

BM 과정 AI Architecture & Business Model

2019

에스튜비즈
대표컨설턴트
이 이 백



목 차

- I. Big Data Architecture와 AI 시스템의 Position
- II. AI 구축 방법론
- III. Industry 4.0
- IV. AI 시스템 구축을 위한 기본 지식과 기술
- V. Deep Learning Revisit
- VI. Business Model과 AI Topic 발굴

II

AI 구축 방법론

1. AI 구축 Process
2. Process Step



1. AI 구축 Process

- AI 시스템의 다른 시스템과의 차이점

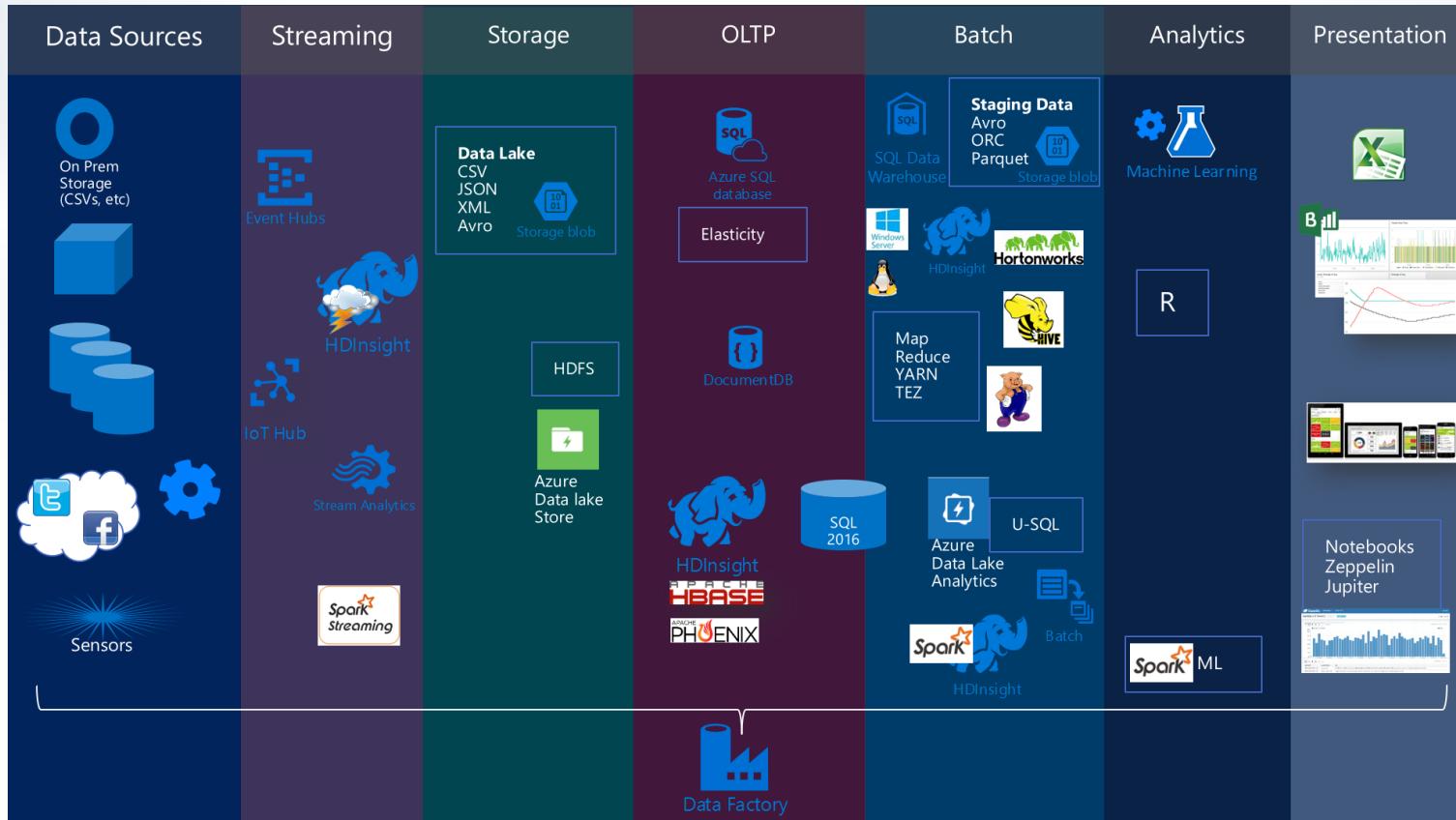
	일반적인 시스템	AI 시스템
정보의 성격	<ul style="list-style-type: none"> 실무자·다수가 요구하는 실무 자체와 이의 요약 컨텐츠 정보의 수요자는 제공 정보의 직관적 이해의 표현 및 정보량의 다양성 요구 	<ul style="list-style-type: none"> 목적의 구체성 혹리스트의 최대한 배제 제공하는 계량 정보에 대한 사전 이해 필요
정보의 생산	<ul style="list-style-type: none"> 요구 정보에 맞게 조건 정제 비즈니스/업무 목적과 철차의 구현 실세계의 비즈니스 흐름에 대해 다수의 공통된 관심을 반영하는 정보 경험 규칙/Rule의 최대 반영 경향 	<ul style="list-style-type: none"> 요구 정보와 직접/간접 연결이 있는 대부분의 Data를 Model Input에 적용 소수의 Supervisor가 수학/컴퓨터공학, 알고리즘으로 구성된 모델의 학습/검증 과정에서 Parameter Tuning을 통해 정보 생산
일반화	<ul style="list-style-type: none"> 경험자와 해당 분야 전문가의 논리적/근거적 제시 	<ul style="list-style-type: none"> 서비스하고자 하는 AI Model의 실세계 검증

[source] 저자의 강의 노트를 BM과정에 맞게 재정리함

1. AI 구축 Process

▪ AI 구축 절차 > Project Life-Cycle 측면

- AI 시스템은 데이터 수집에서 서비스 제공에 이르는 일련의 과정 포함해야 함
- 아래의 그림은 Microsoft의 Cloud 서비스인 Azure의 빅데이터 아키텍처의 제안임.
- AI 구축 절차의 맥락은 이와 유사하지만, AI 생산 정보의 성격에 따라 유연하게 구조화 설계 필요



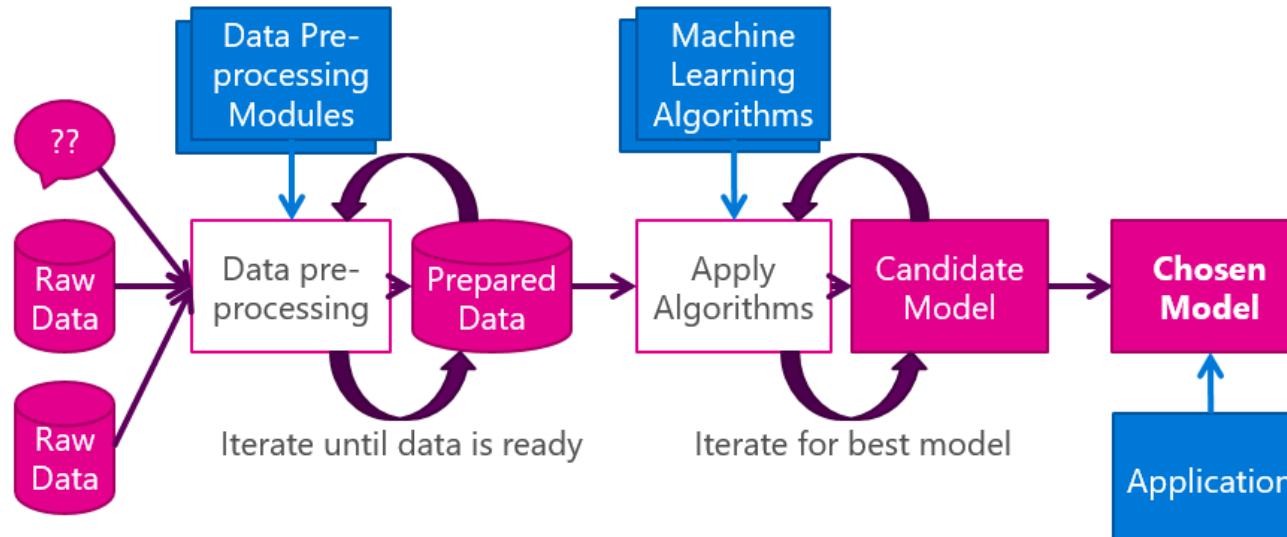
[source] <https://blogs.msdn.microsoft.com/robinlester/2016/03/09/architecting-a-big-data-project-in-azure/>

II. AI 구축 방법론

1. AI 구축 Process

- AI 구축 절차 > Model Application 측면

단계	내용
Data Pre-Processing	<ul style="list-style-type: none"> Data 원천(Source)로 부터 AI시스템으로 Data 획득 Model Input 구조(Label & Feature, Model& Test, Iteration 등)로 Data 전처리
Modeling	<ul style="list-style-type: none"> 다양한 모델(DT, RF, SVM, DNN, CNN, RNN/LSTM, Auto encoder 등) 중 목적에 부합하는 복수개의 기법을 선택 [Modeling ,Validation, Test] 과정의 반복 수행
Application	<ul style="list-style-type: none"> 최종 모델을 서비스하고자 하는 시스템 또는 Device(Sensor Equipment, IoT 등)에 Embedding



[source] <https://www.linkedin.com/pulse/from-data-processing-till-model-selection-machine-payam-mokhtarian>

2. Process Step

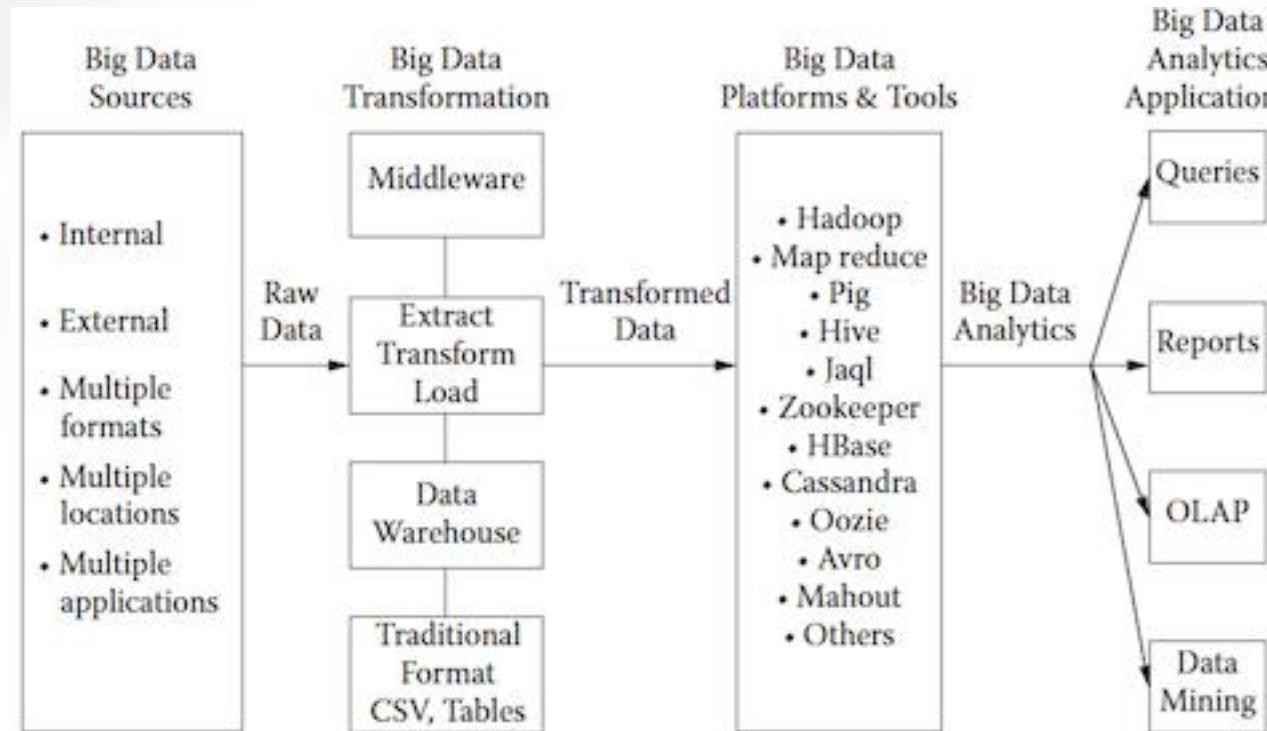
7

- 1) Get Data from Source
- 2) Write Local Data into Hadoop
- 3) Data Processing for Analytic Work
- 4) Analytic Works
- 5) AI Model Save & Restore
- 6) AI Service

2. Process Step

1) Get Data from Source > Data Location 측면

- Data 원천(Source)로 부터 AI시스템으로 Data 획득
- Model Output과 직/간접 관련 모든 데이터를 AI 시스템 Repository로 ETL(Extract, Transformation, Loading)



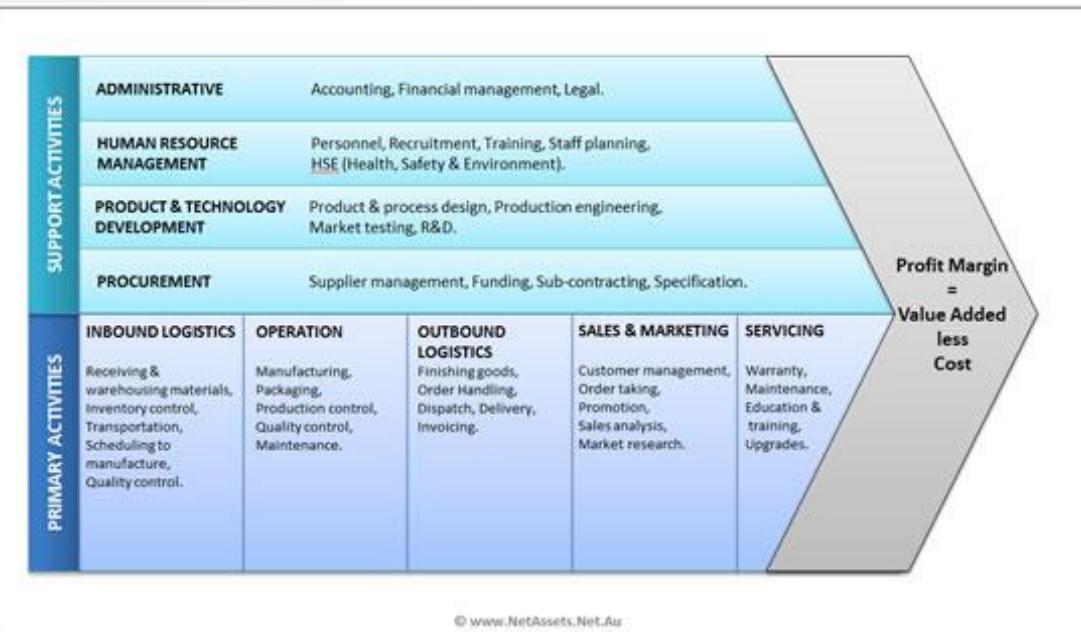
[source] <http://www.ittoday.info/ITPerformanceImprovement/Articles/2014-07Raghupathi.html>

2. Process Step

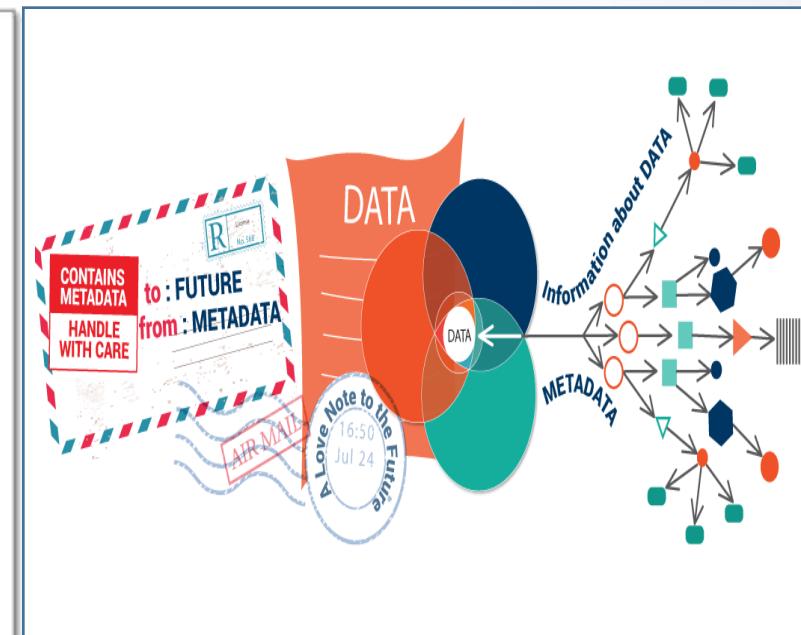
1) Get Data from Source > 내부 Data Contents 측면

- Business Domain의 이해를 바탕으로 Business Value/Process 를 Source Data와 Mapping하고 이의 Meta Data 정의 및 Entity Diagram 작성

Value Chain(Porter, 1985)



Metadata 정의



[source] <http://www.data-group.com.au/business-intelligence-data-sources/>

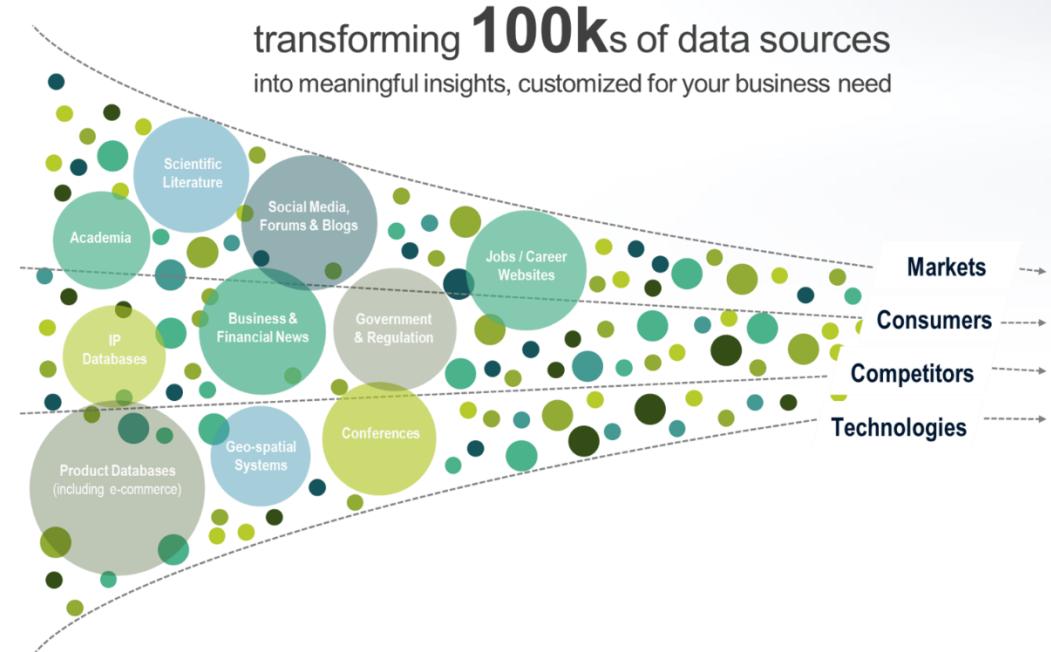
[source] <https://www.ontotext.com/knowledgehub/fundamentals/metadata-fundamental/>

2. Process Step

10

1) Get Data from Source > 외부 데이터 수집 측면

- 조직의 비즈니스와 관련있는 외부데이터의 수집은 정보 가치화의 필수이며 Big Data 수집 기술을 적용해야 능력없이 정확한 시간이 Tag 된 데이터 획득이 가능함
- 데이터의 수집 주기 : 실시간 Streaming, 주기적 Batch



[source] <https://www.smartdatacollective.com/big-data-20-free-big-data-sources-everyone-should-know/>

[source] <https://signals-analytics.com/blog/signals-analytics-approach-to-data-interview-with-asaf-frige-0#>

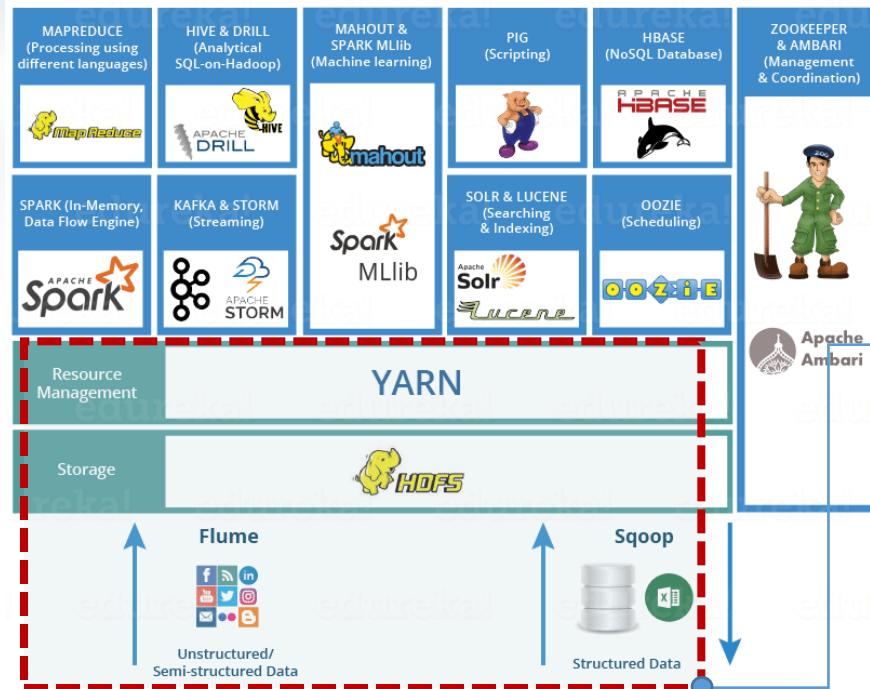
2. Process Step

11

2) Write Local Data into Hadoop

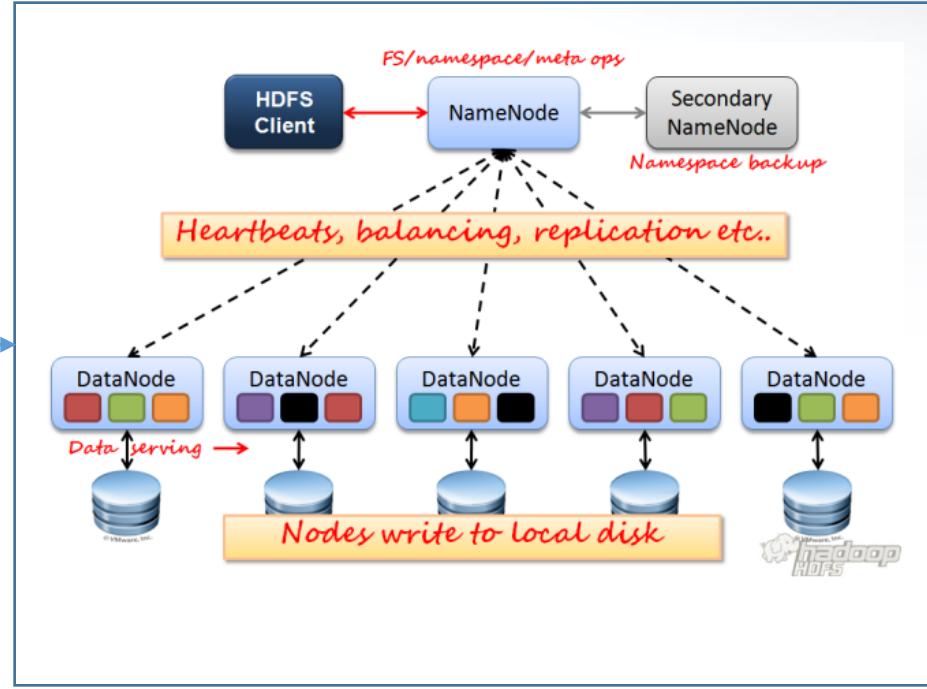
- Local Data : DBMS, Flat File(CSV, TEXT), 시스템 Log 등(외부로 부터 수집하는 데이터 포함)
- AI의 학습용 데이터를 제공할 Input Data의 repository가 반드시 Hadoop일 필요는 없음. 소규모 biz model은 DBMS를 활용해도 무방함. 하지만 대규모의 데이터를 학습해야 하는 경우라면 Hadoop이 이와 동등 수준의 Repository로 구성해야 함.

Hadoop Eco-System



[source] <https://www.edureka.co/blog/hadoop-ecosystem/>

Hadoop Data 저장 구조



[source] <https://yooyoclouds.wordpress.com/tag/hdfs/>

2. Process Step

12

2) Write Local Data into Hadoop

- [실습] AI 구축을 위한 Hadoop의 명령어의 활용
 - AI Model에 활용할 Data의 가장 기본적인 수준의 탐색
 - AI Model Input Data의 잠정적 확장

▣ 자주 활용하는 Hadoop/HDFS Command

Command	설명
hdfs dfs -version	▪ Hadoop Version Check
hdfs dfs -ls	▪ Hadoop File & Directory 조회
hdfs dfs -cat	▪ File 내용을 Console에 display
Hdfs dfs -get	▪ Hadoop File을 Local Directory로 복사
hdfs dfs -put	▪ Local File과 Directory를 Hadoop에 복사
hdfs dfs -copyFromLocal	▪ put과 동일
hdfs dfs -copyToLocal	▪ get과 동일
hdfs dfs -cp	▪ File 복사
hdfs dfs -mv	▪ File 이동

2. Process Step

3) Data Processing for Analytic Work

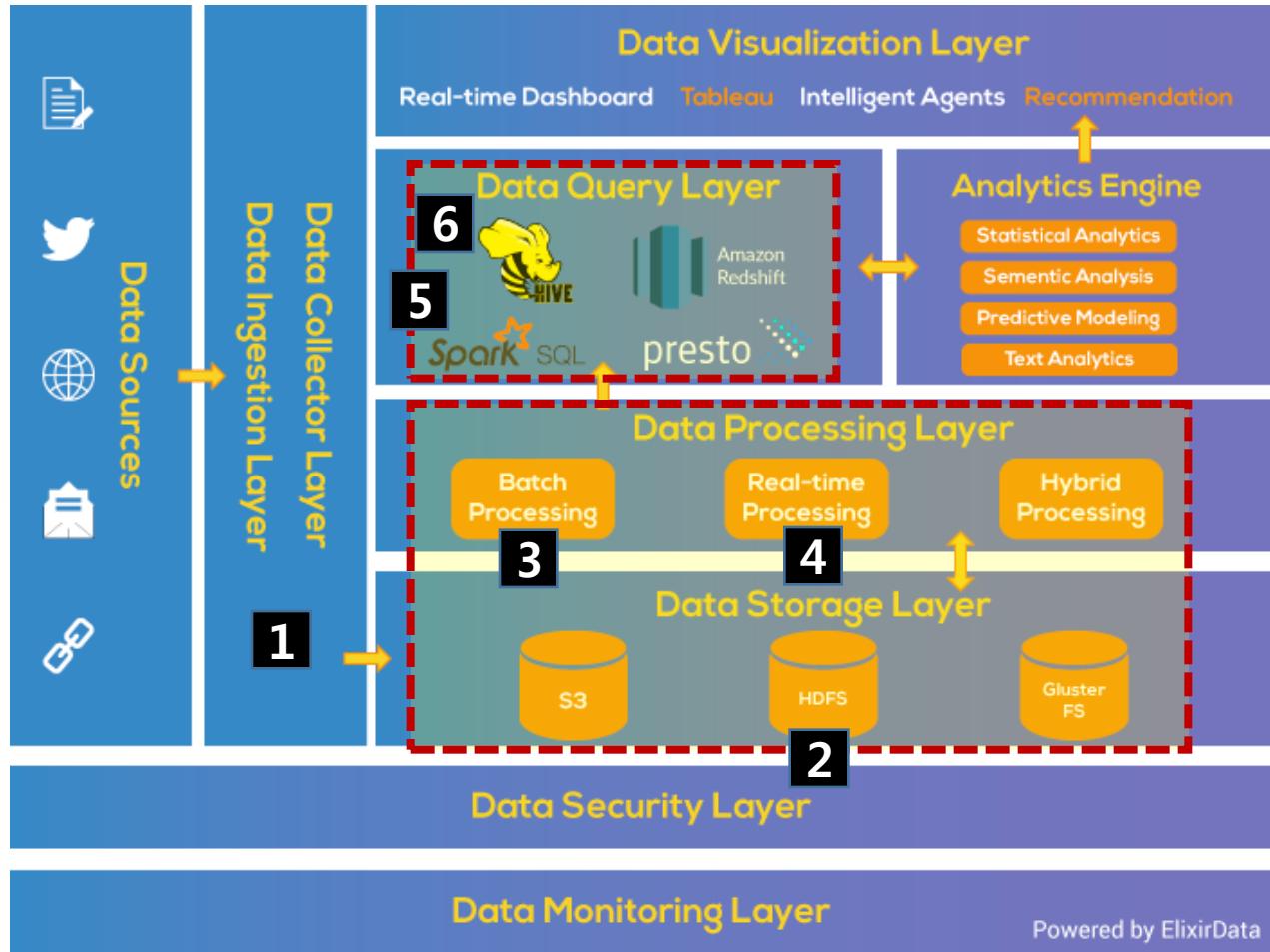
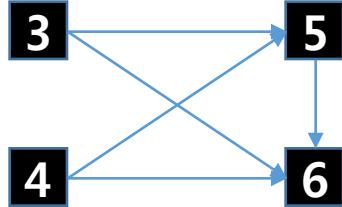
- Data 획득 주기

- Batch 
- Streaming



- Processing

- Exploration
- Model Input



[source] <https://www.xenonstack.com/blog/ingestion-processing-big-data-iot-stream/>

2. Process Step

4) Analytic Works

▪ Data Source에의 접근

- hdfs://<host:port>/<file-path>
- file//<file-path>

▪ Pre-Processing

▪ 특징 탐색

- 기술통계
- Visualization

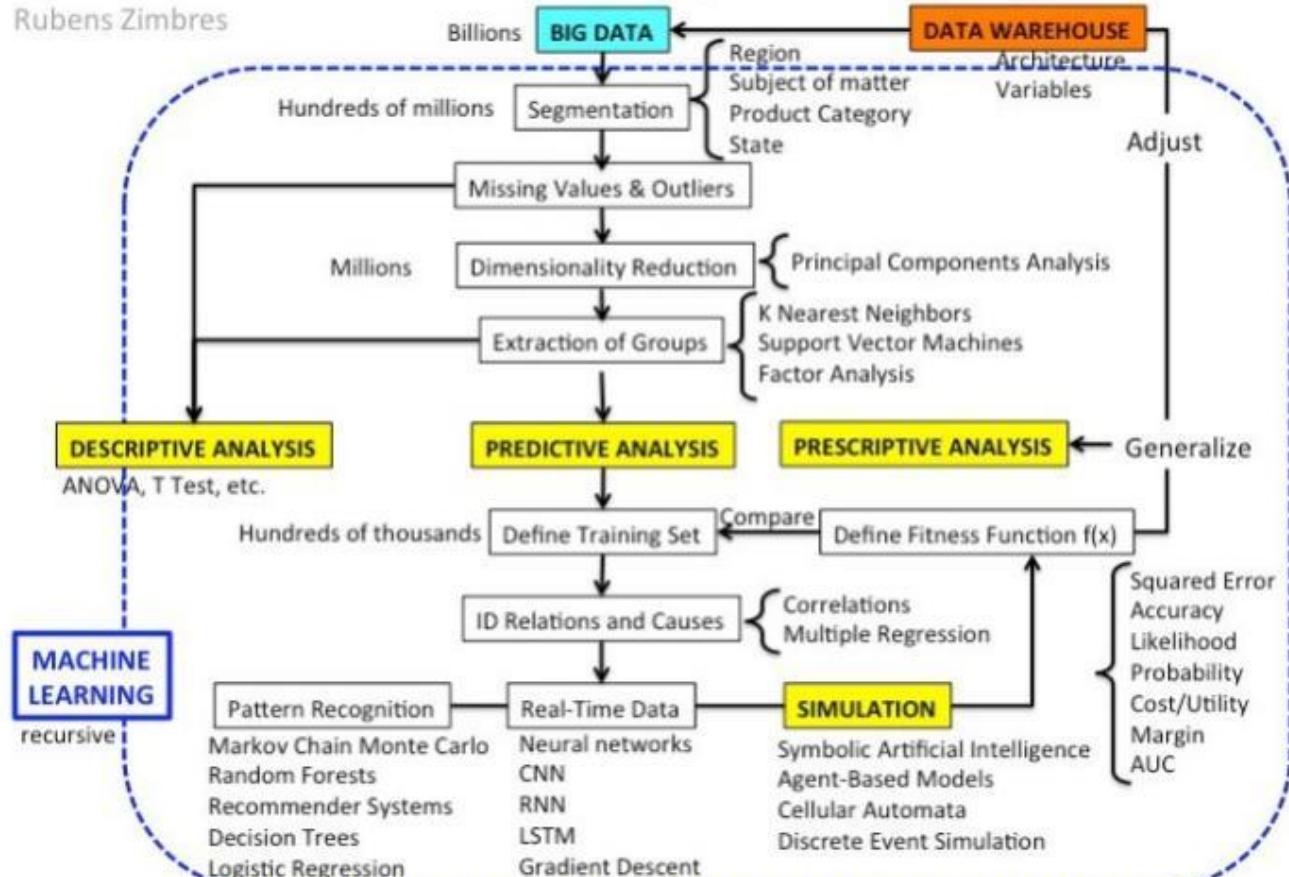
▪ Project 정의

- 인과관계의 규명(RCA)
- 예측
 - Prediction
 - Forecasting

▪ Project Life-Cycle

- Model Selection
- Training & Test
- Evaluation
- Tuning

Machine Learning Applied to Big Data



[source] <https://www.kdnuggets.com/2017/07/machine-learning-big-data-explained.html>

2. Process Step

15

4) Analytic Works

Model 선택

Training & Testing

Evaluation

Tuning

- Business Model에 따른 적합한 알고리즘 선택
 - 개별 알고리즘 고유의 원리와 특징 이해 필요

AI Model

Supervised

Classification

Regression

Unsupervised

Clustering

Semi-supervised

Label Type	Categorical	Continuous	n/a	Categorical
Machine Learning	<ul style="list-style-type: none"> Decision Tree Classification Random Forest Classification Support Vector Classification Naïve Bayes Classification kNN Classification 기타 	<ul style="list-style-type: none"> Decision Tree Regression Random Forest Regression Support Vector Regression Naïve Bayes Regression kNN Regression 기타 	<ul style="list-style-type: none"> K-Means kNN Clustering SVD PCA TSNE 기타 	
Deep Learning	<ul style="list-style-type: none"> MLP Classification CNN Classification RNN/LSTM Classification 	<ul style="list-style-type: none"> MLP Regression CNN Regression RNN/LSTM Regression 	<ul style="list-style-type: none"> Autoencoder GAN 	<ul style="list-style-type: none"> Reinforcement Learning RBM Word2Vec

2. Process Step

4) Analytic Works

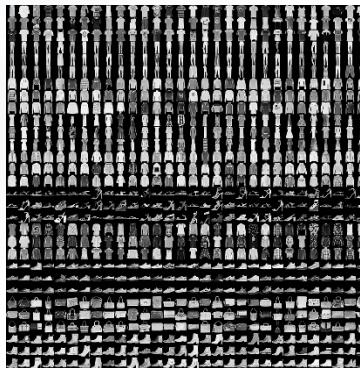
Model 선택

Training & Testing

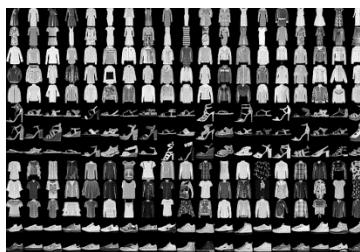
Evaluation

Tuning

- 실세계에 적용할 AI Model의 일반화 과정
 - 다양한 보편적인 실세계 조건에 부합하는 모델의 생성
 - 과적합 방지를 위한 반복 테스트



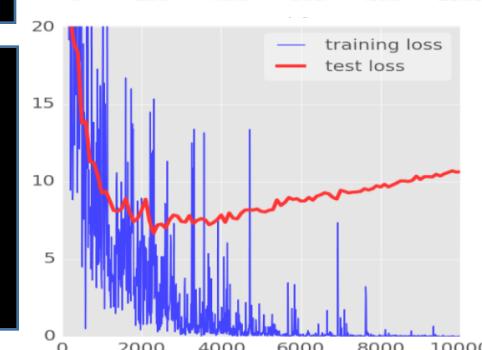
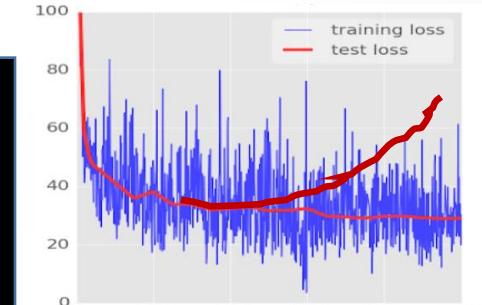
Training Data



Test Data

```
sess.run(  
    cross_entropy  
, feed_dict = {features : train_batch_features  
                           , label : train_batch_label}  
)
```

```
sess.run(  
    cross_entropy  
, feed_dict = {features : test_features  
                           , label : test_label}  
)
```



2. Process Step

17

4) Analytic Works

Model 선택

Training & Testing

Evaluation

Tuning

- Binary Class Classification
 - 평가 지표 : Accuracy, Recall, Precision, Recall, ROC/AUC

		True condition				
		Total population		Prevalence $= \frac{\sum \text{Condition positive}}{\sum \text{Total population}}$	Accuracy (ACC) = $\frac{\sum \text{True positive} + \sum \text{True negative}}{\sum \text{Total population}}$	
Predicted condition	Predicted condition positive	Condition positive	Condition negative	Positive predictive value (PPV), Precision = $\frac{\sum \text{True positive}}{\sum \text{Predicted condition positive}}$	False discovery rate (FDR) = $\frac{\sum \text{False positive}}{\sum \text{Predicted condition positive}}$	
	Predicted condition negative	True positive, Power	False positive, Type I error	False omission rate (FOR) = $\frac{\sum \text{False negative}}{\sum \text{Predicted condition negative}}$	Negative predictive value (NPV) = $\frac{\sum \text{True negative}}{\sum \text{Predicted condition negative}}$	
	True positive rate (TPR), Recall, Sensitivity, probability of detection $= \frac{\sum \text{True positive}}{\sum \text{Condition positive}}$	False positive rate (FPR), Fall-out, probability of false alarm $= \frac{\sum \text{False positive}}{\sum \text{Condition negative}}$	Positive likelihood ratio (LR+) $= \frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) $= \frac{\text{LR+}}{\text{LR-}}$	F ₁ score = $2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$	
		False negative rate (FNR), Miss rate $= \frac{\sum \text{False negative}}{\sum \text{Condition positive}}$	Specificity (SPC), Selectivity, True negative rate (TNR) $= \frac{\sum \text{True negative}}{\sum \text{Condition negative}}$	Negative likelihood ratio (LR-) $= \frac{\text{FNR}}{\text{TNR}}$		

[source] https://en.wikipedia.org/wiki/Evaluation_of_binary_classifiers

Author Yi-Beck Lee(Yibeck.Lee@gmail.com)

2. Process Step

18

4) Analytic Works

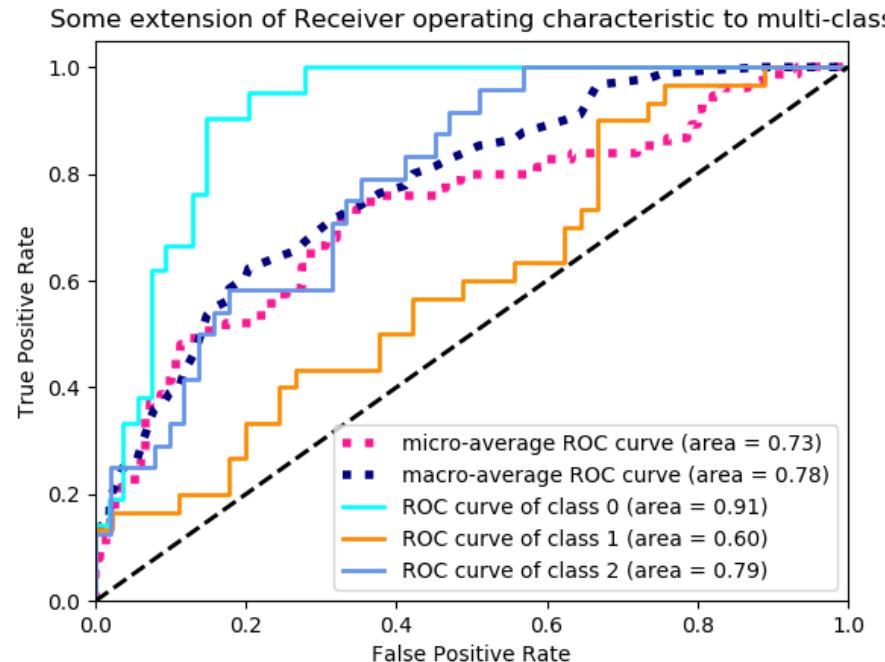
Model 선택

Training & Testing

Evaluation

Tuning

- Binary Class Classification
 - ROC / AUC



[source] https://scikit-learn.org/stable/modules/model_evaluation.html

Author Yi-Beck Lee(Yibeck.Lee@gmail.com)

2. Process Step

4) Analytic Works

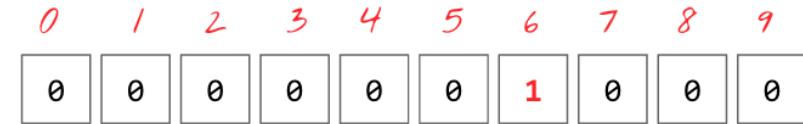
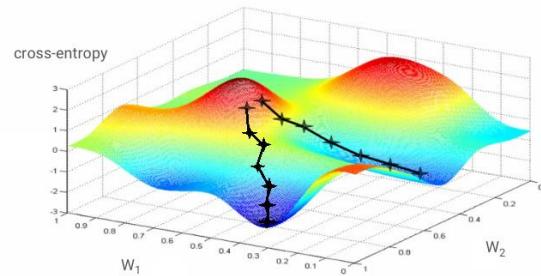
Model 선택

Training & Testing

Evaluation

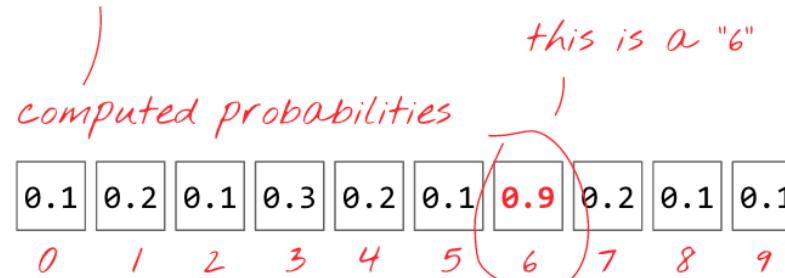
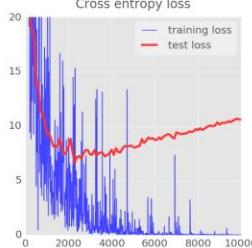
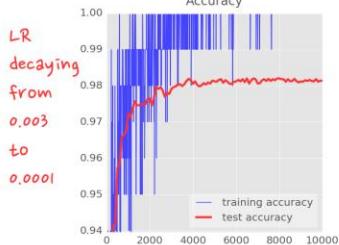
Tuning

- Multi-Class Classification
 - 평가 지표 : Cross-Entropy



actual probabilities, "one-hot" encoded

Cross entropy: $-\sum Y'_i \cdot \log(Y_i)$



[source] <https://codelabs.developers.google.com/codelabs/cloud-tensorflow-mnist/>

2. Process Step

20

4) Analytic Works

Model 선택

Training & Testing

Evaluation

Tuning

- Regression
 - 평가지표 : MSE[, RMSE, MAE, MAPE], Average Cost

MSE

Mean squared error

$$MSE = \frac{1}{n} \sum_{t=1}^n e_t^2$$

Root mean squared error

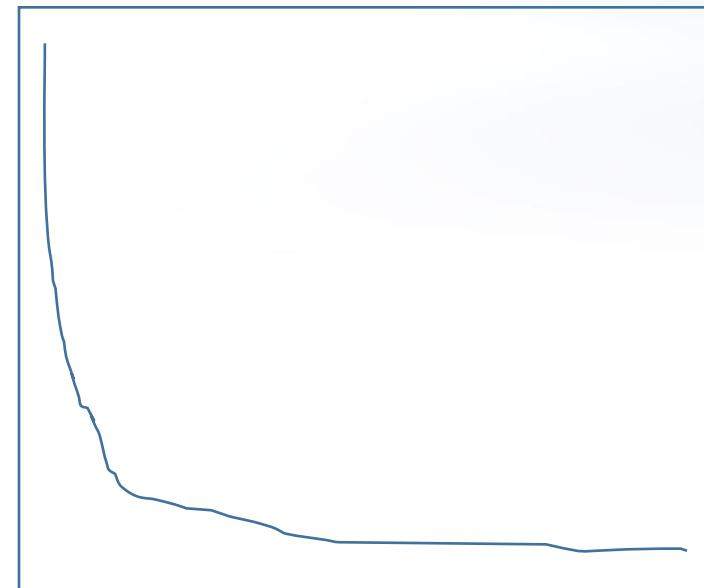
$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n e_t^2}$$

Mean absolute error

$$MAE = \frac{1}{n} \sum_{t=1}^n |e_t|$$

Mean absolute percentage error

$$MAPE = \frac{100\%}{n} \sum_{t=1}^n \left| \frac{e_t}{y_t} \right|$$



Number Of Iteration

2. Process Step

21

4) Analytic Works

Model 선택

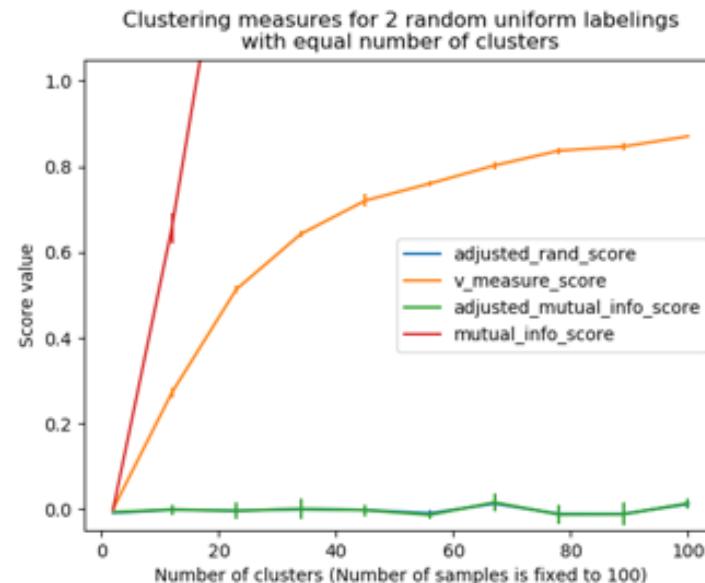
Training & Testing

Evaluation

Tuning

- Unsupervised Model
 - Clustering 주요 지표 : Adjusted Mutual Information Score 外

- Adjusted Mutual Information Score
- Adjusted Rand Score
- Completeness Score
- Fowlkes mallows Score
- Homogeneity Score
- Mutual Information Score
- Normalized Mutual Information Score
- V Measure Score



[source] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.adjusted_mutual_info_score.html

2. Process Step

4) Analytic Works

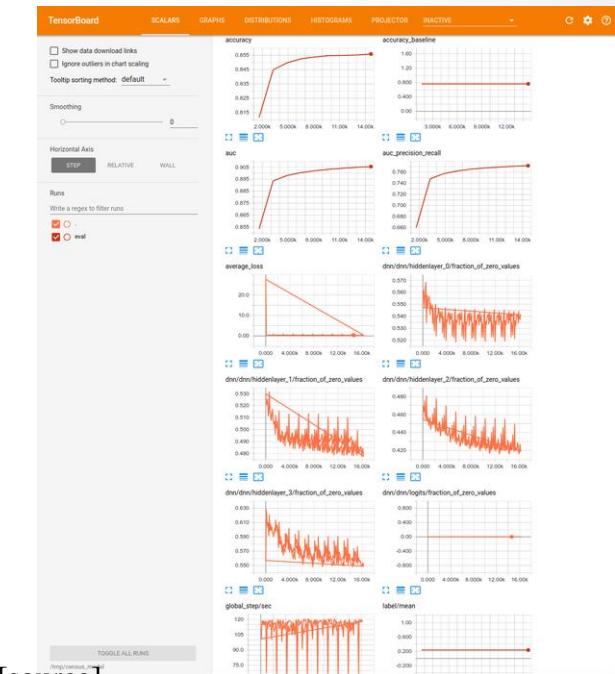
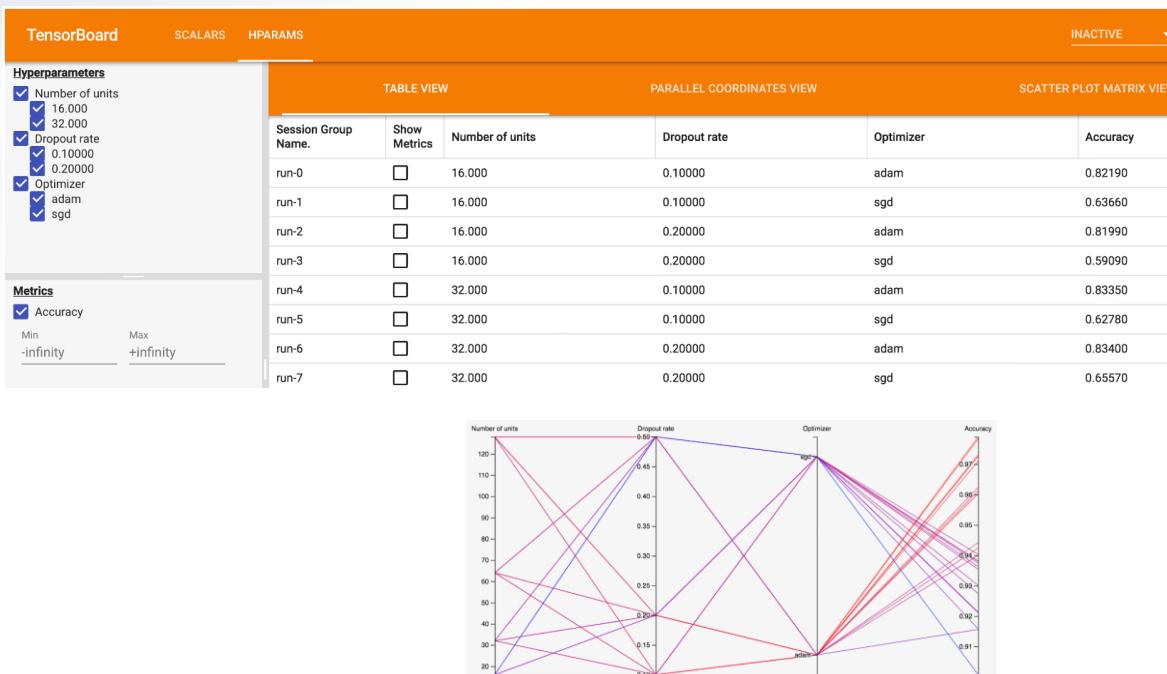
Model 선택

Training & Testing

Evaluation

Tuning

- Business Model에 따른 적합한 알고리즘 선택
 - 알고리즘의 서로 다른 특징의 이해 필요



[source] https://www.tensorflow.org/tensorboard/r2/hyperparameter_tuning_with_hparams

<https://bookdown.org/max/FES/model-optimization-and-tuning.html>

2. Process Step

23

4) Analytic Works

Model 선택

Training & Testing

Evaluation

Tuning

- Business Model에 따른 적합한 알고리즘 선택
 - 알고리즘의 서로 다른 특징의 이해 필요

Units	Dropout	Batch Size	Learn Rate	Grad. Scaling	Decay	Act. Fun.	ROC
7	0.3368	11348	0.00385	0.55811	1.16e-04	sigmoid	0.838
5	0.4023	36070	0.04142	0.95232	3.84e-02	sigmoid	0.837
12	0.6619	38206	0.25837	0.63325	3.09e-02	sigmoid	0.837
6	0.4720	38651	0.02634	0.80618	4.94e-05	sigmoid	0.834
7	0.3918	17235	0.01681	0.36270	2.90e-04	tanh	0.830
3	0.1190	16369	0.22210	0.22683	4.02e-02	relu	0.830
10	0.4979	19103	0.04818	0.83560	1.92e-03	relu	0.828
7	0.6139	38198	0.30864	0.68575	1.67e-03	sigmoid	0.824
2	0.3797	22255	0.10597	0.93841	4.27e-05	sigmoid	0.824
4	0.0694	18167	0.45844	0.94679	2.97e-03	tanh	0.811
14	0.6279	33800	0.18082	0.33286	1.08e-05	tanh	0.810
10	0.1466	33139	0.44443	0.72107	6.22e-05	tanh	0.810
20	0.0497	29465	0.40072	0.49598	7.07e-02	sigmoid	0.804
17	0.1068	17953	0.43256	0.87800	1.99e-04	tanh	0.794
13	0.5570	13558	0.13159	0.20389	5.96e-05	relu	0.793
7	0.6183	30606	0.82481	0.71944	2.61e-03	sigmoid	0.792
14	0.3866	30514	0.92724	0.38651	3.36e-03	tanh	0.782
2	0.0903	33439	0.63991	0.00398	1.92e-03	tanh	0.776
11	0.5909	32417	0.61446	0.63142	3.08e-04	tanh	0.731
10	0.4234	34655	0.49455	0.25216	3.05e-04	relu	0.698

[source] <https://bookdown.org/max/FES/model-optimization-and-tuning.html>

Author Yi-Beck Lee(Yibeck.Lee@gmail.com)

2. Process Step

24

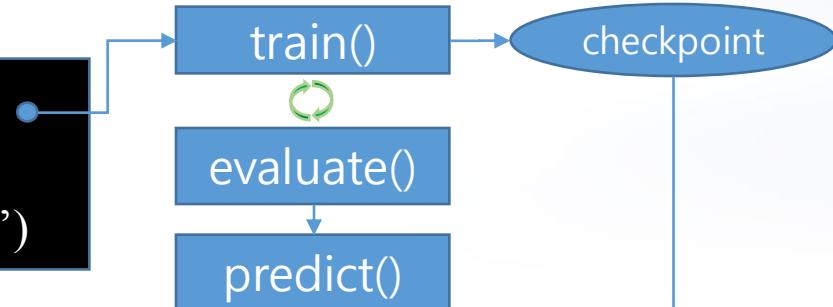
5) AI Model Save & Restore > TensorFlow Case

- Model Save : 수행 모델의 저장(save)
- Model Restore : 저장된 모델의 재학습을 위한 복원 restore)
- 모델링 수행 성과의 추적 가능 - checkpoint

■ Model Save and Restore - TensorFlow

Model Save

```
saver = tf.train.Saver()  
...  
saver.save(sess, 'model')
```



Model Restore

```
tf.reset_default_graph()  
restored = tf.train.import_meta_graph("model.meta")  
...  
restored.restore(sess, tf.train.latest_checkpoint('.'))
```

2. Process Step

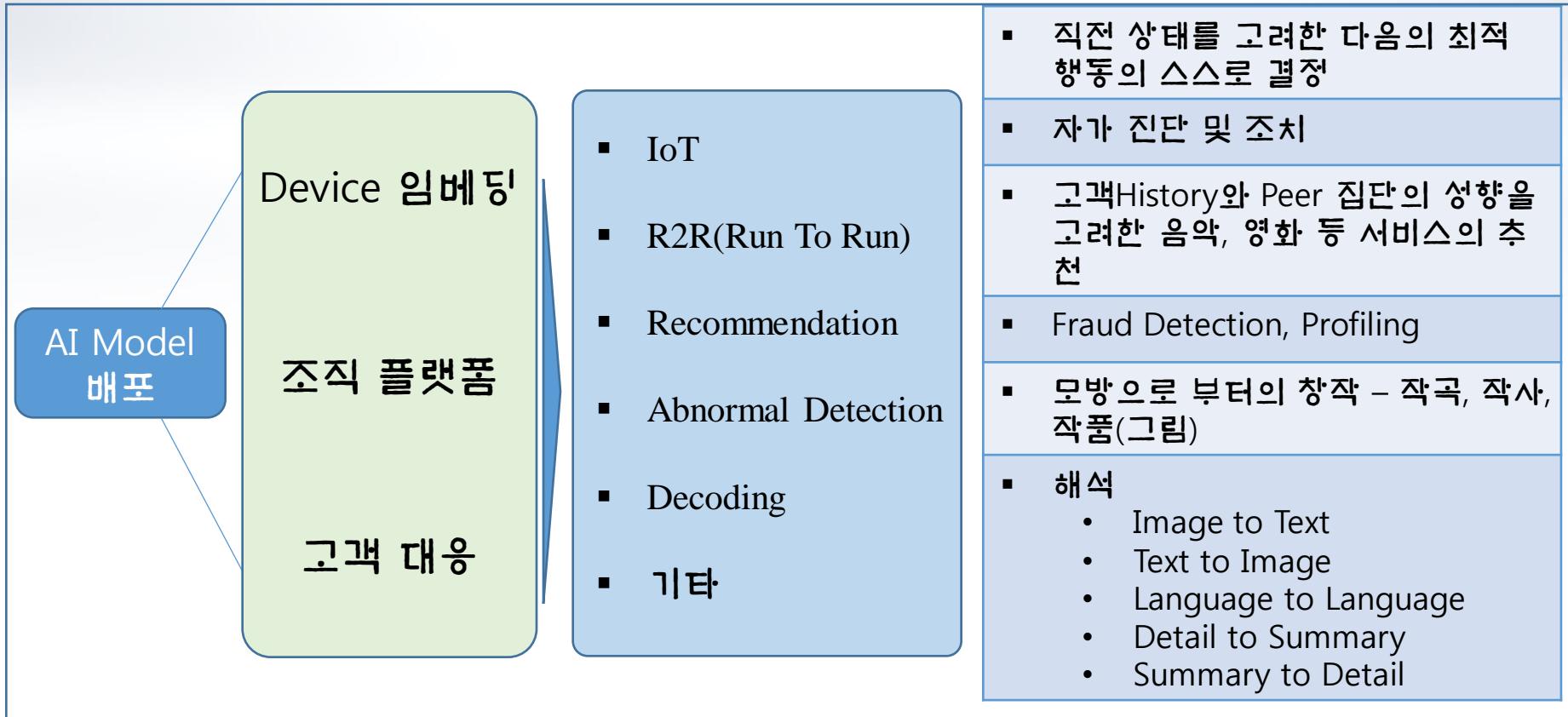
25

6) AI Model Service

- AI Model의 배포(Deploy)

- AI Server에서 학습되어 검증된 모델의 실세계 서비스 시스템에 배포

■ AI Service Model의 다양한 Case들

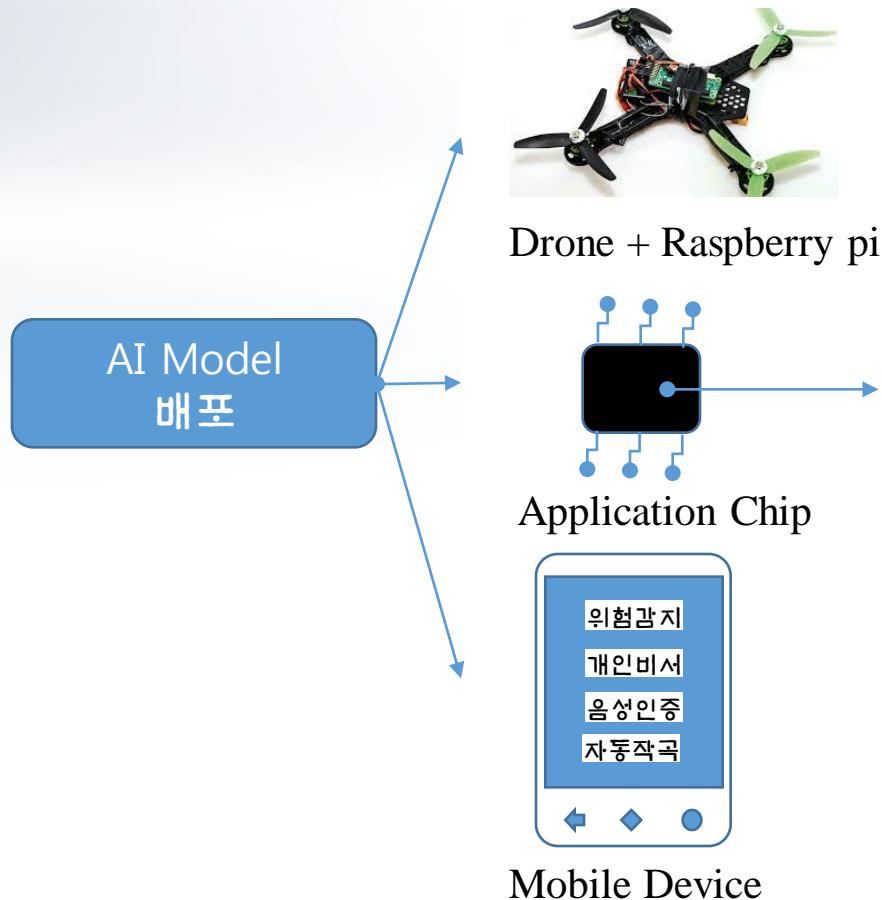


2. Process Step

26

6) AI Model Service

- AI in Device – Sensor, Operation, R2R, Abnormal Detection 등
- Embedded System : Linux/Windows, Raspberry pi, Application Chip



Devices, machines,
and things are becoming
more intelligent



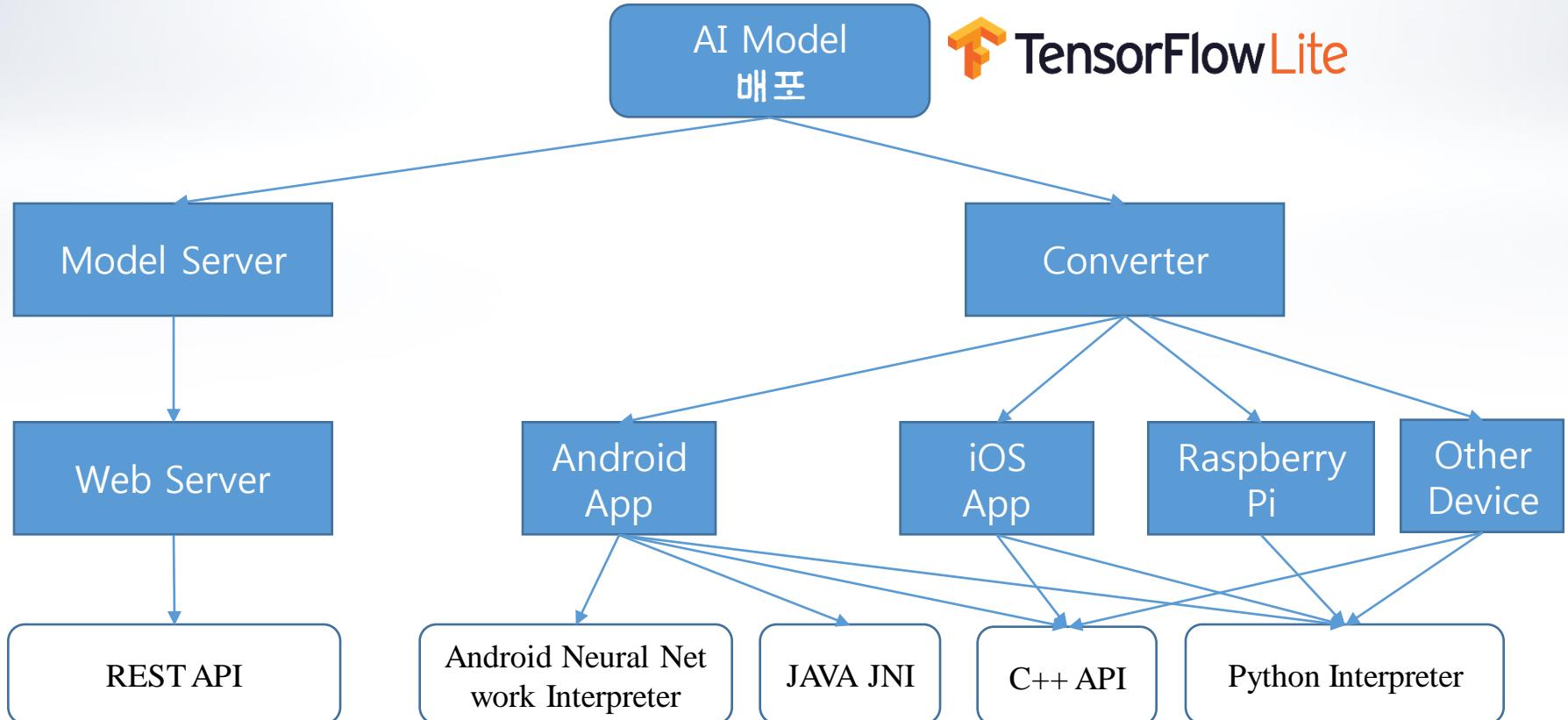
[source] <https://www.qualcomm.com/news/onq/2017/08/16/we-are-making-device-ai-ubiquitous?cmpid=oofyus181544>

2. Process Step

27

6) AI Model Service

- AI Server에서 학습되어 검증된 모델의 실세계 서비스 시스템/Device에 배포
- AI Model의 배포(Deploy) > TensorFlow Case



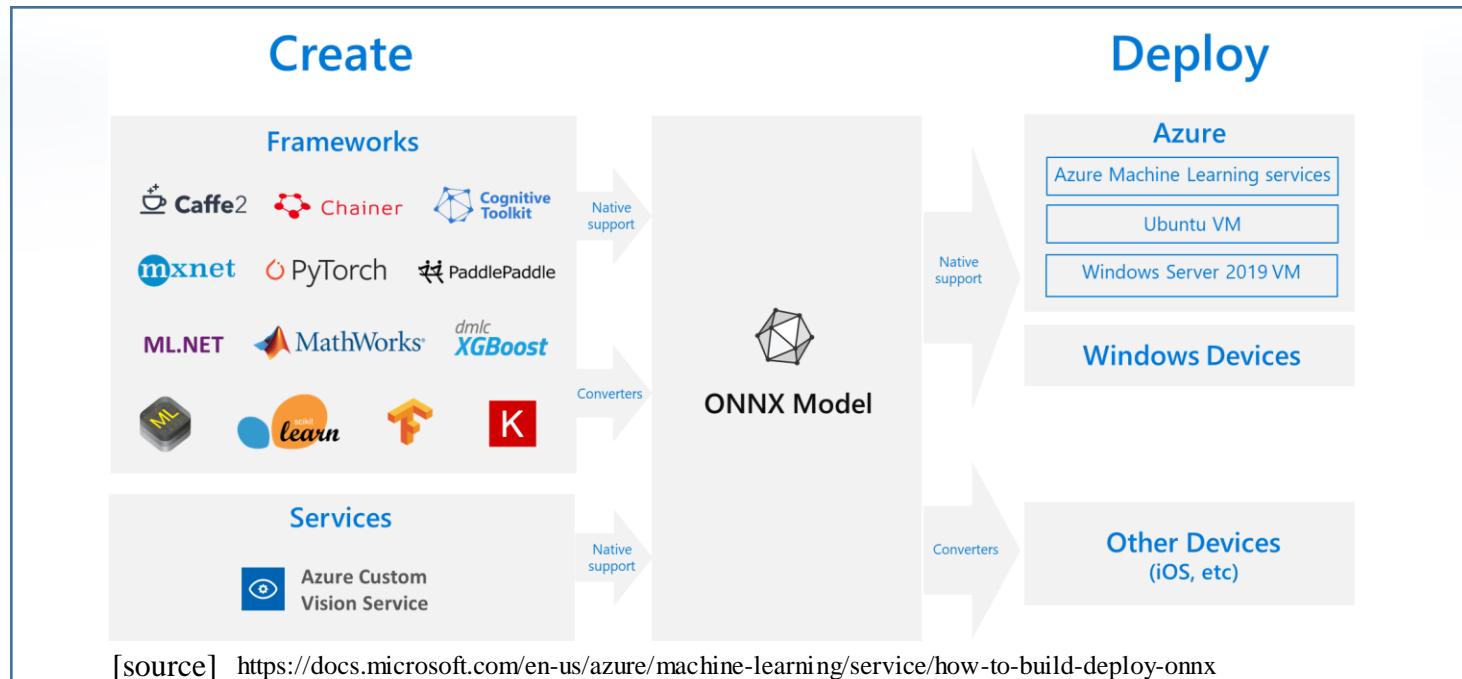
2. Process Step

28

6) AI Model Service

- AI Model의 배포(Deploy) > ONNX Case
 - ONNX : Open Neural Network eXchange
 - 모델의 상호 운용성
 - 서로 다른 framework 간의 Model Import & Export 가능
 - 서로 다른 서비스 Platform(Cloud, Device)에 대한 통합된 라이브러리

■ Azure Cloud 환경에서의 ONNX 기반 Model 배포



2. Process Step

29

6) AI Model Service

- AI Model의 배포(Deploy) > ONNX Case
 - convert TensorFlow models to ONNX models

Convert TensorFlow models to ONNX

tensorflow onnx export convert lstm

996 commits 10 branches 3 releases 19 contributors MIT

Branch: master New pull request Create new file Upload files Find File Clone or download

guschmue Merge pull request #482 from nbcsm/retry	Latest commit 022b718 Apr 26, 2019
ci_build/azure_pipelines support python 3.7 in ci	Apr 25, 2019
examples revert change to custom_op_via_python.py, fix custom_op_handler issue	Mar 29, 2019
tests Merge pull request #482 from nbcsm/retry	Apr 26, 2019
tf2onnx Merge pull request #482 from nbcsm/retry	Apr 26, 2019
tools unify logger name	Apr 16, 2019
.gitignore resolve requests	Mar 7, 2019
.travis.yml travis use onnxruntime 0.1.4	Dec 20, 2018
LICENSE Update LICENSE to match ONNX copyright	Apr 3, 2019
README.md Update README.md	Apr 25, 2019
VERSION_NUMBER increment version	Apr 1, 2019
build.bat initial drop for tf2onnx	Mar 16, 2018
build.sh - simplify unit tests	Mar 26, 2018
setup.cfg enable all tests for pytest	Feb 23, 2019
setup.py support retry for url get	Apr 26, 2019

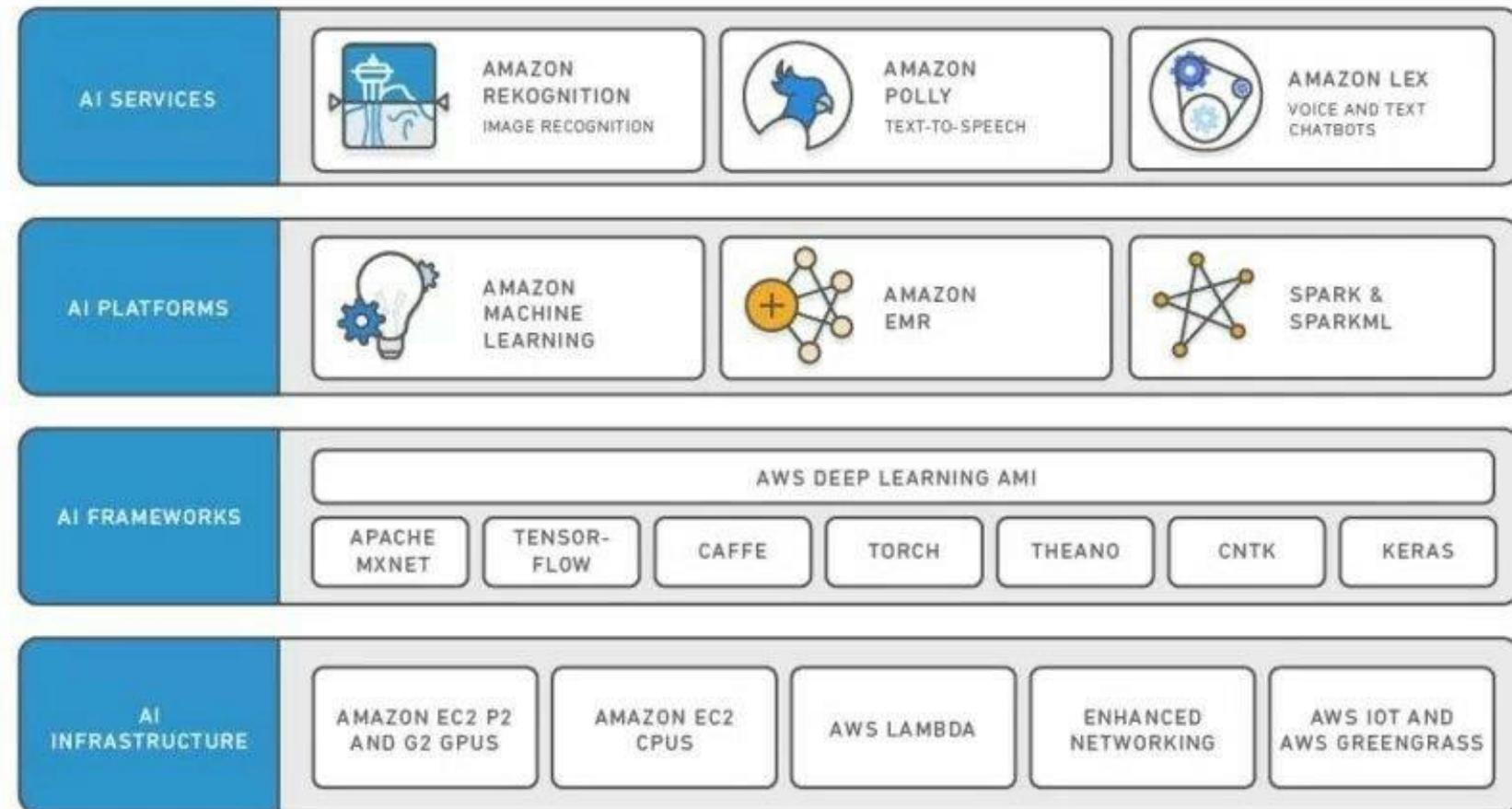
[source] <https://github.com/onnx/tensorflow-onnx>

2. Process Step

30

6) AI Model Service

- AI-a-a-S : Artificial Intelligence as a Service
 - AWS Case



[source] <https://www.nanalyze.com/2017/05/artificial-intelligence-service-core-ai/>
<https://i.pinimg.com/originals/00/17/25/001725085850742f185e5e28f5ef9115.jpg>

Author Yi-Beck Lee(Yibeck.Lee@gmail.com)

2. Process Step

6) AI Model Service

- Top 100 AI Startups

