Trans\_Fomerの使い方

古林　実

ここでは、以下のWEBディレクトリのソフトウェアを利用します。

<https://github.com/minoru0419/TAL/tree/main/actionformer>

まずUbuntuマシンと処理したい動画を用意します。動画は多ければ多いほど良いです。

Ubuntuマシンには高性能NVIDIA社GPUが装備されていて、CUDAのインストール完了済みが前提です。

次に普通にWindowsでアノテーションします。ラベルも決めてセーブしてください。

動画を正規化します。画面が大きかったりFPSが高かったりすると今後の作業に支障があります。

以下のコマンドで正規化してください。ffmpegが必要です。結果はconvというディレクトリに格納されます。

for f in \*.mp4; do ffmpeg -i "$f" -r 30 -vf "scale=640:-1" -qmin 1 "./conv/${f%.mp4}.mp4"; done

正規化した動画のI3D特徴量抽出を行います。変数の横幅は特徴量RGB1024次元、フロー1024次元、合わせて2048次元となります。縦はフレーム数となります。

以下のリポジトリを利用します。

<https://github.com/v-iashin/video_features>

README.mdに従ってミニコンダで環境を整えてください。

専用のパイソンプログラムを作ってあります。必ずフローはRAFTではなくOpencvを使ってください。RAFTを使うと結果が出るのが遅くなり評価が下がって職を失うことになりかねません。それくらい遅いです。

video\_featuresのディレクトリに正規化した動画を配置してその環境で「calcu\_i3d\_opencv.py」を実行します。

それぞれの動画に対応したnpyファイルが作成されます。

次にTransFormer本体を利用します。リポジトリは以下となります。

<https://github.com/happyharrycn/actionformer_release>

README.mdに従ってミニコンダで環境を整えてください。

./dataというディレクトリがありますので以下のような感じでディレクトリを作って内容をコピーしてください。

data/next/annotations : アノテーションJSONファイルを入れる。label\_define.json込。

data/next/i3d\_features : 「calcu\_i3d\_opencv.py」 で作ったI3D特徴量npyファイルを入れる。

data/next/movies : 正規化した動画を入れる

データのセットが終わったら、「convert\_json.py」を起動してVP-MotionのアノテーションをTransFormer用のアノテーションに変換します。「convert\_json.py」はdata/nextに入れておきます。

「next\_all.json」というファイルが生成されます。これがTransFormer用のアノテーションファイルです。

一番最初に示したWEBディレクトリから「next\_i3d.yaml」をconfigディレクトリに入れておきます。

次に学習します。以下のようにタイプしてください。

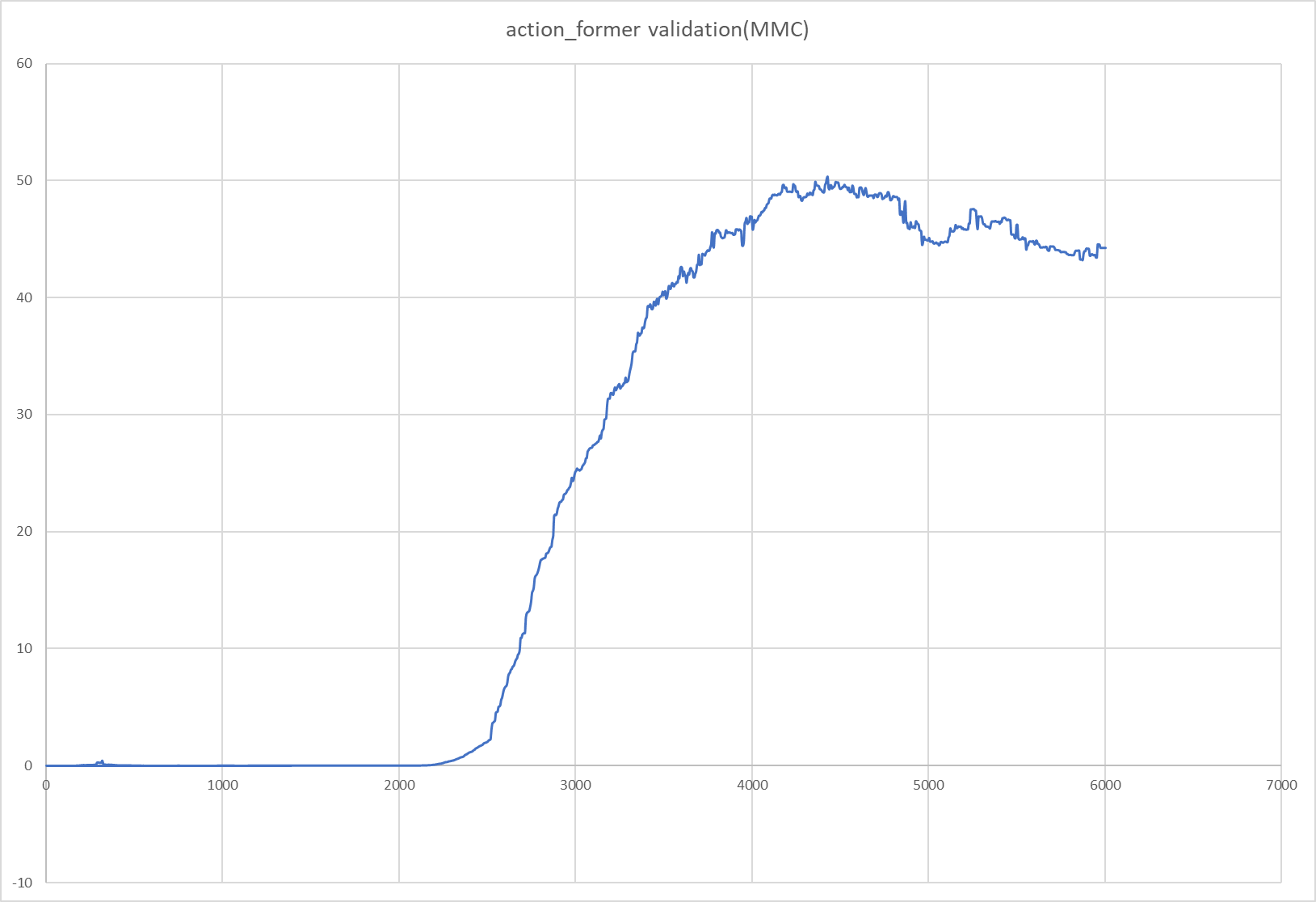
python ./train.py ./configs/thumos\_i3d.yaml --output reproduce

6000イテレーションを設定しています。必要であれば「next\_i3d.yaml」ファイルを変更してください。但しバッチサイズは変えないでください。学習せずに空回りする可能性があります。

次に6000イテレーションの中からベストポイントを検索します。「eval\_search.py」を実行してください。

ベストポイントを計算してくれます。このプログラムは「evaluate\_result.csv」というファイルを生成するので学習の履歴をエクセルの表で確認することができます。

＜表示例＞



ベストポイントが見つかったらその番号のチェックポイントファイルを利用してTransFormerを動作させます。

「eval\_get\_result.py」を動作させてaction\_formerの最終出力を出します。

その前に「eval\_get\_result.py」189行目に即値が入っていますので処理したい動画の名前に変更してください。

if result\_list[i][0] == "動画3":

python eval\_get\_result.py ./configs/next\_i3d.yaml ./ckpt/next\_i3d\_reproduce/epoch\_xxxx.pth.tar

xxxxのところはベストポイントに置き換えてください。

すると動画が表示されて結果がオンスクリーンで表示されます。

時間的行動定位(Temporal Action Localization)の体験はいかがだったでしょうか？実は「eval\_get\_result.py」は動画のI3Dを与えると1.26秒で動作の開始時間と終了時間とフラグとスコアを出してきます。動画が表示しているときには全てが終わっているのです。今回は時間が長いオブジェクトを優先するように「eval\_get\_result.py」を作りましたが、スコア優先もできるかもしれません。

それではごきげんよう。