PREGO: online mistake detection in PRocedural EGOcentric videos -Supplementary Materials-

Alessandro Flaborea*[♦♣] Guido Maria D'Amely di Melendugno*[♦] Leonardo Plini[♦] Luca Scofano[♦]
Edoardo De Matteis[♦] Antonino Furnari[♠] Giovanni Maria Farinella[‡♠] Fabio Galasso[‡♦]
{flaborea,damely,dematteis,galasso}@di.uniroma1.it {plini,scofano}@diag.uniroma1.it
{antonino.furnari,giovanni.farinella}@unict.it

[♦]Sapienza University of Rome, Italy [♣]ItalAI S.r.l. [♠]University of Catania, Italy

This section provides a detailed analysis of the step recognition and anticipation branches. Moreover, we conduct an ablation study on the impact of different prompt structures on the performance of the Large Language Model for mistake detection. Lastly, we analyze the split of Epic-tent-O. The code is available at https://github.com/aleflabo/PREGO.

A. Modelling Details

In this section, we discuss the step recognition architecture, delve into details on symbolic reasoning, and provide insights into the hyperparameters used.

A.1. Step Recognition

As shown in Table 2 of the main paper, we evaluate PREGO using two methods for the step recognition branch: OadTR [5] and [1], adapted for the task of online mistake detection. This section discusses the OadTR architecture, which achieves the best results on the Epic-tent-O dataset.

OadTR comprises a Transformer Encoder with three encoding layers. Each layer incorporates a Multi-Head self-attention module and an Element-Wise addition module. Similarly to [5], PREGO retains the learnable task token along with the frame features, which acquire global discriminative features for the online action detection task. The OadTR-based step recognition model is selected according to its recognition performance. In Tab. 1, we compare two alternatives, which we consider: the original encoder-decoder OadTR [5] model, and an encoder-only variant, which we propose for PREGO. Note that performance is evaluated on the Assembly101-O target benchmark, so the reported estimates differ from what reported when evaluated on [3]. The encoder-only model outperforms the original OadTR architecture for all the window lengths, so we selected it for PREGO to yield action recognition online. The optimal window size, set at 512, demonstrates a 10% improvement over OadTR. Further to performance, the encoder-only model also significantly reduces the parameter count compared to the original model, reducing it approximately three times in size.

Table 1. mAP performance of OadTR [5] as the Step Recognition method considering different window sizes and architectures on Assembly101-O.

	Parms.	**Runtime** (*sec*)	**Window size**				
			64	**128**	**256**	**512**	**768**
Encoder-Decoder (*OadTR*)	74 M	0.031	11.0	12.1	12.7	13.0	12.2
Encoder *(PREGO)*	21 M	0.017	11.3	12.3	12.8	**14.5**	13.2

In Table 1, for the encoder-decoder (OadTR) and encoder-only (PREGO) models, we compare performances achieved by varying the window size, i.e. varying the length in frames of the ingested video excerpt, as input for the recognition model. The mAP exhibits an ascending trend with increasing window sizes, reaching its maximum when the window size equals 512. Beyond this point, the mAP decreases, suggesting that there is saturation and that the model fails to handle the long-term dependencies between frames.

The inference Runtime, computed on a single sample, shows the advantage of using the PREGO architecture due to the strict time requirements of the online setup. The chosen architecture is approximately 45% faster than the original model.

A.2. Symbolic Reasoning

The main paper shows two distinct setups for the step anticipation branch of PREGO, i.e., one using Llama-2 and one adopting the OpenAI GPT-3.5. Specifically, we employ the 7 billion parameters version of Llama-2 [4] as the LLM module for symbolic reasoning. We adjust the temperature and output tokens hyperparameters to 0.6 for the former, which aims to enforce quasi-deterministic outputs, while setting the latter to 4 to ensure answers of the desired length.

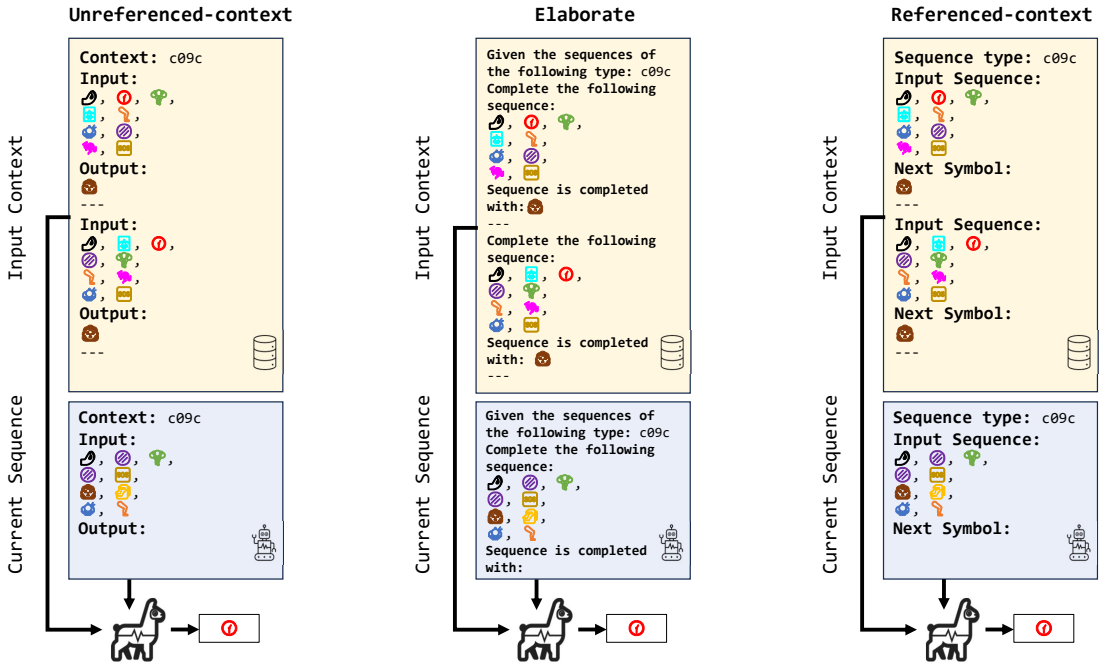

Figure 1. Three different variants, defining different inputs to the LLMs. On the left, the prompt lacks any reference to sequences or symbols to be completed. In the center, the prompt consists of detailed and lengthier requests. On the right, the prompt incorporates the context of the sequence explicitly. This third variant performs best and it is therefore adopted in PREGO.

The second selected LLM is GPT-3.5 [2], developed by OpenAI, provided as a paid API service. It has been trained with reinforcement learning employing reward models learned by human feedback [2]. In our experiments, we fix the temperature to 0.0. Unlike Llama-2, we do not constrain this model on the output length. This is because, in our experiments, GPT-3.5 was more likely to give answers consistent with the form of the prompt when compared with Llama-2.

B. Prompt Context

In Sec. 5.4 of the main paper, we discussed the performance of the Step Anticipation branch using different prompts. In Fig. 1, we show the three prompts, dubbed "Unreferenced-context", "Elaborate", and "Referenced-context". As reported in the main paper, the results of PREGO with the three versions are similar, hence the symbolic reasoning of the LLM is not affected by the input prompts.

C. Epic-tent-O split

As described in Sec. 3.1.2 of the main paper, we propose a new split of the Epic-tent dataset. We opt to include this dataset since it includes different types of procedural mistakes, i.e. ordering, omissions, repetitions, and corrections. It has been recorded open-air, differentiating it from the assembly and kitchen-based datasets in literature. Epic-tent consists of 29 videos, all of which contain annotated procedural errors. To follow the OCC paradigm, we split the dataset according to the confidence the actors annotated while performing the procedure, as shown in Fig. 2. The videos that form the test set have a median confidence score under 0.6, while the others form the train set. Only 22 videos

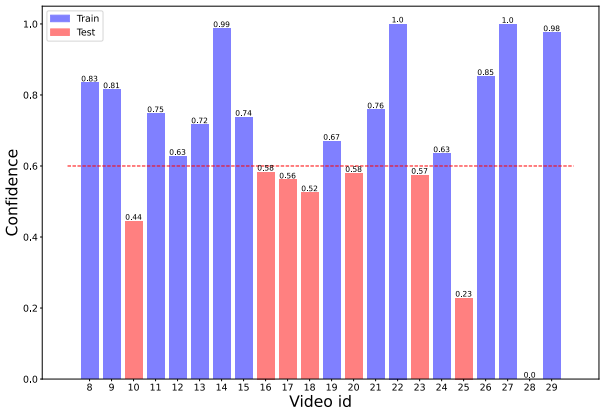

Figure 2. Epic-tent-O split between train and test set based on the self-confidence of actors while performing the procedure. The videos with id between $[1, 7]$ do not have confidence score annotations and are included in the test set.

have the confidence score annotations, while the remaining 7 do not and are assigned to the test set.

References

[1] Joungbin An, Hyolim Kang, Su Ho Han, Ming-Hsuan Yang, and Seon Joo Kim. Miniroad: Minimal rnn framework for online action detection. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10307–10316, 2023. 1

[2] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022. 2

[3] F. Sener, D. Chatterjee, D. Shelepov, K. He, D. Singhania, R. Wang, and A. Yao. Assembly101: A large-scale multi-view video dataset for understanding procedural activities. *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 1

[4] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 1

[5] Xiang Wang, Shiwei Zhang, Zhiwu Qing, Yuanjie Shao, Zhengrong Zuo, Changxin Gao, and Nong Sang. Oadtr: Online action detection with transformers. In *Int. Conf. Comput. Vis.*, pages 7565–7575, 2021. 1