Efficient Temporal Action Localization model development practices

A review and analysis of models and a guide of best methods based on a study case

Author: Paul Misterka - p.m.misterka@student.tudelft.nl
Supervisors: Attila Lengyel, Robert-Jan Bruintjes, Ombretta Strafforello
Responsible Professor: Jan van Gemert

1. Background

Temporal Action Localization (TAL) *[1]* is the task of recognizing actions in video segments and tagging their start and end

- Computationally expensive
- Requires large amounts of training data

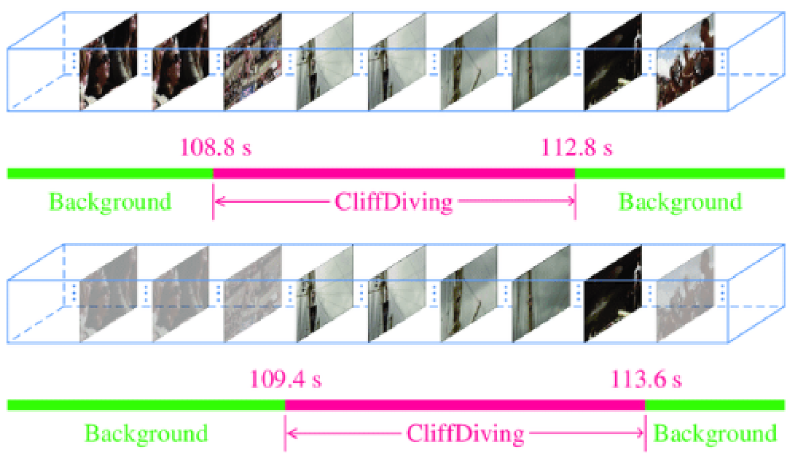


Figure 1: Principle of Temporal Action Localization

Training TAL models is hence difficult, and requires a huge amount of resources.

2. Problem

We aim to **accelerate** the development of novel TAL methods by allowing researchers to experiment and test ideas quicker by:

- Making the code run **efficiently**
- **Extrapolating** results from partial data
- Using our **guidelines** to develop the codebase effectively
- Traininig faster to **iterate** ideas faster

3. Methodology

Taking increasing parts of the THUMOS14 *[2]* dataset, we measure resulting model's *mAP (mean avg. prec.)*.

Algorithm 1 Main procedure

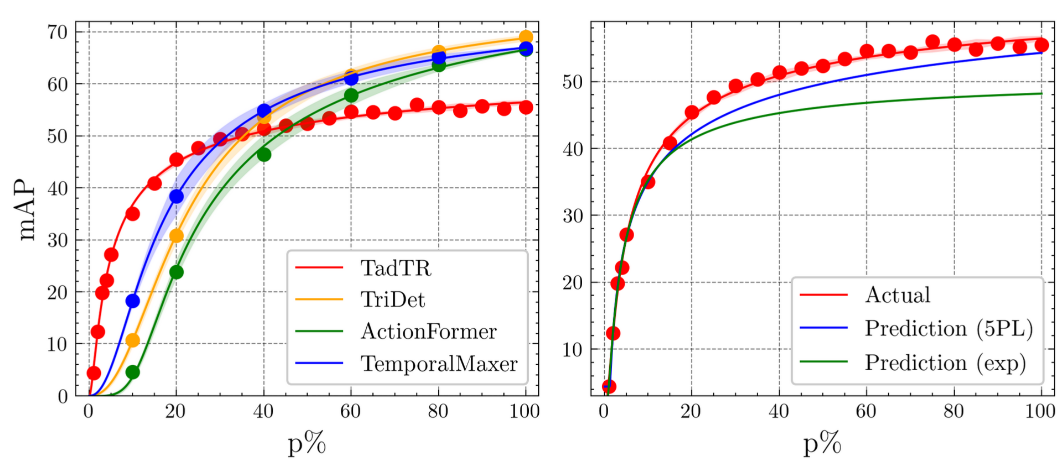
$\mathcal{D} = \{(\mathbf{V}_i, \mathbf{y}_i)\}_{i=1}^{N}$ — $\mathcal{D}$ is the whole dataset
$\mathcal{D}_{train}, \mathcal{D}_{test} \leftarrow \texttt{split}(\mathcal{D}, 0.2)$ — Splits s.t. $\frac{|\mathcal{D}_{train}|}{|\mathcal{D}|} = 0.8$, $\frac{|\mathcal{D}_{test}|}{|\mathcal{D}|} = 0.2$
for $p = 10\%, ..., 100\%$ **do**
  $mAPs \leftarrow \emptyset$
  for $i = 1, ..., 5$ **do**
    $\mathcal{D}_s \leftarrow \texttt{sample}(\mathcal{D}_{train}, p)$ — Samples s.t. $\frac{|\mathcal{D}_s|}{|\mathcal{D}_{train}|} \cdot 100\% = p$
    Train on $\mathcal{D}_s$
    $mAP \leftarrow \texttt{calculate-mAP}(\mathcal{D}_{test})$ — Calculate mAP@tIoU[0.3:0.1:0.7]
    $mAPs \leftarrow mAPs \cup \{mAP\}$
  Report $\texttt{avg}(mAPs)$ and $\texttt{std}(mAPs)$

Figure 2: Limited data setting training procedure

We perform a theoretical and real-world compute performance of our study case, **TadTR** *[3]*, and compare it to other TAL models.

4. Experiments

We find that the p% (% of the dataset) vs mAP curve has an unusual sigmoidal, but **consistent shape** for all tested models, and attempt different fits against it.



Figures 3, 4: Data efficiency of TadTR and other models; p% is the percentage of the dataset used (100% is 212 videos, THUMOS14).

References

[1] Le Wang, Xuhuan Duan, Qilin Zhang, Zhenxing Id, Gang Hua, and Nanning Zheng. Segment-Tube: Spatio-Temporal Action Localization in Untrimmed Videos with Per-Frame Segmentation. Sensors, 18, 05 2018
[2] Y.-G. Jiang, J. Liu, A. Roshan Zamir, G. Toderici, I. Laptev, M. Shah, and R. Sukthankar. THUMOS challenge: Action recognition with a large number of classes. http://crcv.ucf.edu/THUMOS14/, 2014.
[3] Xiaolong Liu, Qimeng Wang, Yao Hu, Xu Tang, Shiwei Zhang, Song Bai, and Xiang Bai. End-to-end Temporal Action Detection with Transformer. IEEE Transactions on Image Processing (TIP), 2022

5. Conclusions

Our conclusions provide guidelines on how to:

- Pick the right amount of data
- Efficiently train TAL models
- Extrapolate relationships in ML models
- Build the codebase in a way that promotes quick experimentation without sacrificing speed

Moreover, most compared TAL do not saturate and would **benefit from more data** (+9% mAP at x3 size). We also evaluate compute performance and provide training and inference times, theoretical computer performance, and a hardware usage study.

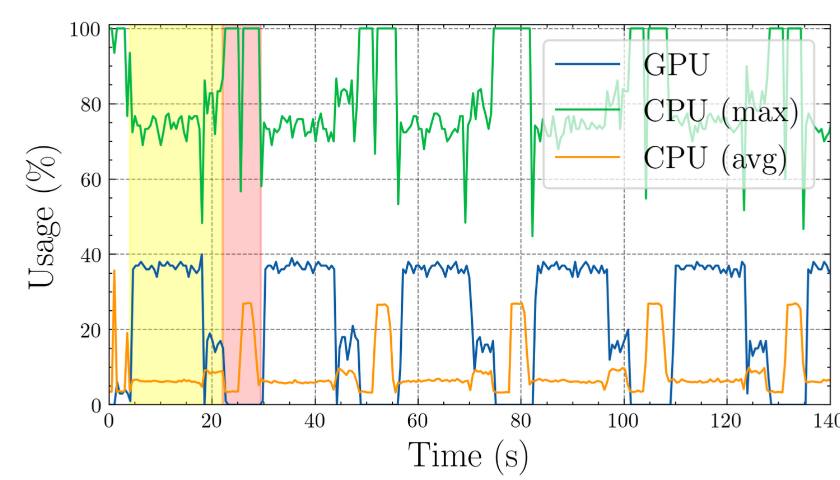


Figure 5: TadTR's hardware usage profile. The model is largely bottlenecked by CPU and memory bandwidth.