RNMP - Homework 3

Stefan Tagarski - 221184

1. Вовед

Оваа домашна задача имплементира комплетен pipeline за предвидување на дијабетес користејќи Apache Spark за машинско учење и Apache Kafka за real-time streaming. Системот е организиран во две главни фази: offline фаза за обука на модели и online фаза за real-time предвидувања врз streaming податоци.

1.1 Цели на проектот

- Обука на машински модели за класификација на дијабетес
- Имплементација на real-time streaming pipeline
- Интеграција на Apache Spark и Apache Kafka
- Оптимизација на performance и scalability

2. Архитектура на системот

Системот следи lambda архитектура со одвоени batch (offline) и streaming (online) слоеви:

2.1 Offline фаза - Обука на модели

Во offline фазата се врши обука на machine learning модели врз историски податоци. Процесот вклучува:

1. Вчитување на offline.csv податоци (80% од dataset)
2. Feature engineering со VectorAssembler
3. Стандардизација на features со StandardScaler
4. Обука на 3 типа модели: Logistic Regression, Random Forest, Decision Tree
5. Cross-validation со 5 folds за секој модел
6. Хиперпараметарска оптимизација со ParamGridBuilder
7. Евалуација со повеќе метрики: F1, Accuracy, Precision, Recall, AUC-ROC
8. Зачувување на најдобриот модел

2.2 Online фаза - Real-time предвидување

Во online фазата се врши real-time предвидување врз streaming податоци:

9. Kafka Producer праќа податоци од online.csv во topic "health_data"
10. Spark Structured Streaming консумира податоци од Kafka
11. Истите трансформации како во offline фазата се применуваат
12. Обучениот модел прави предвидувања
13. Збогатените податоци се праќаат во topic "health_data_predicted"
14. Kafka Consumer ги прима и прикажува резултатите

3. Податочно множество

Користен е Diabetes Health Indicators Dataset од BRFSS 2015, кој содржи 253,680 записи со 21 health-related features.

3.1 Features

Feature	Опис	Тип
Diabetes_binary	Цел променлива (0=Нема, 1=Има)	Binary
HighBP	Висок крвен притисок	Binary
HighChol	Висок холестерол	Binary
BMI	Body Mass Index	Continuous
Smoker	Пушач	Binary
Stroke	Имал инсулт	Binary
HeartDiseaseorAttack	Срцево заболување	Binary
PhysActivity	Физичка активност	Binary
GenHlth	Општо здравје (1-5)	Ordinal
Age	Возрасна категорија	Ordinal
Income	Приходна категорија	Ordinal

3.2 Поделба на податоци

Податоците се поделени во два сета со задржување на класниот баланс (stratified split):

Dataset	Процент	Намена
offline.csv	80%	Обука на модели
online.csv	20%	Real-time streaming тест

4. Имплементација

4.1 Offline Training (offline_optimized.py)

Главни карактеристики на оптимизираната offline скрипта:

- Модуларна структура со посебни функции за секој чекор
- Оптимизирана Spark конфигурација со Adaptive Query Execution
- Паралелна кросвалидација за побрзо извршување
- Comprehensive евалуација со повеќе метрики
- Детално логирање на сите операции
- Автоматско зачувување на најдобриот модел

Конфигурација на модели:

Модел	Хиперпараметри	Grid Size
Logistic Regression	regParam: [0.001, 0.01, 0.1] elasticNetParam: [0.0, 0.5, 1.0]	9 комбинации
Random Forest	numTrees: [20, 50, 100] maxDepth: [5, 10, 15] minInstancesPerNode: [1, 5]	18 комбинации
Decision Tree	maxDepth: [5, 10, 15, 20] minInstancesPerNode: [1, 5, 10] maxBins: [32, 64]	24 комбинации

4.2 Kafka Producer (producer_optimized.py)

Оптимизациите вклучуваат:

- Gzip компресија за намалена network usage
- Batch sending за подобар throughput
- Конфигурабилен delay помеѓу пораки
- Error handling со retry механизам
- Детална статистика за праќање
- Додавање на metadata (timestamp, record_id)

4.3 Spark Streaming (online_stream_optimized.py)

Клучни features:

- Streaming backpressure за автоматска регулација
- Оптимизирани партиции за streaming
- Checkpoint механизам за fault tolerance
- Kryo сериализација за побрза обработка
- Опционален console output за monitoring
- Comprehensive error handling

4.4 Kafka Consumer (consumer_optimized.py)

Подобрувања:

- Форматирано прикажување на резултати
- Real-time статистика за предвидувања
- Автоматско пресметување на класна дистрибуција
- Конфигурабилен statistics interval
- Consumer group management

5. Резултати и евалуација

5.1 Model Performance

Сите модели беа обучени со 5-fold cross-validation. Резултатите се прикажани подолу:

Модел	F1 Score	Accuracy	Precision	Recall	AUC-ROC
Logistic Regression	0.8258	0.8523	0.8456	0.8523	0.8912
Random Forest	0.8264	0.8531	0.8463	0.8531	0.8945
Decision Tree	**0.8319**	**0.8589**	**0.8512**	**0.8589**	**0.8978**

Заклучок: Decision Tree моделот постигна најдобри резултати со F1 score од 0.8319 и беше избран како финален модел за deployment.