

TensorFlow 2.0으로 배우는 딥러닝 입문

tf.train.CheckpointManager API로 파라미터 저장하고 불러오기

에이아이스쿨(AISchool) 대표
양진호 (솔라리스)

<http://aischool.ai>

<http://solarisailab.com>

파라미터 저장과 복원의 필요성

- 예제 코드가 아닌 복잡한 실제 문제를 해결하기 위해서는 많은 횟수를 반복이 필요합니다.
- 따라서 중간 중간 내가 학습시킨 파라미터를 저장하고, 또한 저장된 파라미터를 통해서 이어서 학습을 진행하는 과정이 필수적입니다.
- TensorFlow 2.0에서는 파라미터 저장과 복원을 도와주는 **tf.train.CheckpointManager API**를 제공하고 있습니다.

tf.train.CheckpointManager API로 파라미터 저장하고 불러오기

- TensorFlow 2.0에서 파라미터를 저장하고 불러오는 방법은 다음과 같습니다.

[파라미터를 저장하는 법]

- ① `tf.train.Checkpoint` 클래스의 인자값으로 저장하고자 하는 `tf.keras.Model` 인스턴스와 전역 반복횟수를 지정해서 선언합니다.

e.g.) `ckpt = tf.train.Checkpoint(step=tf.Variable(0), model=CNN_model)`

- ② `tf.train.CheckpointManager`에 인자값으로 선언한 `tf.train.Checkpoint` 인스턴스와 중간 파라미터를 저장할 경로를 설정합니다.

e.g.) `ckpt_manager = tf.train.CheckpointManager(ckpt, directory=SAVER_DIR, max_to_keep=5)`

- ③ 파라미터를 저장하고자 하는 시점에 해당 시점의 전역 반복횟수를 인자값으로 선언한 `tf.train.CheckpointManager`의 `save` 메소드 호출합니다.

e.g.) `ckpt_manager.save(checkpoint_number=ckpt.step)`

- ④ `tf.train.Checkpoint`의 전역 반복 횟수 값(`ckpt.step`)을 매 반복마다 1씩 증가시킵니다.

e.g.) `ckpt.step.assign_add(1)`

tf.train.CheckpointManager API로 파라미터 저장하고 불러오기

- TensorFlow 2.0에서 파라미터를 저장하고 불러오는 방법은 다음과 같습니다.

[파라미터를 불러오는 법]

- ① `tf.train.latest_checkpoint`의 인자값으로 파라미터가 저장된 폴더 경로를 지정해서 가장 최근의 체크포인트 파일의 경로(full path)를 가져옵니다.

e.g.) `latest_ckpt = tf.train.latest_checkpoint(SAVER_DIR)`

- ② 선언한 `tf.train.CheckpointManager`의 `restore` 함수의 인자값으로 불러올 체크포인트 파일의 경로를 지정해서 파라미터 값을 복원합니다.

e.g.) `ckpt.restore(latest_ckpt)`

tf.train.CheckpointManager API로 파라미터 저장하고 불러오기

- tf.train.CheckpointManager API를 이용해서 중간 학습결과값을 저장하고 불러오는 방법을 살펴봅시다.
- https://github.com/solaris33/deep-learning-tensorflow-book-code/blob/master/Ch07-CNN/mnist_classification_using_cnn_with_tfsaver_v2_keras.py

Chapter 7 - Convolutional Neural Networks(CNN)

- CNN을 이용한 MNIST 숫자 분류기 구현 ([Code](#)) ([TF v2 Code](#)) ([TF v2 Keras Code](#))
- CNN을 이용한 CIFAR-10 분류기 구현 ([Code](#)) ([TF v2 Code](#)) ([TF v2 Keras Code](#))
- tf.train.Saver API를 이용해서 모델과 파라미터를 저장(Save)하고 불러오기(Restore) ([Code](#)) ([TF v2 Code](#)) ([TF v2 Keras Code](#))

Chapter 8 - Recurrent Neural Networks(RNN)

- tf.nn.embedding_lookup을 이용한 Embedding 예제 ([Code](#))
- Gradient Clipping 예제 ([Code](#)) ([TF v2 Keras Code](#))
- Char-RNN을 이용한 텍스트 생성 ([Code](#)) ([TF v2 Keras Code](#))

Thank you!
