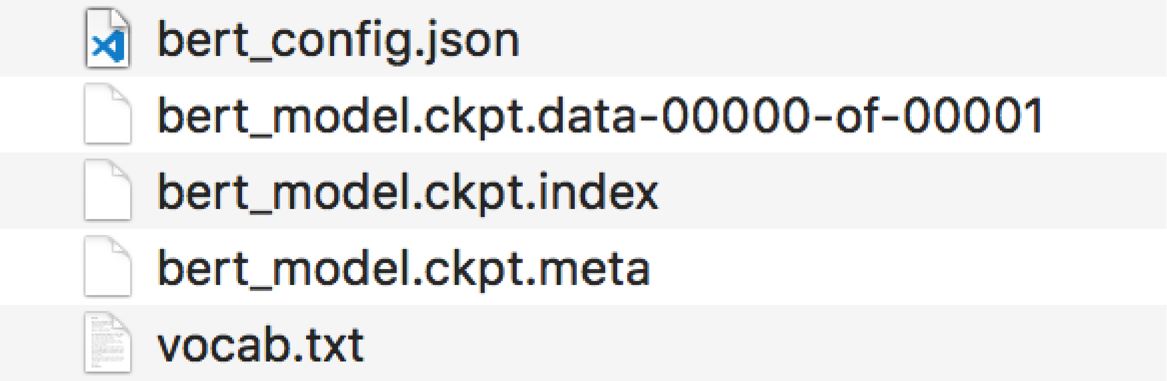
BERT使用问题汇总

使用BERT预训练产出Checkpoint，指定目录报错:”Couldn’t find ‘checkpoint’ file or checkpoints in given directory”。

问题原因



bert_config.json
bert_model.ckpt.data-00000-of-00001
bert_model.ckpt.index
bert_model.ckpt.meta
vocab.txt

我们先看一下Tensorflow加载Checkpoint的源代码:

checkpoint_utils.py文件:

```
def load_checkpoint(filepattern):
  filename = _get_checkpoint_filename(filepattern)
  if filename is None:
    raise ValueError("Couldn't find 'checkpoint' file or checkpoints in "
                     "given directory %s" % filepattern)
  return train.NewCheckpointReader(filename)

def _get_checkpoint_filename(filepattern):
  if gfile.IsDirectory(filepattern):
    return saver.latest_checkpoint(filepattern)
  return filepattern
```

tensorflow/python/training/saver.py文件:

```
latest_checkpoint = checkpoint_management.latest_checkpoint
```

tensorflow/python/training/checkpoint_management.py文件:

```
@tf_export("train.latest_checkpoint")
def latest_checkpoint(checkpoint_dir, latest_filename=None):
  ckpt = get_checkpoint_state(checkpoint_dir, latest_filename)
  if ckpt and ckpt.model_checkpoint_path:
    # Look for either a V2 path or a V1 path, with priority for V2.
    v2_path = _prefix_to_checkpoint_path(ckpt.model_checkpoint_path,
                                         saver_pb2.SaverDef.V2)
    v1_path = _prefix_to_checkpoint_path(ckpt.model_checkpoint_path,
                                         saver_pb2.SaverDef.V1)
    if file_io.get_matching_files(v2_path) or file_io.get_matching_files(
        v1_path):
      return ckpt.model_checkpoint_path
    else:
      logging.error("Couldn't match files for checkpoint %s",
                    ckpt.model_checkpoint_path)
  return None

@tf_export("train.get_checkpoint_state")
def get_checkpoint_state(checkpoint_dir, latest_filename=None):
  ckpt = None
  coord_checkpoint_filename = _GetCheckpointFilename(checkpoint_dir,
                                                     latest_filename)
  f = None
  try:
    # Check that the file exists before opening it to avoid
    # many lines of errors from colossus in the logs.
    if file_io.file_exists(coord_checkpoint_filename):
      file_content = file_io.read_file_to_string(
          coord_checkpoint_filename)
      ckpt = CheckpointState()
      text_format.Merge(file_content, ckpt)
      if not ckpt.model_checkpoint_path:
        raise ValueError("Invalid checkpoint state loaded from "
                         + checkpoint_dir)
      # For relative model_checkpoint_path and all_model_checkpoint_paths,
      # prepend checkpoint_dir.
      if not os.path.isabs(ckpt.model_checkpoint_path):
        ckpt.model_checkpoint_path = os.path.join(checkpoint_dir,
                                                  ckpt.model_checkpoint_path)
      for i in range(len(ckpt.all_model_checkpoint_paths)):
        p = ckpt.all_model_checkpoint_paths[i]
        if not os.path.isabs(p):
          ckpt.all_model_checkpoint_paths[i] = os.path.join(checkpoint_dir, p)
  except errors.OpError as e:
    # It's ok if the file cannot be read
    logging.warning("%s: %s", type(e).__name__, e)
    logging.warning("%s: Checkpoint ignored", coord_checkpoint_filename)
    return None
  except text_format.ParseError as e:
    logging.warning("%s: %s", type(e).__name__, e)
    logging.warning("%s: Checkpoint ignored", coord_checkpoint_filename)
    return None
  finally:
    if f:
      f.close()
  return ckpt

def _GetCheckpointFilename(save_dir, latest_filename):
  if latest_filename is None:
    latest_filename = "checkpoint"
  return os.path.join(save_dir, latest_filename)
```

我们可看到，代码会检测指定目录下是否存在checkpoint这个文件，很显然从上面的图片我们可以看到这个文件在BERT的checkpoint下是不存在的，所以get_checkpoint_state就会返回None，进而latest_checkpoint返回None，因此就导致抛出了我们上面提到的异常。

解决方法

将指定的checkpoint的路径改为 存储路径/bert_model.ckpt即可，这样就会由tf.train.NewCheckpointReader进行变量读取，不过NewCheckpointReader函数的实现并没有找到，估计是在C或者C++代码中实现的。