязательные компоненты:

4. Выбор лучшей модели (3 балла): Обоснование выбора, анализ компромисов
5. Глубокая оценка модели (4 балла): Confusion Matrix, ROC-AUC, Precision-Recall, анализ ошибок
6. Статистические тесты (3 балла): Проверка значимости улучшений, bootstrap, A/B тест симуляция

ДОПОЛНИТЕЛЬНЫЕ БЛОКИ

БЛОК 4: АРХИТЕКТУРА ДАННЫХ (15 баллов)

- 4.1 Feature Store (7 баллов): Offline и Online хранилища признаков
- 4.2 Потоковая обработка (8 баллов): Kafka Producer/Consumer или упрощенная версия

БЛОК 5: ИНФРАСТРУКТУРА И РАЗВЕРТЫВАНИЕ (15 баллов)

- 5.1 REST API (7 баллов): FastAPI с валидацией и тестами
- 5.2 Контейнеризация (8 баллов): Docker и Docker Compose

БЛОК 6: КАЧЕСТВО И ДОКУМЕНТАЦИЯ (10 баллов)

- 6.1 Тестирование (5 баллов): Unit и интеграционные тесты, покрытие > 60%
- 6.2 Документация и Git (5 баллов): README, docstrings, чистый Git

ТАБЛИЦА ОЦЕНИВАНИЯ

Блок	Компонент	Баллы	Обязательный
1. Анализ и проектирование		**15**	Да
	1.1 Сбор требований	5	Да
	1.2 Архитектура решения	5	Да
	1.3 План экспериментов	5	Да
2. Работа с данными		**20**	Да
	2.1 EDA и предобработка	10	Да
	2.2 Логические слои данных	10	Да
3. Машинное обучение		**25**	Да
	3.1 Baseline модель	5	Да
	3.2 Эксперименты	10	Да
	3.3 Финальная модель	10	Да
ИТОГО ОБЯЗАТЕЛЬНЫХ		**60**	
4. Архитектура данных		15	Нет
5. Инфраструктура		15	Нет
6. Качество и документация		10	Нет
МАКСИМУМ		**100**	

ШКАЛА ОЦЕНИВАНИЯ

Баллы	Оценка	Описание
90-100	Отлично (5)	Все обязательные + большинство дополнительных. Высокое качество
75-89	Хорошо (4)	Все обязательные + часть дополнительных. Хорошее качество
60-74	Удовлетворительно (3)	Все обязательные блоки на базовом уровне
< 60	Неудовлетворительно (2)	Обязательные блоки не выполнены полностью

ТРЕБОВАНИЯ К ОТЧЕТУ

Структура отчета (5-15 страниц):

- **1. Введение:** Описание бизнес-задачи, актуальность, цели (0.5-1 стр)
- **2. Требования:** Функциональные и нефункциональные требования, метрики (1-2 стр)
- **3. Архитектура:** Диаграмма системы, описание компонентов, технологии (1-2 стр)
- **4. Анализ данных:** Описание датасета, EDA, проблемы и решения (2-3 стр)
- **5. Методология:** План экспериментов, выбор алгоритмов, метрики (1-2 стр)
- **6. Результаты:** Сравнение моделей, выбор финальной, оценка (2-3 стр)
- **7. Выводы:** Достижения, ограничения, рекомендации (0.5-1 стр)

СТРУКТУРА ПРОЕКТА

Рекомендуемая структура репозитория:

```
project_name/
│
├── README.md
├── requirements.txt
├── .gitignore
├── .env.example
│
├── data/
│   ├── raw/
│   ├── processed/
│   └── samples/
│
├── sql/
│   ├── schema.sql
│   ├── layers.sql
│   └── queries.sql
│
├── notebooks/
│   ├── 01_data_preparation.ipynb
│   ├── 02_baseline_model.ipynb
│   ├── 03_model_experiments.ipynb
│   └── 04_final_model.ipynb
│
├── src/
│   ├── data/
│   ├── models/
│   ├── utils/
│   └── etl_pipeline.py
│
├── api/          # опционально
├── streaming/    # опционально
├── tests/        # опционально
├── deployment/   # опционально
│
└── docs/
    ├── architecture.png
    ├── report.pdf
    └── presentation.pdf
```