INNOVATE

AI/ML EDITION

24 February 2022

모두를 위한 클라우드 네이티브 한국어 자연어 처리 모델 훈련 및 활용법

허깅페이스와 Amazon SageMaker가 만났다!

김대근
AIML 전문 솔루션즈 아키텍트
AWS

Agenda

- Hugging Face on Amazon SageMaker
- Amazon SageMaker Training Compiler
- Demo: 한국어 자연어 처리 모델 훈련, 배포 및 자동화

Hugging Face on Amazon SageMaker

AWS와 허깅페이스의 강력한 파트너십

Hugging Face 라이브러리

Open-source
`Datasets`, `Tokenizers` and `Transformers`

Popular
55,000개 이상의 GitHub star (2021년 12월 기준), 월별 1백만 건 이상의 다운로드

Intuitive
PyTorch & TensorFlow를 기반으로 하는 NLP 전용 Python 프론트엔드

State of the art
트랜스포머 기반 모델은 최첨단state-of-the-art모델로, 전이 학습transfer learning 및 스케일링이 간편합니다.

Comprehensive
10,000개 이상의 모델 아키텍처, 240개 이상의 다국어가 있는 모델 허브hub

huggingface.co/daekeun-ml/koelectra-small-v3-nsmc?text=최악의+영화입니다.+절대로+보지+마세요

Hugging Face | Search models, datasets, users... | Models | Datasets | Spaces | Docs | Solutions | Pricing

Hugging Face is way more fun with friends and colleagues! 🤗 Join an organization | Dismiss this message

daekeun-ml / **koelectra-small-v3-nsmc** | like 0

Text Classification | PyTorch | Transformers | electra | Infinity Compatible

Model card | Files and versions | Settings | Train | Deploy | Use in Transformers

Edit model card

Sentiment Binary Classification (fine-tuning with KoELECTRA-Small-v3 model and Naver Sentiment Movie Corpus dataset)

Usage (Amazon SageMaker inference applicable)

It uses the interface of the SageMaker Inference Toolkit as is, so it can be easily deployed to SageMaker Endpoint.

inference_nsmc.py

```
import json
import sys
import logging
import torch
from torch import nn
from transformers import ElectraConfig
from transformers import ElectraModel, AutoTokenizer, ElectraTokenizer, ElectraF

logging.basicConfig(
    level=logging.INFO,
    format='[%(filename)s:%(lineno)d] %(levelname)s - %(message)s',
    handlers=[
        logging.FileHandler(filename='tmp.log')
```

Downloads last month
151

Hosted inference API

Text Classification | Example 1

Your sentence here...

Compute

Computation time on cpu: 0.0156 s

0 0.999

1 0.001

JSON Output | Maximize

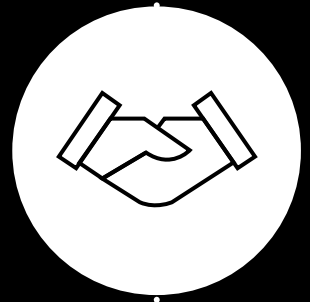
https://hf.co/daekeun-ml/koelectra-small-v3-nsmc

자연어 처리Natural Language Processing의 강력한 파트너십

Hugging Face

Hugging Face는 최신 NLP 기술을 제공하는 가장 인기 있는 오픈 소스 회사입니다.

https://huggingface.co/



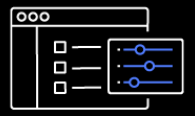
AWS

Amazon SageMaker는 NLP 모델을 훈련하고 배포하기 위한 고성능 리소스를 제공합니다.

https://aws.amazon.com/sagemaker/

Hugging Face on Amazon SageMaker의 이점

Cost-effective - SageMaker는 비용을 줄이기 위해 스케일링, 성능performance 및 효율성effectiveness 을 최적화합니다. EC2 스팟 인스턴스로 비용을 추가로 절약할 수 있습니다.



MLOps-ready - 자동화된 메타데이터 지속성 및 SageMaker 메타스토어의 검색, Amazon CloudWatch 로그 추출, SageMaker 디버거debugger 및 프로파일러profiler를 사용한 모니터링, 실험 관리가 포함됩니다.

Scalable – Amazon SageMaker Data Parallelism 및 Model Parallelism을 사용하여 GPU 클러스터를 효율적으로 사용할 수 있습니다. API는 비동기asynchronous 모드로 여러 동시 작업을 동시에 시작하는 기능을 제공합니다.

Secure – 저장 및 전송 중 암호화, VPC 연결 및 세분화된 IAM 권한을 비롯한 다양한 메커니즘을 통해 보안에 대한 높은 기준을 제시합니다.

Hugging Face 딥러닝 컨테이너 (DLC)

Top-quality – Hugging Face는 AWS의 관리형 컨테이너를 통해 곧바로 사용할 수 있으며, 지속적으로 최신 Hugging Face 컨테이너가 업데이트됩니다.

Less heavy lifting – Amazon ECR을 자체 관리할 필요가 없으며, 별도의 인프라 설정, 추가 SDK 설치가 필요 없습니다.

HuggingFace training containers

Framework	Job Type	CPU/GPU	Python Version Options	Example URL
PyTorch 1.9.1 with HuggingFace transformers	training	GPU	3.8 (py38)	763104351884.dkr.ecr.us-east-1.amazonaws.com/huggingface-pytorch-training:1.9.1-transformers4.12.3-gpu-py38-cu111-ubuntu20.04
TensorFlow 2.5.1 with HuggingFace transformers	training	GPU	3.7 (py37)	763104351884.dkr.ecr.us-east-1.amazonaws.com/huggingface-tensorflow-training:2.5.1-transformers4.12.3-gpu-py37-cu112-ubuntu18.04

HuggingFace inference containers

Framework	Job Type	CPU/GPU	Python Version Options	Example URL
PyTorch 1.9.1 with HuggingFace transformers	inference	CPU	3.8 (py38)	763104351884.dkr.ecr.us-east-1.amazonaws.com/huggingface-pytorch-inference:1.9.1-transformers4.12.3-cpu-py38-ubuntu20.04
PyTorch 1.9.1 with HuggingFace transformers	inference	GPU	3.8 (py38)	763104351884.dkr.ecr.us-east-1.amazonaws.com/huggingface-pytorch-inference:1.9.1-transformers4.12.3-gpu-py38-cu111-ubuntu20.04
TensorFlow 2.5.1 with HuggingFace transformers	inference	CPU	3.7 (py37)	763104351884.dkr.ecr.us-east-1.amazonaws.com/huggingface-tensorflow-inference:2.5.1-transformers4.12.3-cpu-py37-ubuntu18.04
TensorFlow 2.5.1 with HuggingFace transformers	inference	GPU	3.7 (py37)	763104351884.dkr.ecr.us-east-1.amazonaws.com/huggingface-tensorflow-inference:2.5.1-transformers4.12.3-gpu-py37-cu112-ubuntu18.04

Available DLC images: https://github.com/aws/deep-learning-containers/blob/master/available_images.md

aws

Hugging Face 딥러닝 컨테이너 (DLC): AWS CLI

Training DLC

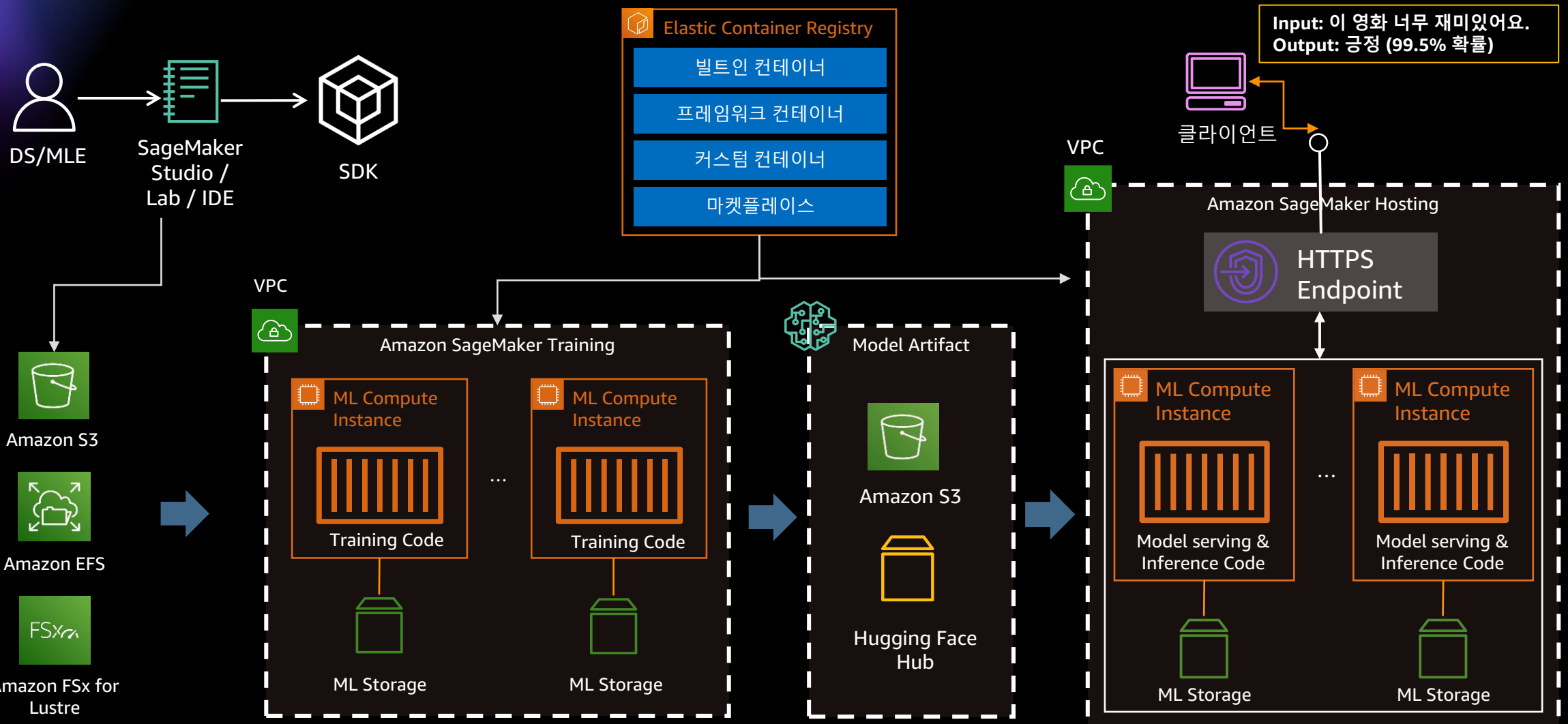
```
$ aws ecr list-images --repository-name huggingface-pytorch-training --registry-id 763104351884
$ aws ecr list-images --repository-name huggingface-tensorflow-training --registry-id 763104351884
```

Inference DLC

```
$ aws ecr list-images --repository-name huggingface-pytorch-inference --registry-id 763104351884
$ aws ecr list-images --repository-name huggingface-tensorflow-inference--registry-id 763104351884
```


```
"imageIds": [
    {
        "imageDigest": "sha256:3891f97bd1f86f8cda9d7b0218467d24ea1f0df1b00174d6004032064e91ff25",
        "imageTag": "1.9.1-transformers4.12.3-gpu-py38-cu111-ubuntu20.04-v1.0"
    },
    {
        "imageDigest": "sha256:3891f97bd1f86f8cda9d7b0218467d24ea1f0df1b00174d6004032064e91ff25",
        "imageTag": "1.9-gpu-py38-cu111-ubuntu20.04-v1"
    },
    {
        "imageDigest": "sha256:cb36c58bd99baf69eb5204c7888c3f414c39aa9aa9007fb869ad7959b542c5d4",
        "imageTag": "1.8.1-transformers4.6.1-gpu-py36-cu111-ubuntu18.04-v1.0-2021-08-24-21-56-55"
    },
```

End-to-end Hugging Face w/ SageMaker



Hugging Face 모델 훈련 (SageMaker SDK)

```
from sagemaker.huggingface import HuggingFace
Hyperparameters = {
    "model_name": "distilbert-base-uncased", ...
}
distribution = {
    "smdistributed": {"dataparallel": { "enabled": True }}
}

hf_estimator = HuggingFace(
    entry_point="src/train.py",
    instance_type="ml.p3.2xlarge",
    transformers_version="4.11.0",
    pytorch_version="1.9.0",
    py_version="py38",
    hyperparameters=hyperparameters, ...
)
hf_estimator.fit(
    {"train": [YOUR-S3-PATH], "test": [YOUR-S3-PATH]}
)
```

1. 하이퍼파라메터 설정
2. 분산 훈련 설정 (SageMaker Data Parallel, SageMakerk Model Parallel)
3. Hugging Face estimator 인스턴스 생성
4. 훈련 job 시작 (훈련 인스턴스 프로비저닝)

aws

Hugging Face 모델 배포 (SageMaker SDK)

1 기존 방법

```
# Trained model artifact
s3_path="s3://[YOUR-BUCKET]/model.tar.gz"



hf_model = HuggingFaceModel(
   model_data=s3_path,
   transformers_version="4.11.0",
   pytorch_version="1.9.0",
   py_version="py38",
)

predictor = hf_model.deploy(
   initial_instance_count=1,
   instance_type="ml.m5.xlarge"
)
```

2 Hub에서 직접 가져오기 NEW

```
# Hub model configuration; hf.co/models
hub = {
  "HF_MODEL_ID":"bert-base-uncased",
  "HF_TASK":"question-answering"
}

hf_model = HuggingFaceModel(
   env=hub,
   transformers_version="4.11.0",
   pytorch_version="1.9.0",
   py_version="py38",
)

predictor = hf_model.deploy(
   initial_instance_count=1,
   instance_type="ml.m5.xlarge"
)
```

https://huggingface.co/transformers/main_classes/pipelines.html

daekeun-ml/koelectra-small-v3-nsmc

like 0

Text Classification PyTorch Transformers electra Infinity Compatible

Model card Files and versions Settings Train Deploy Use in Transformers

Edit model card

Sentiment Binary Classification (fine-tuning with KoELECTRA-Small-v3 model and Naver Sentiment Movie Corpus dataset)

Usage (Amazon SageMaker inference applicable)

It uses the interface of the SageMaker Inference Toolkit as is, so it can be easily deployed to SageMaker Endpoint.

inference_nsmc.py

```
import json
import sys
import logging
import torch
from torch import nn
from transformers import ElectraConfig
from transformers import ElectraModel, AutoTokenizer, ElectraTokenizer, ElectraF

logging.basicConfig(
    level=logging.INFO,
    format='[{%(filename)s:%(lineno)d} %(levelname)s - %(message)s',
    handlers=[
        logging.FileHandler(filename='tmp.log')
```

Downloads last month
151

Hosted inference API

Text Classification Example 1

정말 감동적인 스토리입니다. 꼭 보세요

Compute

Computation time on cpu: 0.0144 s

0 0.024

1 0.976

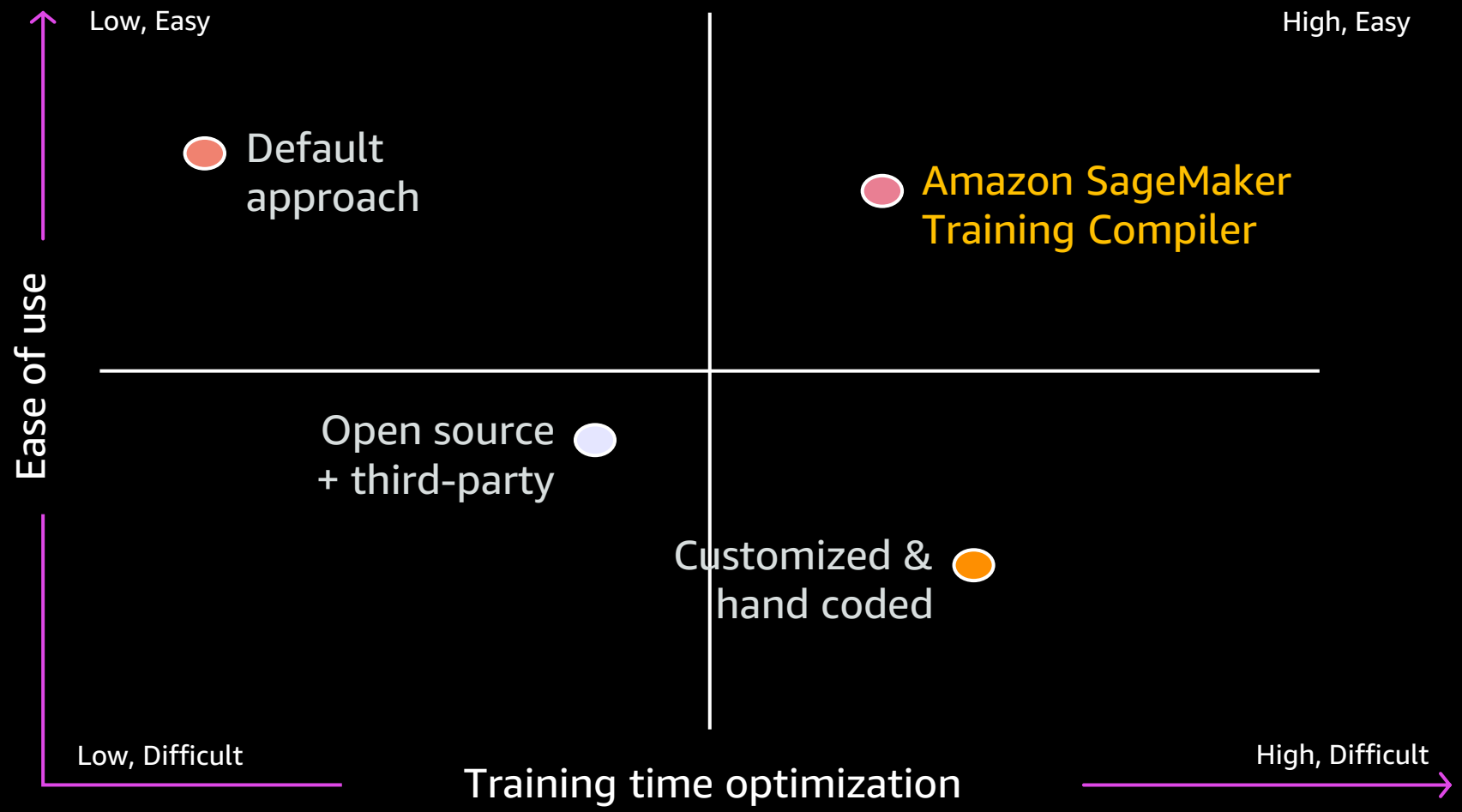
JSON Output Maximize

Amazon SageMaker Training Compiler

딥러닝 모델 훈련 속도 가속화

aws

딥러닝 훈련 컴파일링



NEW

Amazon SageMaker Training Compiler

GPU에서 대규모 딥러닝 모델을 훈련하는 빠르고 쉬운 방법

딥러닝 모델 훈련 가속화

훈련 속도 최대 50% 향상

최소한의 코드 변경 필요

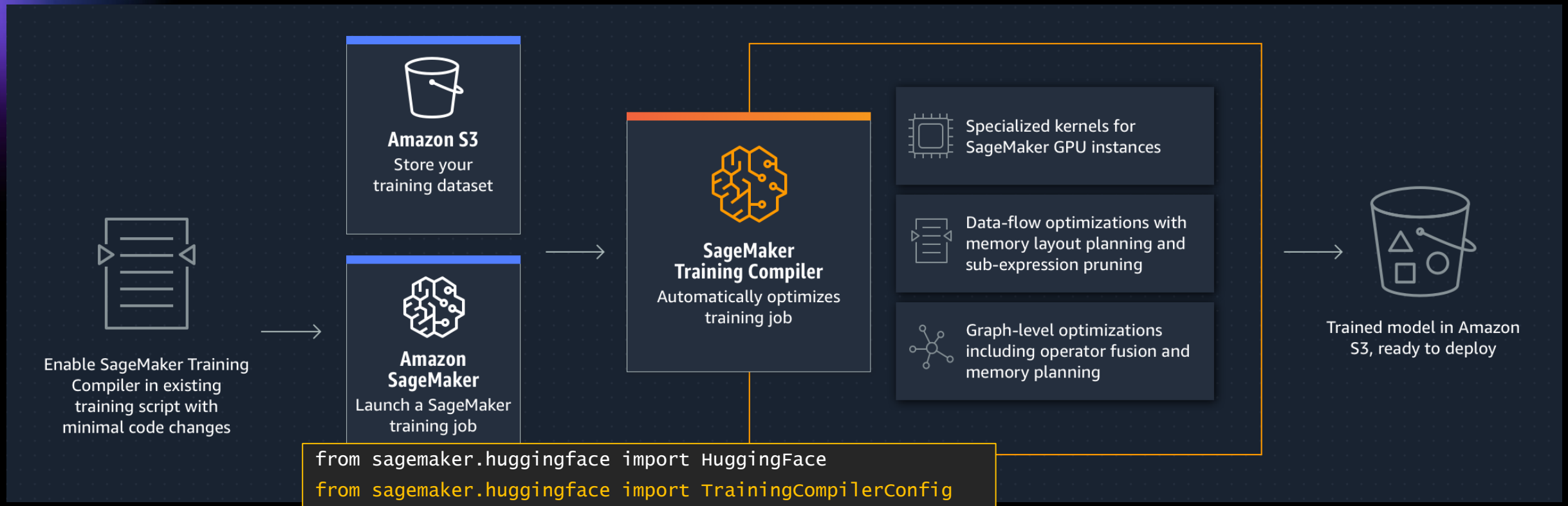
워크플로 변경 없이 몇 분 안에 활성화

훈련 비용 절감

SageMaker에서 무료로 사용할 수 있으며, 훈련 시간 단축으로 추가 비용 절감

aws

동작 원리: 워크플로 중단 없이 훈련 가속화

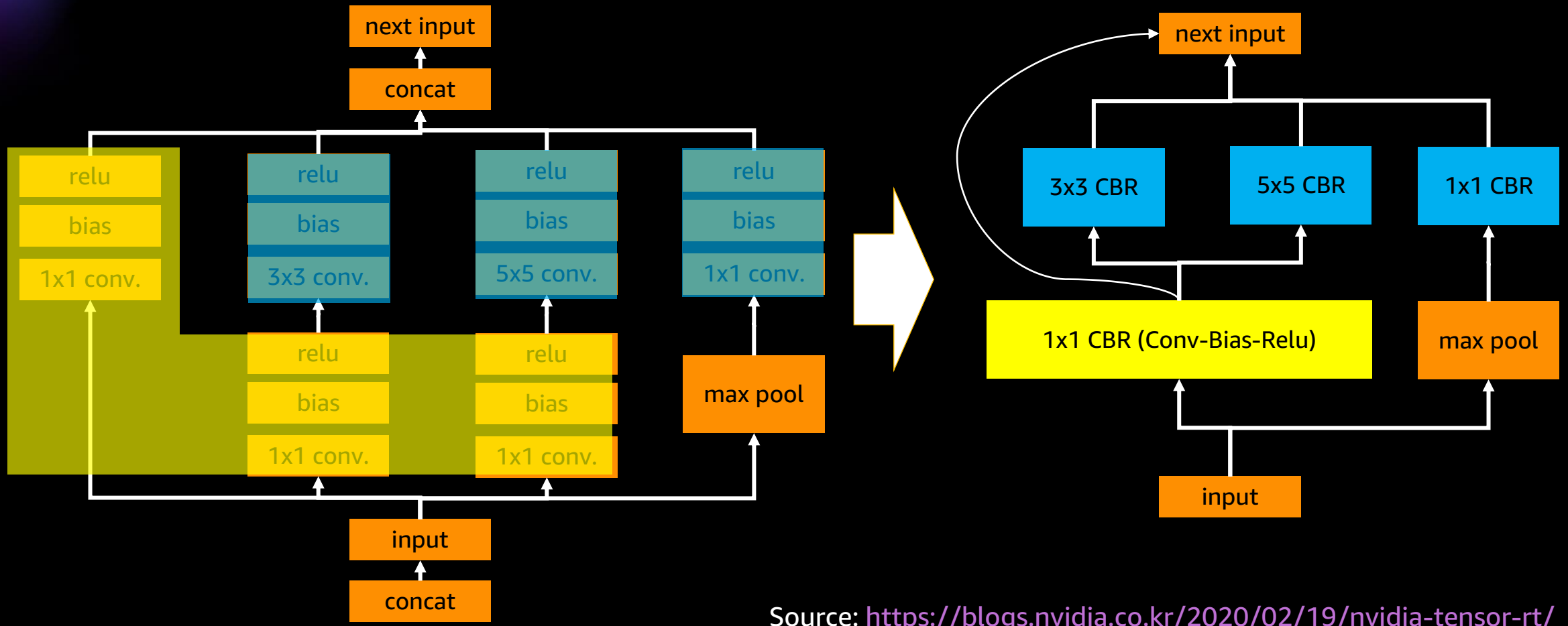



```
from sagemaker.huggingface import HuggingFace
from sagemaker.huggingface import TrainingCompilerConfig
hf_estimator = HuggingFace(
            entry_point='train.py',
            instance_type='ml.p3.2xlarge',
            pytorch_version='1.9.0',
            transformers_version='4.11.0',
            compiler_config=TrainingCompilerConfig(),
            ...
)
hf_estimator.fit(...)
```

TrainingCompilerConfig()만 설정하면 훈련 job 시작 시 자동으로 SageMaker Training Compiler 활성화

Graph Optimization

Operator Fusion, Tensor Fusion 및 Layer fusion으로 모델 그래프 단순화



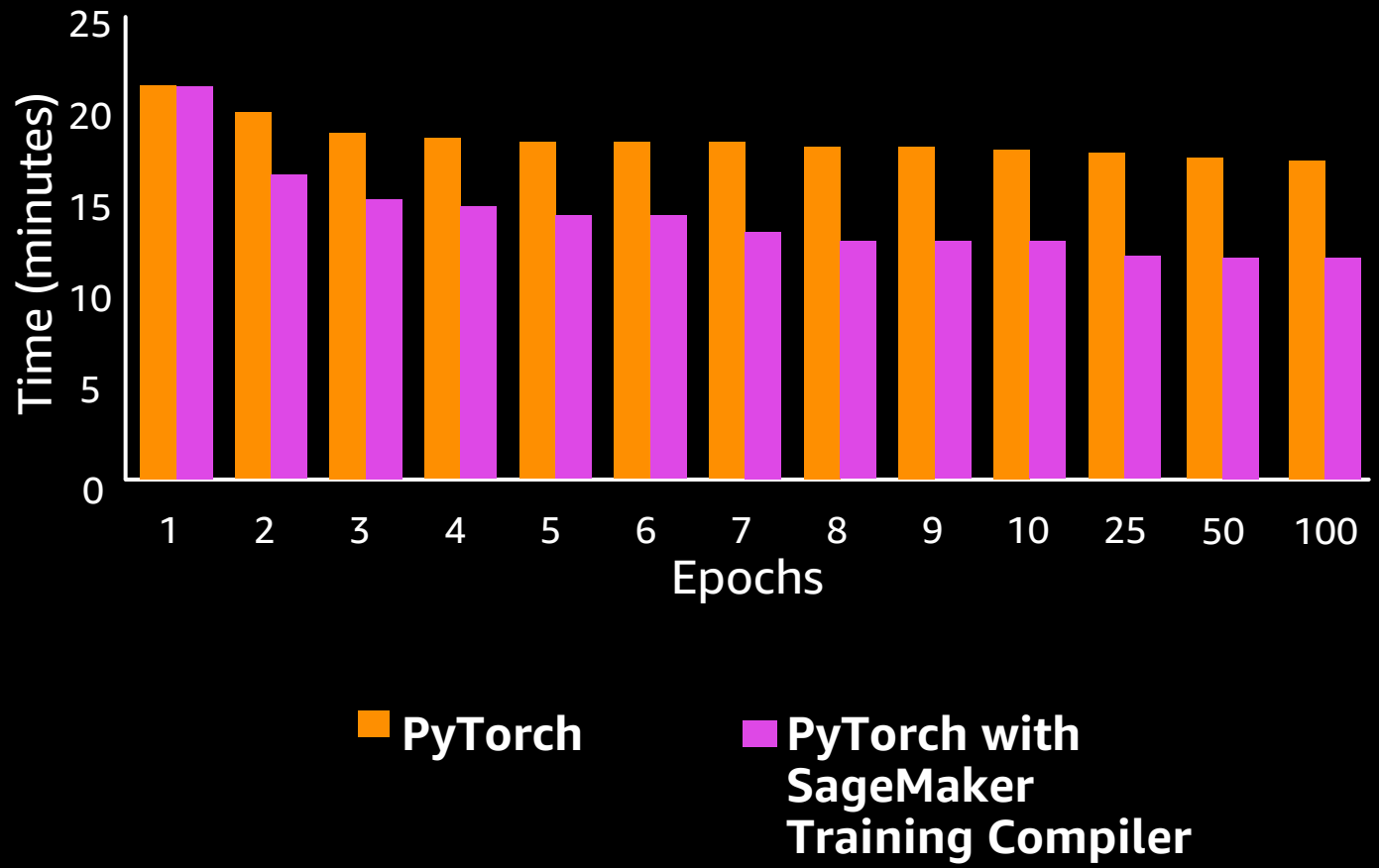
Source: https://blogs.nvidia.co.kr/2020/02/19/nvidia-tensor-rt/

수 분 내에 모델 컴파일

30분에 불과한 훈련 작업에도 RoBERTa 모델 훈련 시

27% 훈련 시간 단축

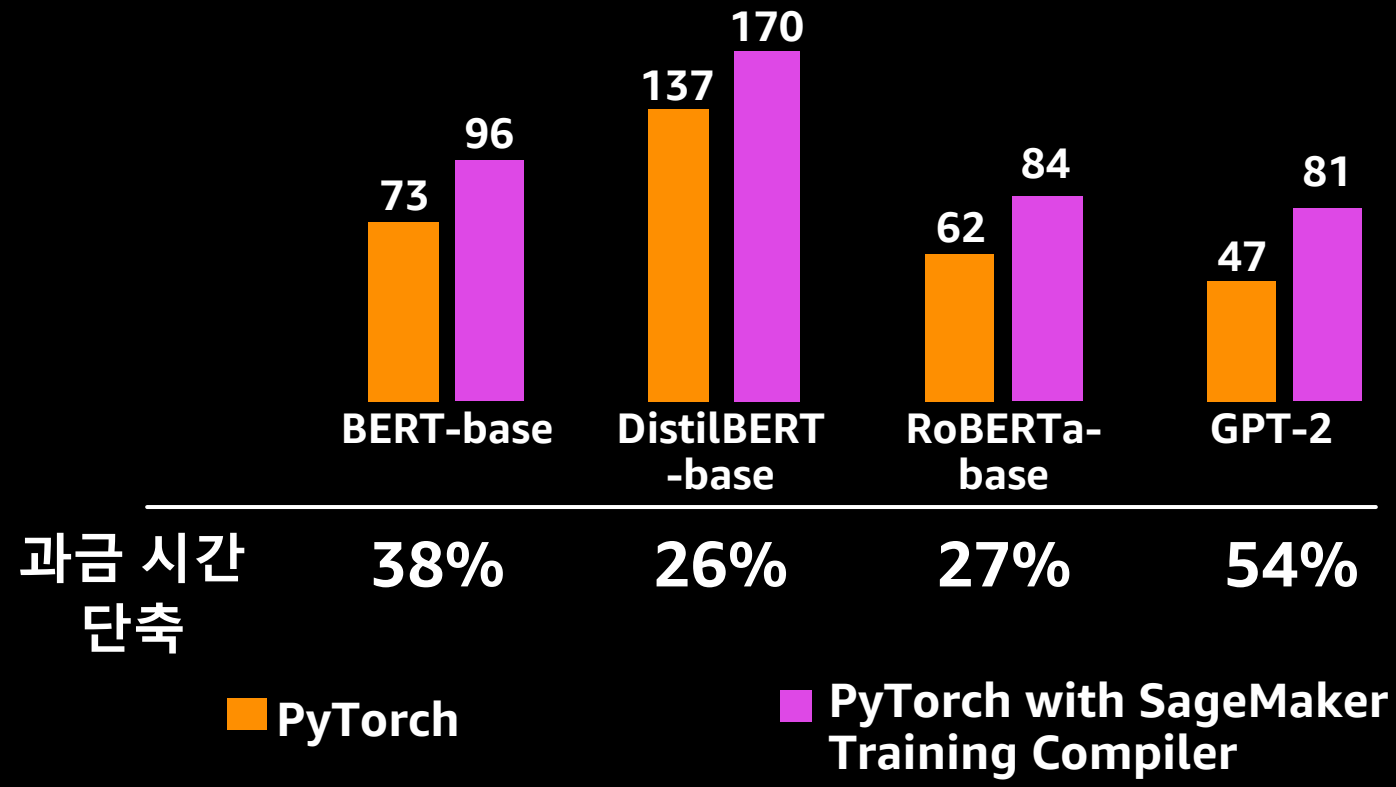
에폭Epoch 당 평균 훈련 시간 (RoBERTa-base with SST2 dataset)



다양한 모델 지원

50% 까지 훈련 가속화

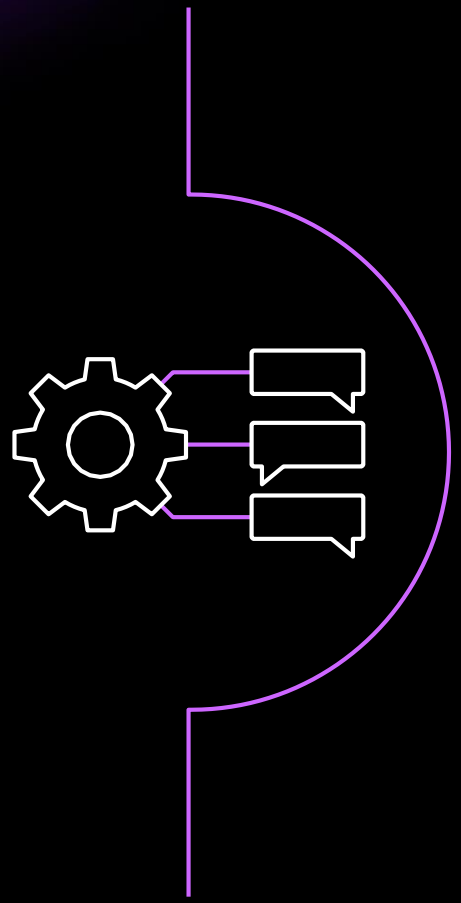
훈련 샘플 처리량[1] (samples/second)



[1] Test parameters: ml.p3.2xlarge, PyTorch with Hugging Face Trainer API, 25 epochs, sequence length of 512

Baseline used Hugging Face Deep Learning Container from ECR

SageMaker Training Compiler에서 검증된 NLP 모델[1)]



- bert-base-uncased
- bert-large-uncased
- roberta-base
- gpt2
- bert-base-cased
- xlm-roberta-base
- bert-base-chinese
- roberta-large
- distilbert-base-uncased
- distilbert-base-uncased-finetuned-sst-2-English
- cl-tohoku/bert-base-japanese-whole-word-masking
- bert-base-multilingual-cased
- distilgpt2
- albert-base-v2
- gpt2-large

1) ELECTRA를 비롯한 NLP 모델(한국어 모델 포함)을 기본적으로 모두 지원하지만, 최적의 하이퍼파라메터(batch size, learning rate 등)은 테스트가 필요합니다.

aws

SageMaker Training Compiler (SageMaker SDK)

```
batch_size_native = 32  # For vanilla language model
batch_size = 48         # For SageMaker training compiler
num_gpus = 8
learning_rate = learning_rate_native / batch_size_native *
batch_size * num_gpus

hyperparameters = {
    'training_script': 'train.py',
    'n_gpus': num_gpus, ...
}

hf_estimator = HuggingFace(
    entry_point          = 'src/launcher.py',
    compiler_config      = TrainingCompilerConfig(),
    disable_profiler     = True,
    debugger_hook_config = False,
    checkpoint_s3_uri    = chkpt_s3_path,
    checkpoint_local_path= '/opt/ml/checkpoints', ...
)
```

- 학습률learning rate 조절
- launcher.py 스크립트 생성 (분산 훈련에 공통적으로 쓰임)
- SageMaker Debugger 기능은 끄는 것을 권장
- 체크포인트 사용 권장
- Spot instance 옵션 병행 권장 (훈련 비용 대폭 절감)

Demo

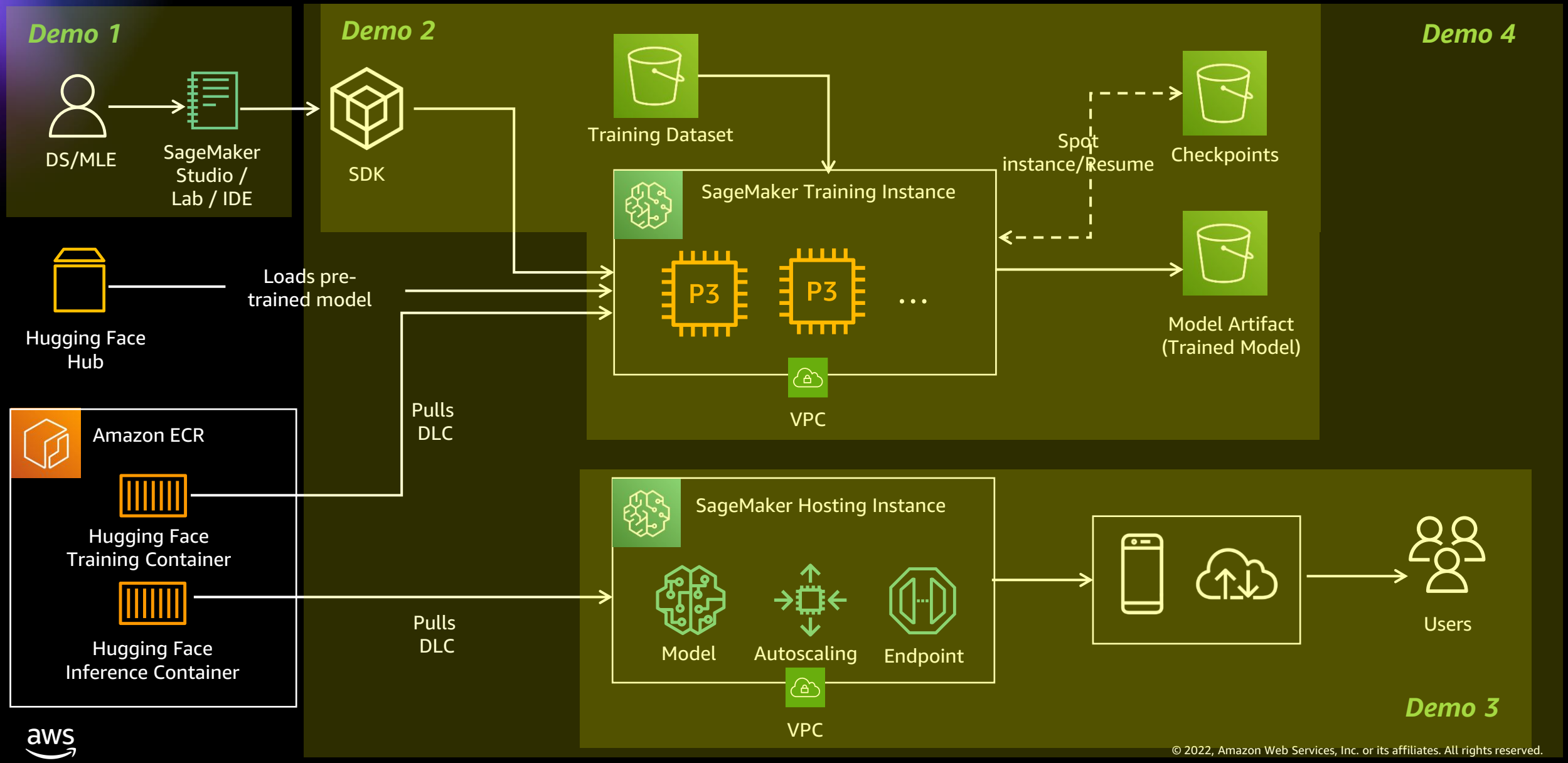
한국어 자연어 처리 모델 훈련, 배포 및 자동화

https://github.com/daekeun-ml/sm-huggingface-kornlp

References

- KoELECTRA: https://github.com/monologg/KoELECTRA
- Naver Sentiment Movie Corpus v1.0: https://github.com/e9t/nsmc
- Hugging Face examples: https://github.com/huggingface/notebooks/tree/master/sagemaker

End-to-end ML pipeline Demo



Resources

Hugging Face on AWS

개발자 가이드	https://docs.aws.amazon.com/sagemaker/latest/dg/hugging-face.html
AWS 안내 페이지	https://aws.amazon.com/ko/machine-learning/hugging-face/
Hugging Face 공식 튜토리얼	https://huggingface.co/docs/sagemaker/main

SageMaker Training Compiler

개발자 가이드	https://docs.aws.amazon.com/sagemaker/latest/dg/training-compiler.html
샘플 코드	https://github.com/aws/amazon-sagemaker-examples/tree/master/sagemaker-training-compiler/

Hands-on labs

한국어 자연어 처리 (본 강연의 데모)	https://github.com/daekeun-ml/sm-huggingface-kornlp
Hugging Face 공식 예제	https://github.com/huggingface/notebooks/tree/master/sagemaker

AI & ML 리소스 허브

AWS가 제공하는 AI 및 ML에 관한 다양한 자료들을 통해 더욱 심층적으로 학습해보세요!

- 기계 학습 여정 가이드
- 기계 학습의 7가지 주요 사용 사례
- 데이터, 분석 및 기계 학습을 위한 전략 플레이북
- 올바른 클라우드 서비스 및 인프라를 통한 기계 학습 혁신 가속화 전략 가이드
- 기계 학습에 적합한 컴퓨팅 인프라 선택 가이드
- 컨택트 센터의 서비스 개선 및 비용 절감 방법

- + 외의 다양한 동영상 학습 자료 및 기술 학습 자료!

https://bit.ly/3yUk0Kx

리소스 허브 방문하기

AWS Innovate - AI/ML 특집에 참석해주셔서 대단히 감사합니다.

저희가 준비한 강연, 어떻게 보셨나요?
더 나은 세미나를 위하여 **설문을 꼭 작성해 주세요!**


aws-korea-marketing@amazon.com


twitter.com/AWSKorea


facebook.com/amazonwebservices.ko


youtube.com/user/AWSKorea


linkedin.com/company/amazon-web-services


twitch.tv/aws

Thank you!