Structured Context Learning for Generic Event Boundary Detection

In this supplementary material, we provide additional ablation results to further validate the effectiveness of our method.

1. Additional Ablation Experiments

A. Effect of temporal model. The design of the Structured Partition of Sequence (SPoS) allows for flexibility in choosing different temporal models to achieve a better speed-accuracy trade-off. In addition to Transformers, we found that Recurrent Neural Networks (RNNs) also achieve good performance. As shown in Table 1a, LSTM and GRU achieve competitive results and run faster than Transformers. We infer that SPoS provides local structured context information for boundary detection, which is critical for accurate prediction, and the subsequent temporal model is responsible for semantic learning. As a result, different temporal models have only a minor influence on the final performance. This confirms the effectiveness of SPoS and provides more options for selecting temporal models.

B. Effect of loss function. The GEBD task can be interpreted as a binary classification task at the frame level, where the goal is to classify each frame as a boundary or non-boundary after capturing temporal context information. To train our model, we use binary cross-entropy (BCE) loss and mean squared error (MSE) loss, with the option of turning on or off Gaussian smoothing (as introduced in section 3.3). As shown in Table 1b, we observe that Gaussian smoothing can improve performance in both settings, indicating its effectiveness. We attribute this improvement to two factors: 1) Consecutive frames often have similar feature representations in the latent space, leading to a tendency for consecutive frames to output similar responses. Hard labels violate this tendency and may lead to poor convergence. 2) The annotations for GEBD are inherently ambiguous, and Gaussian smoothing can prevent the network from becoming overconfident. For all our experiments, we use the "BCE + Gaussian" setting.

C. Effect of similarity function. We investigate the impact of different distance metrics on our proposed method, which we refer to as similarity metrics since we used negative values to calculate them. Specifically, we evaluated four commonly used distance metrics, namely Cosine, Euclidean, Manhattan, and Chebyshev. It should be noted that each distance metric has its unique properties that can influence the performance of the model. For example, the Euclidean distance is sensitive to outliers, whereas the cosine distance is not. The Manhattan distance is more suitable for measuring the distance between two points in a grid-like structure. The experimental results, presented in Table 1c, demonstrate that our method is effective with all four metrics. However, we choose the Cosine metric for our experiments due to its better performance compared to the other metrics.

Temporal model	0.05	0.25	0.5	avg	speed(ms)
Transformer	**0.784**	**0.896**	**0.911**	**0.883**	**1.9**
LSTM	0.772	0.893	0.909	0.879	1.1
GRU	0.773	0.894	0.909	0.880	1.0

(a) Effect of temporal model.

BCE	MSE	Gaussian	0.05	0.25	0.5	avg
	✓		0.762	0.885	0.901	0.869
	✓	✓	0.775	0.894	0.909	0.880
✓			0.773	0.892	0.907	0.878
✓		✓	**0.784**	**0.896**	**0.911**	**0.883**

(b) Effect of loss function.

Similarity Function	0.05	0.25	0.5	avg
Chebyshev	0.773	0.889	0.906	0.877
Manhattan	0.781	0.894	0.908	0.880
Euclidean	0.783	0.895	0.910	0.882
Cosine	**0.784**	**0.896**	**0.911**	**0.883**

(c) Effect of similarity-function$(\cdot, \cdot)$ in Equation (3)

Table 1. Our Structured Context Learning method ablation experiments on Kinetics-GEBD validation dataset. We report F1 scores for Rel.Dis. thresholds of 0.05, 0.25, and 0.5. The "avg" column represents the average F1 score across all Rel.Dis. thresholds ranging from 0.05 to 0.5, with an interval of 0.05. Default settings are marked in gray.

D. Detailed results on Kinetics-GEBD. Table 2, Table 3 and Table 4 for Kinetics-GEBD respectively present the detailed results of precision, recall and f1 scores for various methods. It is noteworthy that some methods like TCN

Rel.Dis. threshold	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	avg
BMN [3]	0.128	0.141	0.148	0.152	0.156	0.159	0.162	0.164	0.165	0.167	0.154
BMN-StartEnd [3]	0.396	0.479	0.509	0.525	0.534	0.540	0.544	0.547	0.549	0.551	0.517
TCN-TAPOS [2]	0.518	0.622	0.665	0.690	0.706	0.718	0.727	0.733	0.738	0.743	0.686
TCN [2]	0.461	0.519	0.538	0.547	0.553	0.557	0.559	0.561	0.563	0.564	0.542
PC [4]	0.624	0.753	0.794	0.816	0.828	0.836	0.841	0.844	0.846	0.849	0.803
DDM-Net [5]	0.732	0.812	0.836	0.849	0.856	0.860	0.863	0.865	0.867	0.869	0.841
Ours	**0.754**	**0.831**	**0.850**	**0.862**	**0.868**	**0.873**	**0.876**	**0.878**	**0.880**	**0.883**	**0.855**

Table 2. **Precision** on Kinetics-GEBD validation split with Rel.Dis. threshold set from 0.05 to 0.5 with 0.05 interval.

Rel.Dis. threshold	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	avg
BMN [3]	0.338	0.369	0.385	0.397	0.407	0.414	0.420	0.426	0.430	0.434	0.402
BMN-StartEnd [3]	0.648	0.766	0.817	0.846	0.864	0.876	0.885	0.892	0.897	0.900	0.839
TCN-TAPOS [2]	0.420	0.508	0.550	0.576	0.594	0.609	0.619	0.627	0.633	0.639	0.577
TCN [2]	0.811	0.894	0.923	0.938	0.947	0.952	0.956	0.959	0.961	0.963	0.930
PC [4]	0.626	0.764	0.814	0.843	0.859	0.871	0.879	0.885	0.889	0.892	0.832
DDM-Net [5]	0.800	0.875	0.899	0.912	0.920	0.926	0.930	0.933	0.935	0.937	0.907
Ours	**0.816**	**0.882**	**0.906**	**0.920**	**0.926**	**0.932**	**0.935**	**0.937**	**0.940**	**0.941**	**0.913**

Table 3. **Recall** on Kinetics-GEBD validation split with Rel.Dis. threshold set from 0.05 to 0.5 with 0.05 interval.

Rel.Dis. threshold	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	avg
BMN [3]	0.186	0.204	0.213	0.220	0.226	0.230	0.233	0.237	0.239	0.241	0.223
BMN-StartEnd [3]	0.491	0.589	0.627	0.648	0.660	0.668	0.674	0.678	0.681	0.683	0.640
TCN-TAPOS [2]	0.464	0.560	0.602	0.628	0.645	0.659	0.669	0.676	0.682	0.687	0.627
TCN [2]	0.588	0.657	0.679	0.691	0.698	0.703	0.706	0.708	0.710	0.712	0.685
PC [4]	0.625	0.758	0.804	0.829	0.844	0.853	0.859	0.864	0.867	0.870	0.817
SBoCo-Res50 [1]	0.732	-	-	-	-	-	-	-	-	-	0.866
DDM-Net [5]	0.764	0.843	0.866	0.880	0.887	0.892	0.895	0.898	0.900	0.902	0.873
Ours	**0.784**	**0.856**	**0.877**	**0.890**	**0.896**	**0.901**	**0.904**	**0.907**	**0.909**	**0.911**	**0.883**

Table 4. **F1** results on Kinetics-GEBD validation split with Rel.Dis. threshold set from 0.05 to 0.5 with 0.05 interval.

achieves high recall yet low precision because they make as many predictions as possible and recall many false positives. As a result, they do not achieve superior F1 score.

References

[1] Hyolim Kang, Jinwoo Kim, Taehyun Kim, and Seon Joo Kim. Uboco : Unsupervised boundary contrastive learning for generic event boundary detection. *CoRR*, abs/2111.14799, 2021. 2

[2] Colin Lea, Austin Reiter, René Vidal, and Gregory D. Hager. Segmental spatiotemporal cnns for fine-grained action segmentation. In *ECCV*, pages 36–52, 2016. 2

[3] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. BMN: boundary-matching network for temporal action proposal generation. In *ICCV*, pages 3888–3897. IEEE, 2019. 2

[4] Mike Zheng Shou, Deepti Ghadiyaram, Weiyao Wang, and Matt Feiszli. Generic event boundary detection: A benchmark for event segmentation. *CoRR*, abs/2101.10511, 2021. 2

[5] Jiaqi Tang, Zhaoyang Liu, Chen Qian, Wayne Wu, and Limin Wang. Progressive attention on multi-level dense difference maps for generic event boundary detection. In *CVPR*, pages 3345–3354. IEEE, 2022. 2