

# Supplemental Material for SynopGround: A Large-Scale Dataset for Multi-Paragraph Video Grounding from TV Dramas and Synopses

## 1 Supplementary Material

In this supplementary material, we provide more information including additional experimental results, more implementation details and qualitative analysis based on visualization.

### 1.1 Additional Experimental Results

In this part, we present some additional comparison and ablation results, to further investigate and validate the effectiveness of our proposed designs for the baseline model LGMR.

**Comparison on MAD dataset.** In Table S1, we show the comparison results with other state-of-the-art methods on the 3-min version of MAD dataset, following the setting in [S7], to verify the superiority of our proposed LGMR method over existing baselines. As can be seen, our method consistently outperforms state-of-the-art multi-query methods in all metrics by a large margin.

**Table S1: Performance comparison on the MAD dataset.**

Method	Query Input	R@ 0.1	R@0.3	R@0.5	mIoU
DepNet [S1]	Multiple	21.5	15.0	8.3	9.6
PRVG [S6]	Multiple	37.9	15.0	5.7	12.3
LGMR (Ours)	Multiple	<b>51.7</b>	<b>31.4</b>	<b>14.6</b>	<b>20.9</b>

**Ablation Study on Loss Weights.** As shown in Table S2, we actually determined the two loss weights by a grid search. It can be observed that setting  $\mathcal{L}_1$  to be relatively larger than  $\mathcal{L}_2$  gives a decently good model performance, and the best choice is  $\mathcal{L}_1 = 1.0$  and  $\mathcal{L}_2 = 0.2$ .

**Table S2: Hyper-parameter search in terms of  $\mathcal{L}_1$  and  $\mathcal{L}_2$**

$\mathcal{L}_1$	$\mathcal{L}_2$	R@ 0.3	R@0.5	R@0.7	mIoU
1.0	1.0	64.4	42.4	17.0	41.0
1.0	0.5	64.4	44.4	19.0	41.9
1.0	0.2	<b>67.9</b>	<b>46.7</b>	<b>21.8</b>	<b>44.4</b>
0.5	0.2	67.5	46.7	20.5	43.7

**Comparison with Single-Sentence Methods.** For a more comprehensive comparison, we show the comparative results of our LGMR with three single-query state-of-the-arts, i.e., CONE [S4], 2D-TAN [S10] and VSLNet [S9]. As shown in Table S3, our LGMR surpasses all single-query methods by a large margin. Note that CONE performs the worse since it is a method directly built on top of pre-trained vision-text models while the vision and text features in our dataset are not pre-aligned. VSLNet is the best-behaved single-query method since it generally models long-term video inputs by a well-designed split-and-concat mechanism.

**Table S3: Comparison with extra single-query baselines.**

Method	Query Input	R@ 0.3	R@0.5	R@0.7	mIoU
CONE [S4]	Single	4.7	1.8	0.5	-
2D-TAN [S10]	Single	-	8.8	3.2	11.5
VSLNet [S9]	Single	45.0	30.8	18.6	32.8
DepNet [S1]	Multiple	47.2	28.7	12.8	30.7
PRVG [S6]	Multiple	52.7	29.3	10.5	34.7
LGMR (Ours)	Multiple	<b>67.9</b>	<b>46.7</b>	<b>21.8</b>	<b>44.4</b>

### 1.2 More Implementation Details

We provide further implementation details for our proposed baseline model, including feature dimensions, positional encodings, subparagraph extractor, and loss calculation.

**Feature Dimensions.** We use pre-extracted SlowFast [S3], CLIP [S5], and OCR features with dimensions of 2304, 768, and 768, respectively. The CLIP and OCR features are first adaptively pooled at the sequence dimension to have the same length as SlowFast features. Then the three types of features are concatenated together at the hidden dimension as the input video features with a hidden dimension of 3840. A fully-connected layer is used to project the input video features to a 512-dimensional video representation for the temporal encoding and query decoding. Likewise, the pre-extracted text features have a hidden dimension of 768, and they are projected by a fully-connected layer to have the same feature dimension of 512. For all transformer layers in the encoders and decoders, the feature dimension is 512. All feed-forward layers have a hidden dimension of 2048 and the number of attention heads is set to 8.

**Positional Encodings.** As proposed in the vanilla transformer architecture [S8], we adopt a fixed set of high-dimensional sinusoidal embeddings to indicate positional information. The positional embeddings are employed on all transformer layers, including transformer layers used in the local-global temporal encoder and the local-global iterative decoder. Following the designs in DETR [S2], we only add positional embeddings with the feature inputs of query projection layers and key projection layers in all attention blocks.

**Subparagraph Extractor.** To construct the local subparagraph features from token-level text features for local-global iterative reasoning in our decoder, we adopt a set of learnable vectors  $O^S \in \mathbb{R}^{E \times D}$  to represent potential meaningful local semantics in a paragraph and then use one transformer decoder layer to extract useful subparagraph features in an end-to-end manner. Here we empirically set the number of learnable vectors  $E$  to be 10 in all our experiments and the positional embeddings are also only added to the inputs of query projection layers and key projection layers.

**Loss Calculation.** We calculate the localization loss  $\mathcal{L}_{loc}$  and attention loss  $\mathcal{L}_{att}$  for all transformer decoder layers in our local-global iterative decoder, following the common practice in DETR [S2] series works. Specifically, we feed the output global paragraph features of each decoder layer to an MLP predictor to predict the starting and ending timestamps. Note that we use a shared layer normalization module to normalize the output features of all layers before using them to predict the temporal interval corresponding to each paragraph query. During training,  $\mathcal{L}_{loc}$  is calculated on predictions produced by each decoder layer. Similarly, our attention loss is calculated on the temporal attention weights produced by each decoder layer during training. For testing, we only take the timestamp predictions from the last decoder layer of the model.

### 1.3 Qualitative Analysis

In this section, we aim to conduct qualitative analysis based on the visualization results, which can give a more intuitive understanding of our multi-paragraph video grounding dataset and model.

**1.3.1 Visualization Results.** First of all, we visualize and present a complete video-synopsis pair from the test set, as shown in Figure S1. This synopsis is composed of seven paragraph queries and most of them are very lengthy and complex. In addition, it can be seen that these paragraph queries typically contain multiple sentences that describe a variety of concepts at different levels of abstraction. For example, there are some abstract and concise expressions like “confided her troubles to” that summarize a long character conversation and convey the abstract concept “troubles” that may need a certain level of contextual reasoning capability to acquire an accurate understanding of it. Also, there are some concrete and detailed descriptions like “Jeremy flipped through his diary and saw a description of Vicky turning into a monster”, which requires to comprehend rich visual details presented in the video content for multimodal understanding. As a result of the above characteristics, jointly conducting contextual understanding of the video storylines and comprehensive perception of the visual details in each paragraph poses a crucial challenge for the video-language grounding models to overcome. Note that the target moments are also lengthy with a duration of several minutes, which requires models to effectively capture the more complex temporal structures of the video moments while retaining the ability of memorizing long-term visual contexts for better reasoning across multiple moments.

In Figure S1, we also compare our model’s predicted temporal intervals with the ground-truth timestamps to intuitively demonstrate the abilities of our multi-paragraph video grounding model. Overall, our model can make predictions close to the ground truth and correctly determine most of the temporal boundaries for the target video moments described by the given paragraph queries, although in some cases the boundary locations predicted by the model may not be very precise. To conduct a more detailed analysis of the model predictions, we further present the text content of each paragraph query in the synopsis and visualize some frames from the video moments corresponding to these queries, as illustrated in Figure S1. On the one hand, we can see the model is able to successfully predict the temporal intervals that have a high degree of overlap with the ground truth for the first two and the fourth paragraph queries, i.e.,  $Q_1$ ,  $Q_2$  and  $Q_4$ . In these cases, the paragraph queries are complicated and lengthy while containing rich complex concepts such as “Jeremy keeps asking about Vicky’s death, and Elena refuses Damon to continue hypnotizing Jeremy” and “Elena and Jeremy’s uncle paid a surprise visit, but Jenna didn’t welcome him”. Understanding these complex concepts requires the model to have a strong ability to associate a broad range of textual semantics in the paragraphs with the dialogue information as well as the visual activities in the video for precise temporal grounding.

**1.3.2 Analysis on Attention Weights.** On the other hand, we also observe some cases where the predictions are not very accurate. For instance, one of the two predicted temporal boundaries is accurate

while the other one deviates from the ground truth by a considerable margin for the third, fifth and seventh paragraphs, i.e.,  $Q_3$ ,  $Q_5$  and  $Q_7$ . We analyze them to find potential reasons case by case. For the third paragraph query in the synopsis, the model predicts a very accurate starting timestamp but predicts a much later ending timestamp in the video. In this case, localizing the ending timestamp is closely relevant to finding out the dialogue information between the drama characters referred to by the description “Elena found that Stefan was in bad shape, which Damon thought was the reason why Stefan had been depressed for too long”. Furthermore, we find the model incorrectly predicts the ending timestamp to be around the 20-th minute in the video, where a salient visual activity concerning the physical conflicts between two characters is located. This might indicate our model’s deficiency in resisting the distraction from irrelevant salient visual activities. For the fifth paragraph query, its predicted starting timestamp is later than the ground-truth starting timestamp. In this example, we find that the incorrectly predicted starting time is actually corresponding to the third sentence in this paragraph, i.e., “Taylor and Kelly flirted and got into a fight after Matt found out”, which means the model has missed the information associated with the first two sentences in the paragraph during video-language grounding. This phenomenon highlights the importance of fully understanding all the necessary detailed information contained in the long-term textual content of the paragraph queries and our model still needs to be improved in this aspect.

For the seventh paragraph query, we notice that the model predicts the starting timestamp to be around the point where the two characters’ dialogue mentions “the need is too strong” which is directly related to the key word “thirst” in the given paragraph. The model can only make its prediction for this case by considering the above kind of simple correlations between the video content and query semantics, thus causing an inaccurate starting boundary. Actually, the described character shows a very struggling and painful expression and body movements at the ground-truth starting time, which implicitly indicates the start of the video content specified by the query. However, the model fails to perceive such subtle human facial expressions and body movements to associate them with the plot contexts for predicting the starting boundary, which suggests this kind of ability needs to be further developed in future research. Last but not least, we also find the overall position of the predicted temporal interval of the sixth paragraph query  $Q_6$  is shifted a bit to the right at the time axis. In this case, we find that the starting time given by the model is very close to the moment corresponding to the third sentence in the paragraph, i.e., “Jeremy flipped through his diary and saw a description of Vicky turning into a monster”, which implies the model may miss the query information of the first two sentences in the paragraph. For the ending time, we find that localizing the ground-truth boundary requires to capture the short-term dialogue information implicitly corresponding to the query description “John told Alaric that the ring was inherited by the Gilbert family and that he had given it to Isobel”, while localizing a piece of short-term information from a long-term input is challenging and the model can be struggling to handle such cases.

In conclusion, the temporal intervals predicted by our baseline model can decently overlap with the ground truth in various cases, demonstrating the model’s ability to associate most concepts across

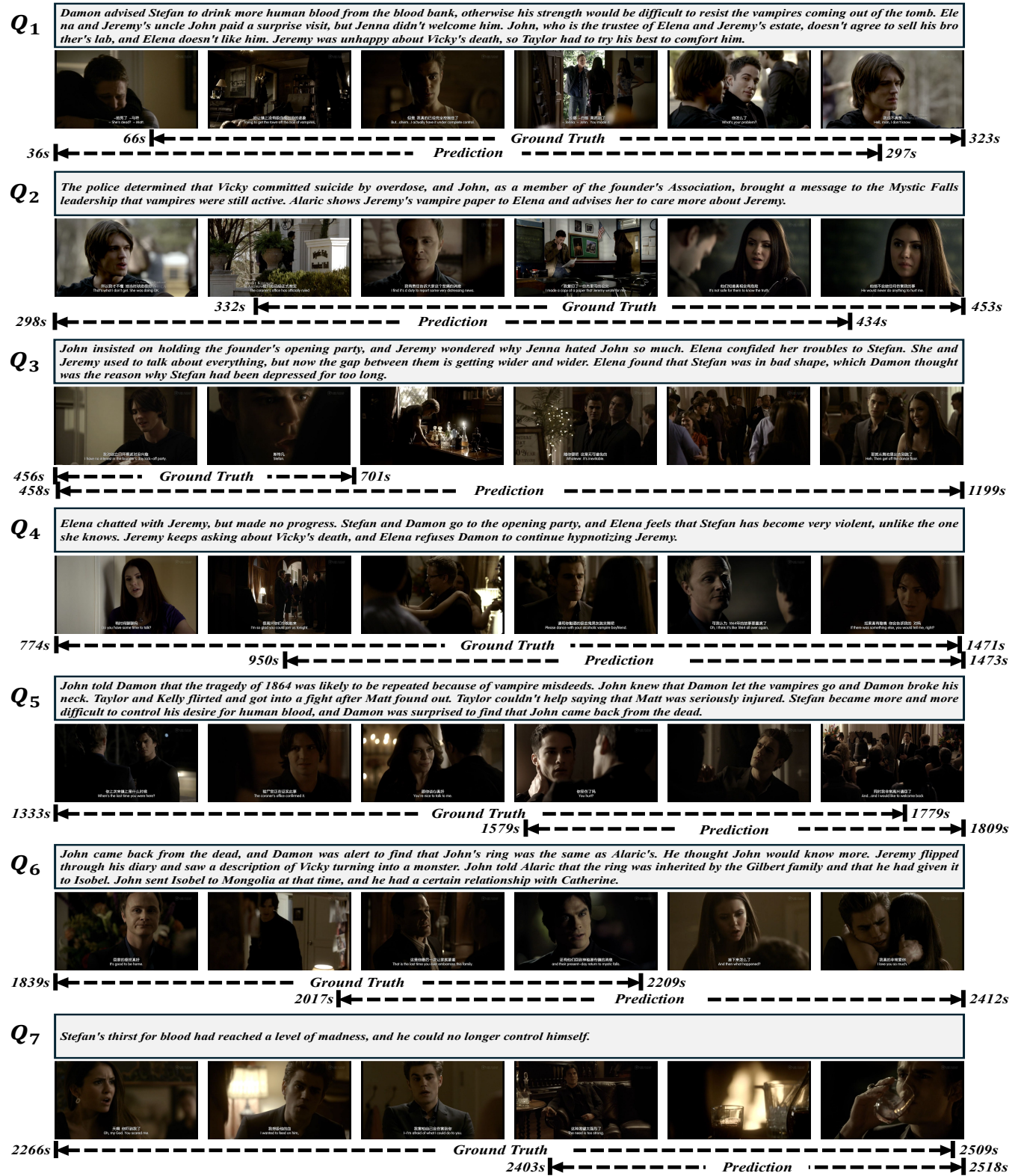


Figure S1: Visualization results of the model predictions and ground truth for multi-paragraph video grounding in SynopGround dataset. This example is selected from the test set and the video is the 18-th episode of the TV Drama *Vampire Diaries Season 1*. Due to space limitation, we uniformly sample frames from the temporal interval that encloses both the model predictions and ground truth for better visual presentation. (Best viewed on screen when zoomed in)

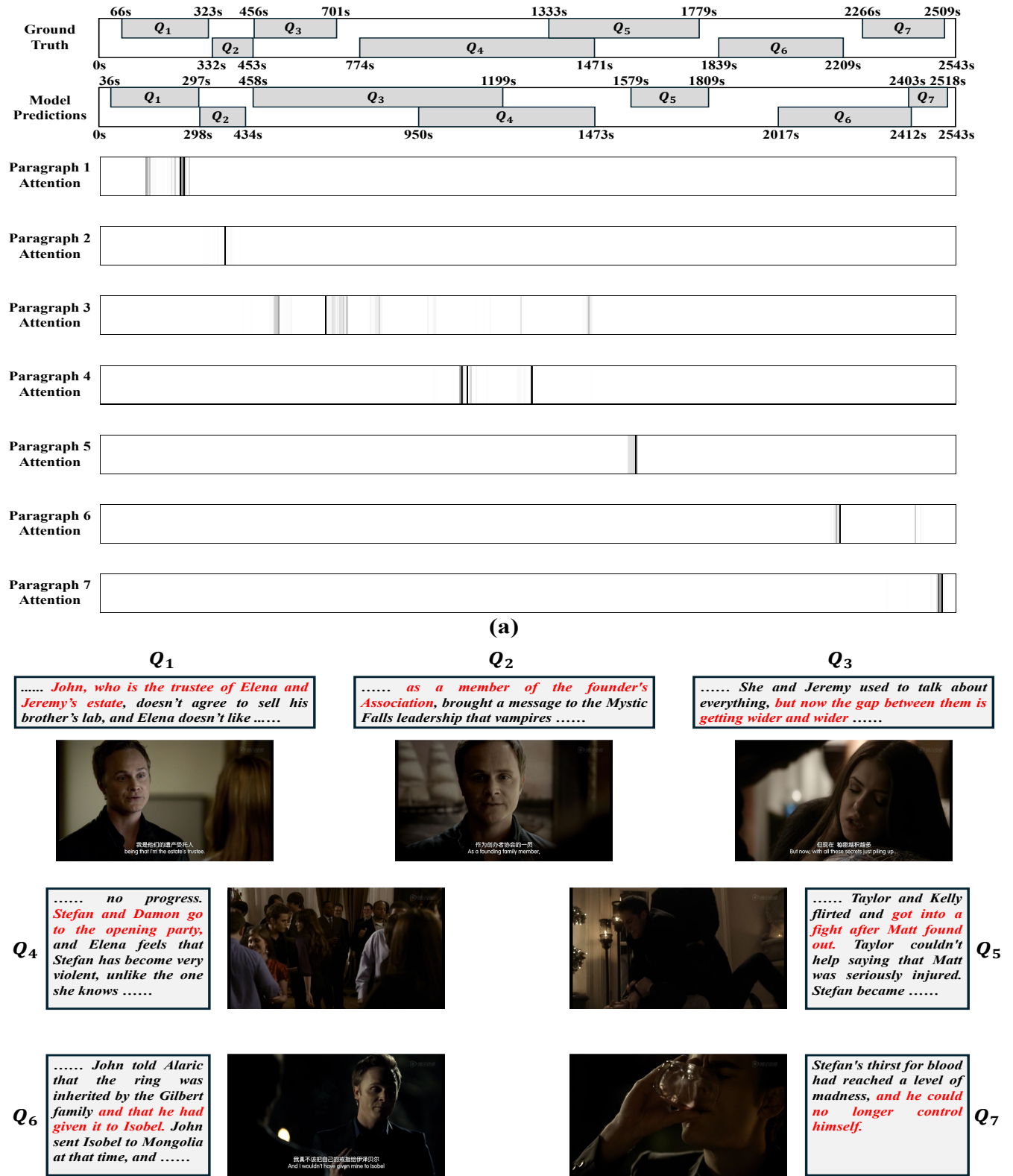
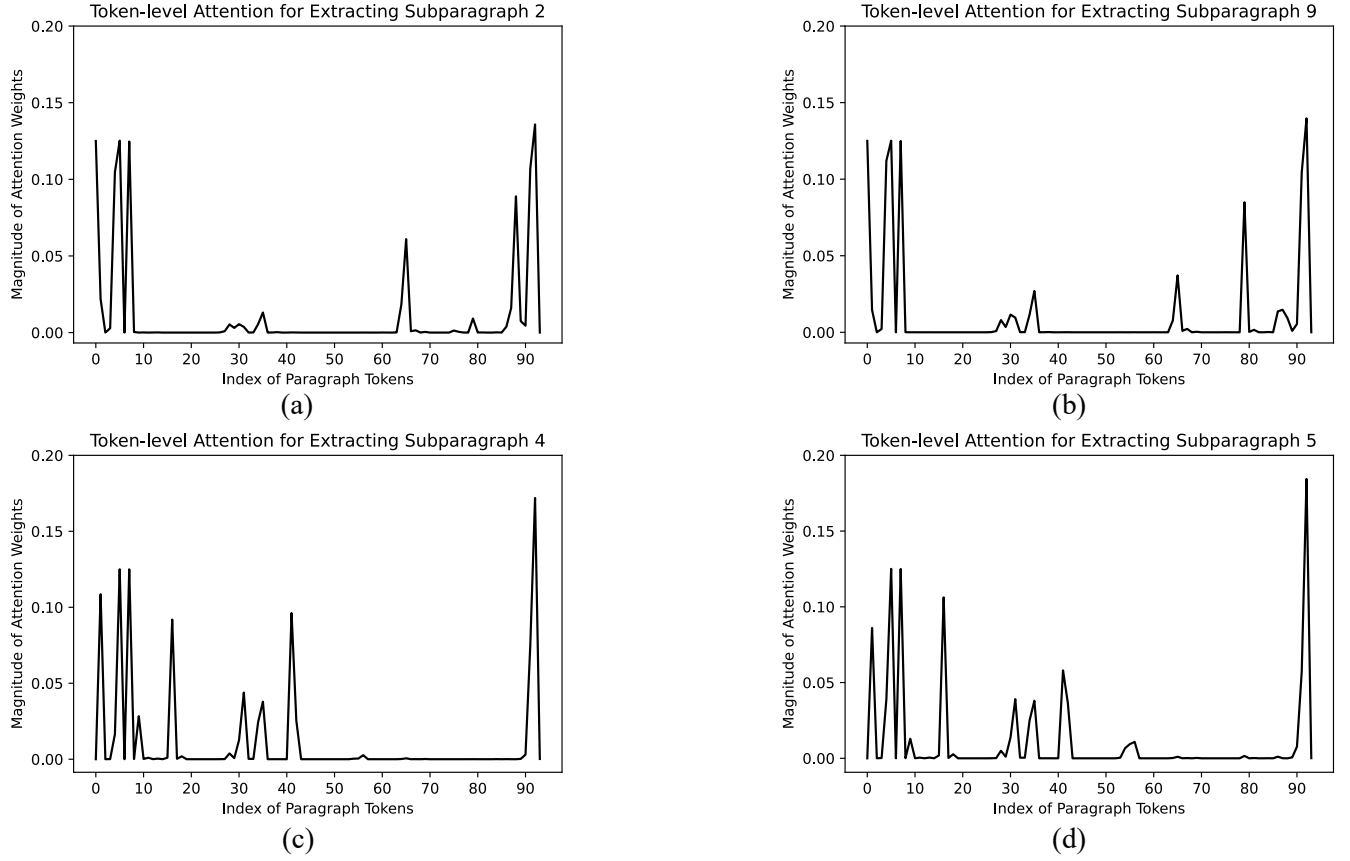


Figure S2: (a) Visualization on the paragraph-to-video attention weights from the last decoder layer. (b) Visualization on frames around the attention peak of the paragraph queries. Red text denotes the part of content within each paragraph query in the synopsis that is directly related to the visual frame located around an attention peak. (Best viewed on screen when zoomed in)



**Figure S3: Visualization results of the token-level attention weights for different local subparagraph representations. (a), (b), (c) and (d) respectively illustrate the attention weights from subparagraph 2, 9, 4 and 5 in the subparagraph extractor.**

visual and linguistic modalities. However, our model still struggles to predict very precise temporal boundaries in some challenging cases that demand deep understanding and complex reasoning of the story’s global context and crucial nuances. This points to an important direction for future work to develop stronger models that can better integrate global context and local details across modalities and conduct complex reasoning in a long contextual scope for better multi-paragraph video grounding.

**Paragraph-to-Video Attention Weights.** In Figure S2, we visualize the learned temporal attention weights from the last layer of the query decoder for the sample discussed in Section 1.3.1. To make a clearer visualization presentation, we additionally show the predicted timestamps of all paragraph queries in this sample along with the corresponding ground-truth labels in the upper part of Figure S2 (a). As we can see, both of the model’s final predictions and attention weights obviously follow a consistent temporal order with the ground-truth temporal intervals and the model’s predictions are highly correlated with the temporal positions where higher attention weights occur. This phenomenon intuitively suggests that the learned correlation between language queries and video content is crucial for achieving accurate temporal event localization. Encouraging high attention weights for relevant query and video elements is therefore beneficial for video-language grounding, which has also been quantitatively verified by the remarkable

effect of the cross-modal attention loss according to our manuscript. In particular, we also observe that for the third paragraph query  $Q_3$ , there are some temporal positions far away from the target moment that are spuriously attended by the model, which directly leads to a considerably delayed ending timestamp predicted by the model. Upon manually reviewing the corresponding video content that is spuriously attended by the model, we find that the main reason for the inaccurate prediction in this case lies in the model’s inability to correctly understand “She and Jeremy used to talk about everything”. In fact, the mistakenly attended frames are about the two characters talking with each other along the riverside, while the model incorrectly associates such content with “used to talk”, leading to inaccurate boundaries.

To more comprehensively understand the attention patterns of the decoder, we select video frames that are located around the temporal attention peaks of different paragraph queries and visualize them in Figure S2 (b). Overall, we observe that these frames with high attention weights from the paragraph queries are consistently correlated with certain descriptions in the corresponding paragraph, as shown by the red text in Figure S2 (b). Specifically, the dialogue information of the video content can be viewed as directly correlated with some part of the query content for  $Q_1$ ,  $Q_2$ ,  $Q_3$  and  $Q_6$ . In these cases, characters in the frames are talking about crucial information mentioned by the query. For example, the character

is introducing his identity as a “founding family member” in the visualized frame of  $Q_2$ , while this information is exactly mentioned in the second query by “as a member of the founder’s Association”. In addition to that, there are also cases where the dialogue information in the visualized frame is implicitly related to the query text. For instance, for the third query  $Q_3$ , the character is saying “with all these secrets just piling up”. This dialogue does not explicitly mention information about the “gap” but actually implies the gap between the two characters is becoming wider, which is described in the query as “but now the gap between them is getting wider and wider”. Furthermore, there are also cases where the visualized frames present the visual activities referred to by the corresponding query content, such as the situations in  $Q_4$ ,  $Q_5$  and  $Q_7$ . Concretely, characters are seen fighting in the frame relevant to  $Q_5$ , which is exactly described by the query as “got into a fight after Matt found out”. Particularly, the character is shown to be struggling inside and finally ends up drinking a cup of blood on the table, and this visual activity actually corresponds to the description “and he could no longer control himself” in the seventh query. In summary, we find that our model has the ability to find cross-modal correlations between query descriptions and the video content, regardless of whether information from different modalities is correlated explicitly or implicitly through character dialogues or visual activities.

**Subparagraph-to-Token Attention Weights.** In Figure S3, we further visualize the subparagraph-to-token attention weights in the query decoder to better analyze the local-level structure modeling in our local-global reasoning process. As presented, the different subparagraph features successfully learn to attend over different parts of the language tokens in the paragraph query. Intuitively, the attention weights from different subparagraphs can be roughly grouped into two patterns, i.e., the pattern shared by (a) and (b) and the pattern shared by (c) and (d), while different subparagraphs belonging to the same pattern still show a certain level of diversity. Moreover, the local semantics captured by different patterns of subparagraph representations are highly complementary to each other. For example, the attention weights in (c) mainly focus on language tokens at the front of the paragraph while the attention weights in (a) focus more on language tokens at the end of the paragraph. The complementary information contained by multiple subparagraphs helps the model to efficiently extract local semantic details. Therefore, adaptively extracting subparagraph features is an effective way to construct the local-global cross-modal reasoning process regarding the long-term multimodal inputs.

## References

- [S1] Peijun Bao, Qian Zheng, and Yadong Mu. 2021. Dense events grounding in video. In *AAAI*.
- [S2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *ECCV*.
- [S3] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. 2019. Slowfast networks for video recognition. In *ICCV*.
- [S4] Zhijian Hou, Wanjun Zhong, Lei Ji, Difei Gao, Kun Yan, W.k. Chan, Chong-Wah Ngo, Mike Zheng Shou, and Nan Duan. 2023. CONE: An Efficient COarse-to-fINE Alignment Framework for Long Video Temporal Grounding. In *ACL*.
- [S5] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*.
- [S6] Fengyuan Shi, Weilin Huang, and Limin Wang. 2024. End-to-end dense video grounding via parallel regression. *Computer Vision and Image Understanding* (2024).
- [S7] Mattia Soldan, Alejandro Pardo, Juan León Alcázar, Fabian Caba, Chen Zhao, Silvio Giancola, and Bernard Ghanem. 2022. Mad: A scalable dataset for language grounding in videos from movie audio descriptions. In *CVPR*.
- [S8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*.
- [S9] Hao Zhang, Aixin Sun, Wei Jing, Liangli Zhen, Joey Tianyi Zhou, and Rick Siow Mong Goh. 2021. Natural language video localization: A revisit in span-based question answering framework. *TPAMI* (2021).
- [S10] Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. 2020. Learning 2d temporal adjacent networks for moment localization with natural language. In *AAAI*.