Spark ML

2022.10.12

발제자: 오효근, 우아라

Review: Spark

- Features
  - 오픈 소스 클러스터 컴퓨팅 프레임 워크
  - 클러스터 환경 데이터 병렬 처리 라이브러리 집합으로 구성
  - UC Berkely의 AMP Lab.에서 최초 개발
    - Zaharia, Matei, et al. "Spark: Cluster computing with working sets." *2nd USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 10)*. 2010.
  - Big data application 개발에 필요한 통합 플랫폼 제공 지향
  - 데이터 읽기, SQL 처리, Machine learning 등 데이터 분석 작업과 같은 연산 엔진과 일관성 있는 API로 구성

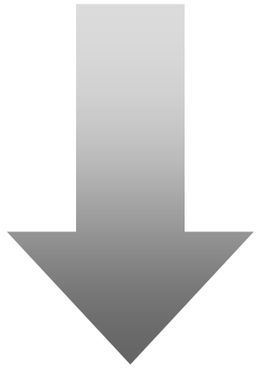
- Benefits
  - S/W license 비용 없이 대용량 데이터 처리를 위한 최적의 솔루션
  - 대용량 DW 영역에서 영역 확장 진행 중
  - 다양한 오픈 소스 에코 시스템 지원
  - 오픈 소스 기반 대량 데이터 처리의 중심

Spark ML

- Spark ML
  - Definition: Spark에서 제공하는 ML library
  - Spark MLlib vs. Spark ML
    - Spark MLlib: RDD (Resilient Distributed Dataset) 기반 ML library
    - Spark ML: DataFrame 기반 ML library

Spark MLlib

- Spark의 초기 버전 ML library
- Spark 2.0 부터 더 이상 사용되지 않음



Spark ML

- Spark의 최신 ML library
- DataFrame과 integration
- Scikit-Learn의 API 등에서 많은 영향

- Machine Learning
  - Representative Models
    - Supervised Learning
      - Decision Tree
      - Random Forest
      - Naïve Bayes
      - Support Vector Machine (SVM)
    - Unsupervised Learning
      - K-Means Clustering
      - DBSCAN
      - Principal Components Analysis (PCA)
      - t-distributed stochastic neighbor embedding (t-SNE)
  - Definition: 컴퓨터가 특정 작업을 수행하기 위해 개발 프로세스 또는 특정 프로그래밍 없이도 지속적 학습 및 데이터 기반으로 목표를 수행하는 기술

Features of Spark ML

- ❑ 일반적인 machine learning algorithm 제공
- ❑ Feature extraction, feature selection, transformation, dimensionality reduction 등 다양한 도구 제공
- ❑ 알고리즘과 모델 및 파이프라인의 저장, 로드, 데이터 처리, 통계학 등 유틸리티
- ❑ GPU를 활용하는 데이터 브릭스의 Spark Cluster 구성 사용 가능
- ❑ Hyperparameter tuning 기능 제공
- ❑ Scalar, Java, Python, R 등 각종 환경의 API 제공

Estimator

- Classification
- Regression
- Clustering
- Collaborative Filtering

Transformer

- StringIndexer
- OneHotEncoder
- VectorAssebler
- StandardScaler
- Tokenizer

Model Selection (Tuning)

- randomSplit
- TrainValidationSplit
- CrossValidator
- ParamGridBuilder

Pipeline

- Pipeline

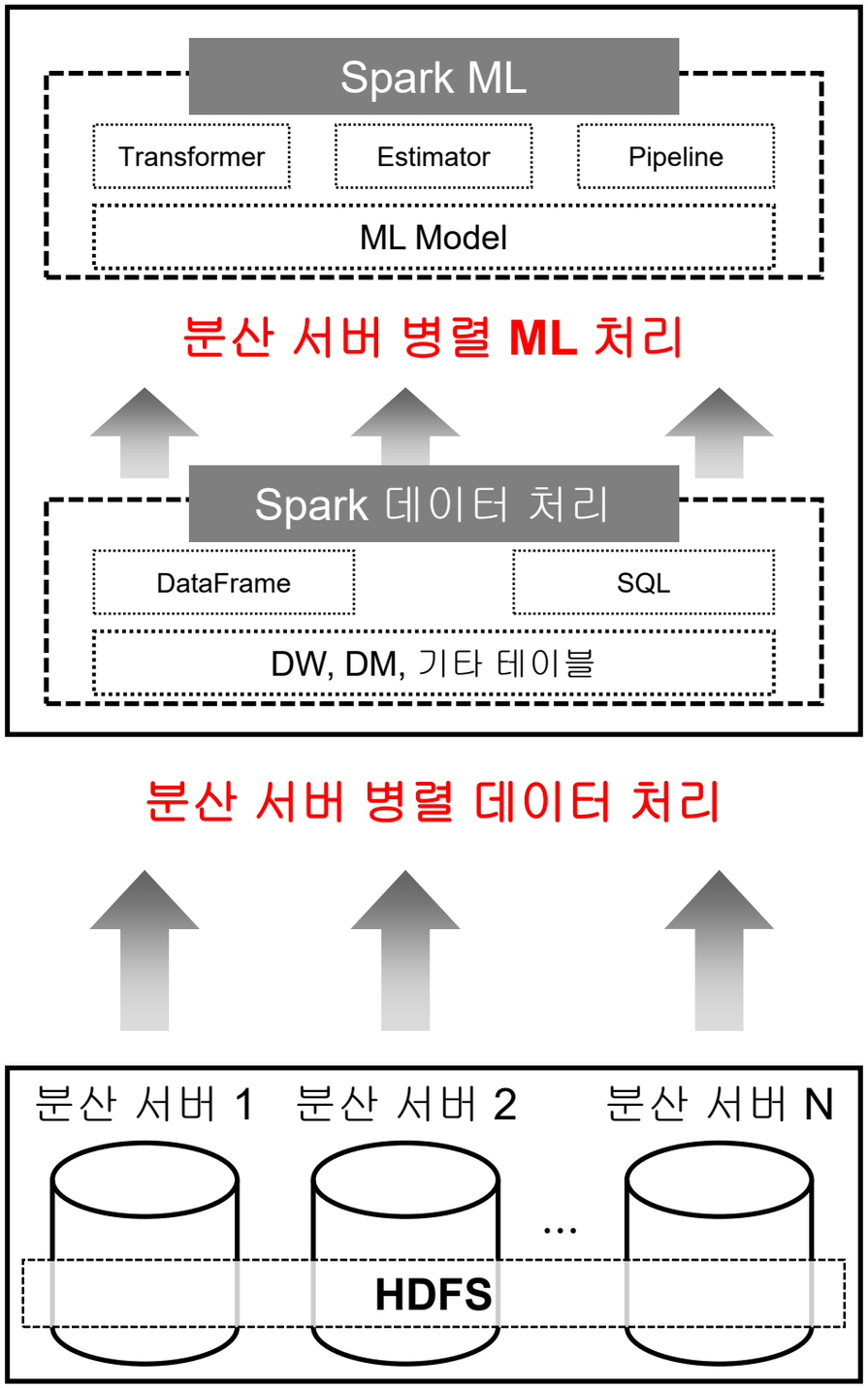
Evaluator

- RegressionEvaluator
- BinaryClassificationEvluator
- MultiClassificationEvaluator

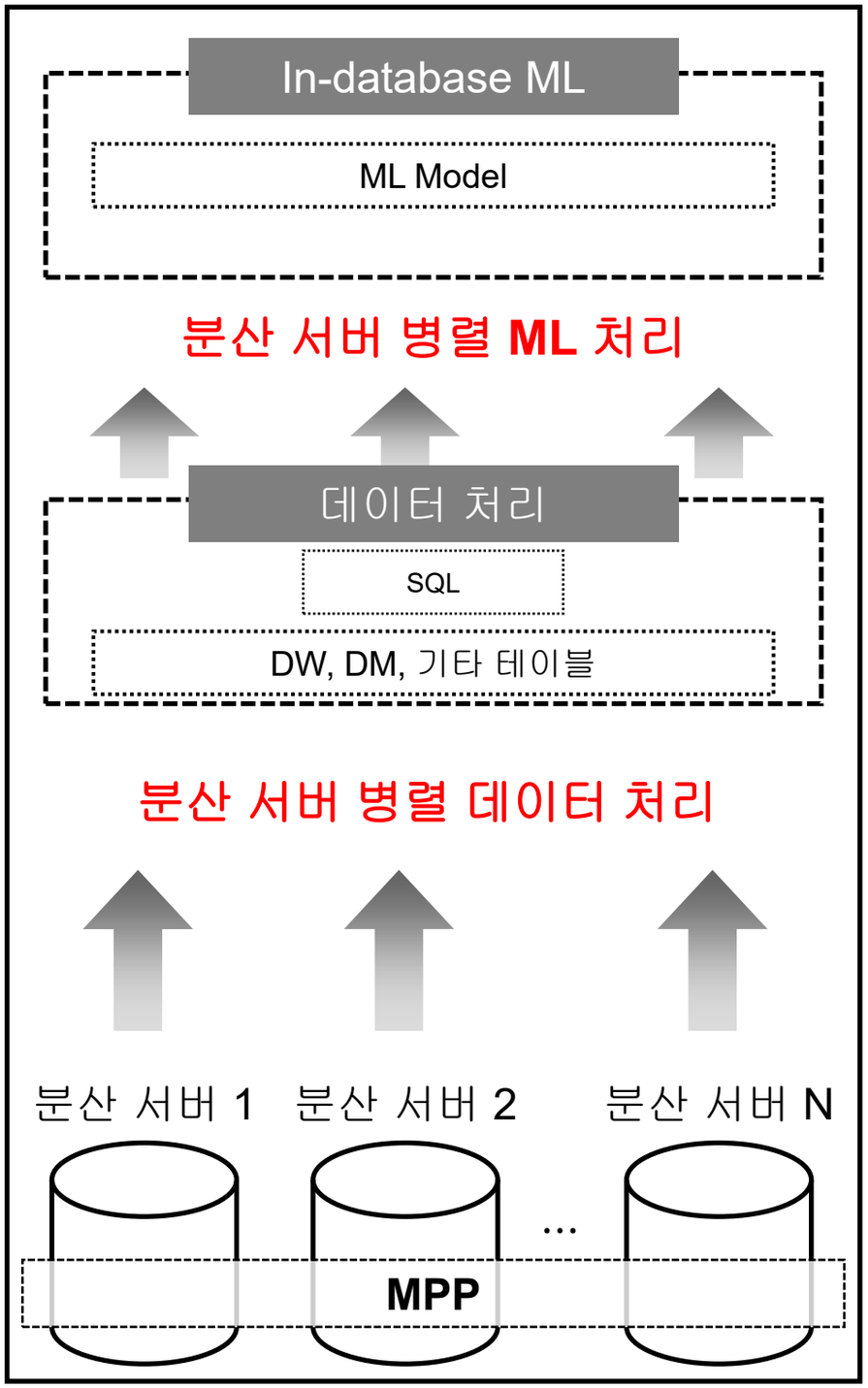
Extractor, Feature Selection

- TF-IDF
- Word2Vec
- ChiSqSelector

Spark ML vs. In-database ML



VS



Machine Learning Process

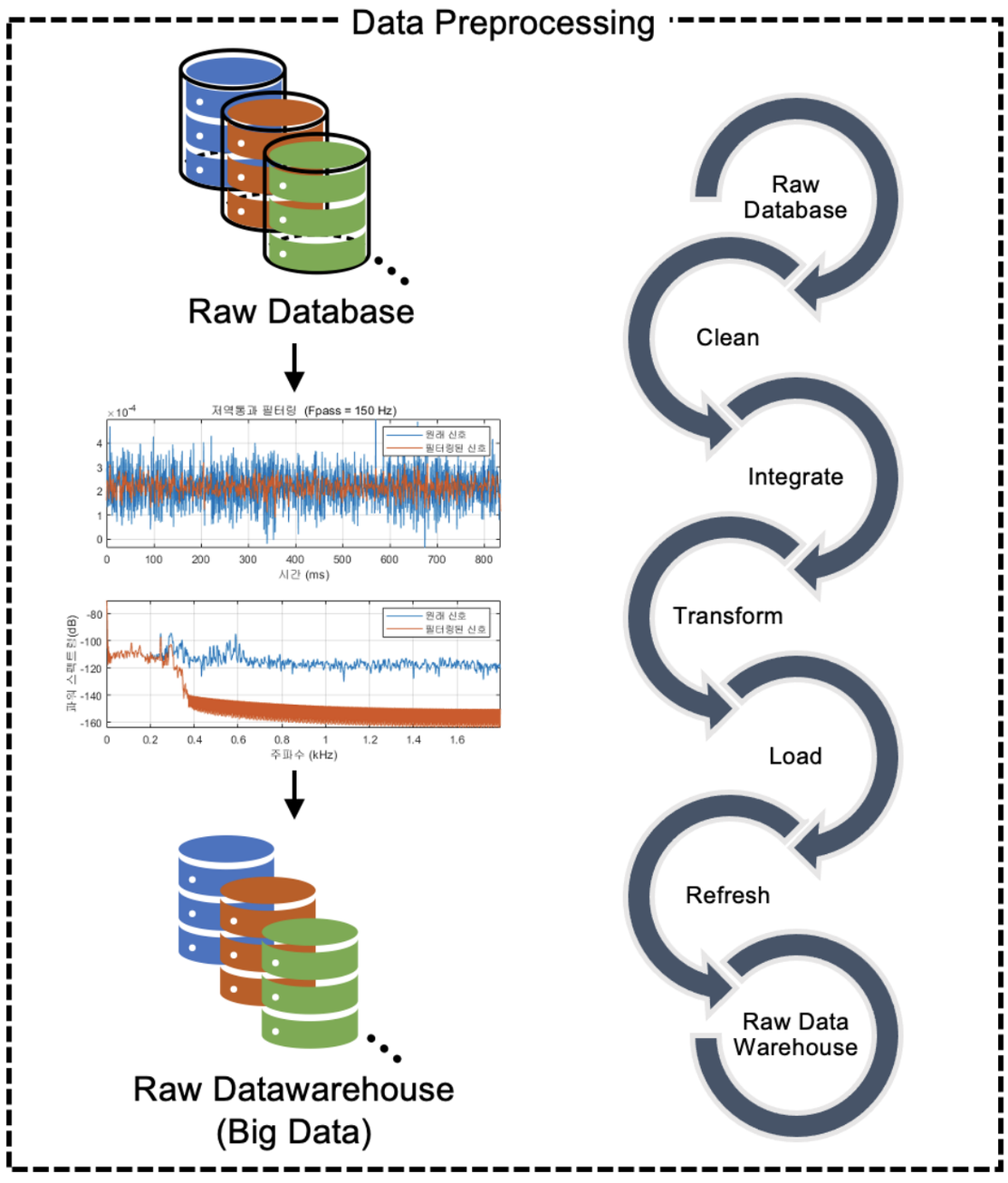
1. 데이터 전처리
   - 데이터 클렌징
   - 결측치 처리: Null / NaN
   - 데이터 인코딩: 레이블, 원-핫 인코딩
   - 데이터 스케일링
   - 이상치 제거
   - Feature 추출 및 선택

2. 데이터 세트 분리
   - 학습 데이터 및 테스트 데이터 분리

3. 모델 학습 및 검증 평가
   - 알고리즘 학습
   - Hyperparameter 튜닝

4. 예측 수행
   - 테스트 데이터로 예측 수행

5. 평가
   - 예측 평가
   - 교차 검증 (cross validation)



Machine Learning Process in Spark ML

1. 데이터 소스를 Spark DataFrame으로 변환
   - iris_sdf = spark.createDataFrame(iris_pdf)

2. DataFrame을 학습 및 테스트 DataFrame으로 분할
   - train_sdf, test_sdf = iris_sdf.randomSplit([0.8, 0.2], seed = 42)

3. 학습 시키려는 여러 개의 feature columns를 하나의 feature vector로 변환
   - iris_columns = [‘sepal_length’, ‘sepal_width’, ‘petal_length’, ‘petal_width’]
   - vec_assembler = VectorAssembler(inputCols = iris_columns, outputCol = ‘features’)
   - train_feature_vector_df = vec_assembler.transform(train_sdf)

4. ML algorithm 객체 생성 후 모델 학습
   - lr = logisticRegression(featureCol = ‘features’, labelCol = ‘target’)
   - lr_model = lr.fit(train_feature_vector_df)

5. 학습 모델을 이용하여 테스트 데이터 예측
   - test_feature_vector_df = vec_assembler.transform(test_sdf)
   - predictions = lr_model.transform(test_feature_vector_df)

시스템 24