Supplementary Material

Data annotation

We recruited annotators on Amazon Mechanical Turk. Figure A shows the annotation interface. Average time to complete one annotation is about 8 minutes. Some examples of annotated moments are in Fig B. The purple bars represent re-annotated moments and gray area represents a ground truth moment provided by the original dataset. We can see that the five annotators are likely to agree each other for some extent. On the other hand, the re-annotated moments sometimes do not overlap with ground truth. ActivityNet Captions dataset are created by writing a paragraph that describes the whole video and by annotating the temporal location in the video for each sentence. This annotation procedure is different from actual video moment retrieval task, because the original annotators are exposed to the surrounding sentences in the paragraph, which contextualizes the sentence, whereas on only a single sentence is available for the model for prediction as well as for our annotators.

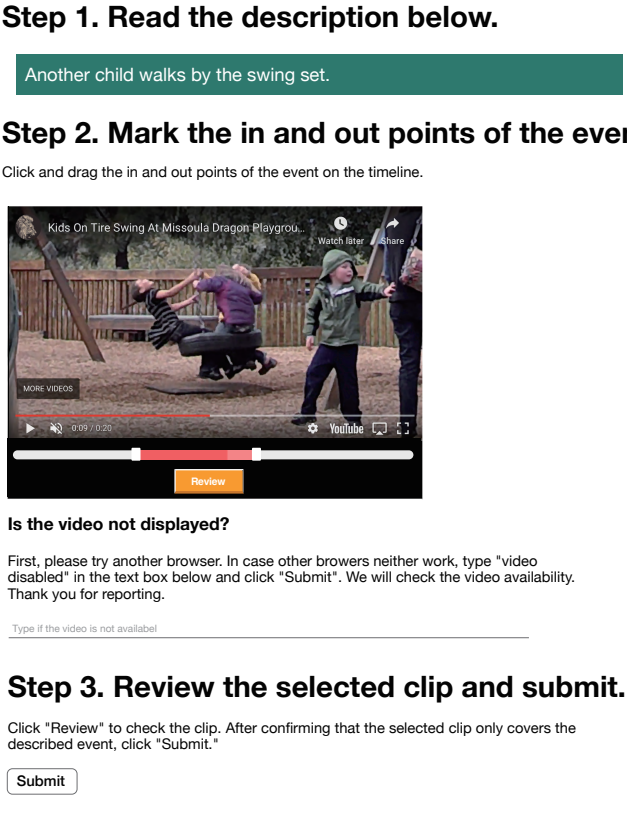
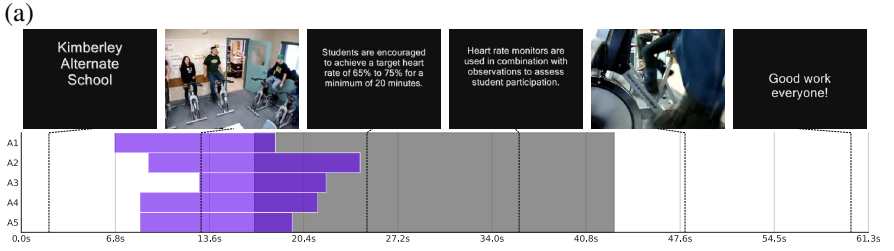
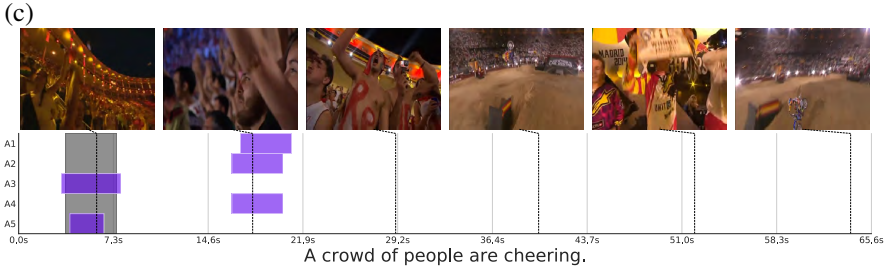


Figure A: Annotation interface. Annotators drag the sliders on the control panel and mark the start and end points of moment. Clicking “Review” button plays the selected moment.

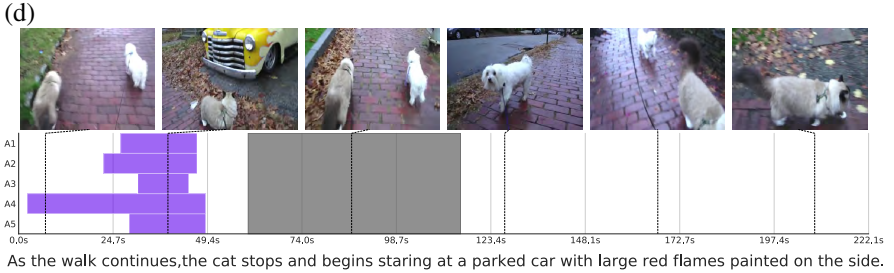


More people are seen riding on the exercise machine while the camera pans around their movements.

Then he sprays something on the car and sprays water on the car leaving it sparkly clean.



A crowd of people are cheering.



As the walk continues,the cat stops and begins staring at a parked car with large red flames painted on the side.

A male gymnast walks up to a beam.

Figure B: Examples of re-annotated moments on ActivityNet Captions.

Distributions of temporal locations

Figure C and D shows prior distributions of locations of moments. Each plot show a prior distribution of moments described by a verb on top right. We selected the top-30 frequent verbs of each dataset. Horizontal and vertical axes represent the start time and duration of a moment.

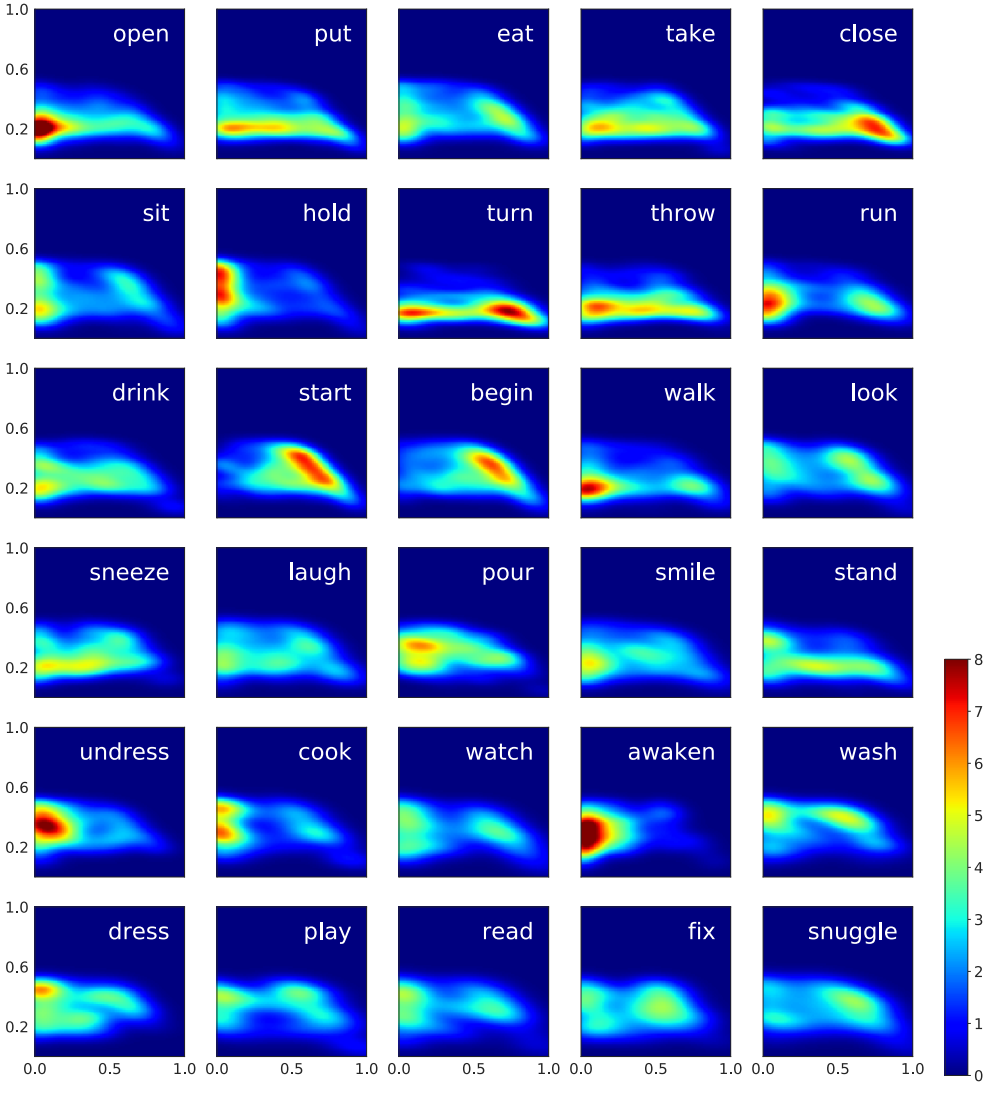


Figure C: Distributions of temporal locations of target moments on Charades-STA. The distributions are generated for the verb on top right. Color represents values of probability density functions.

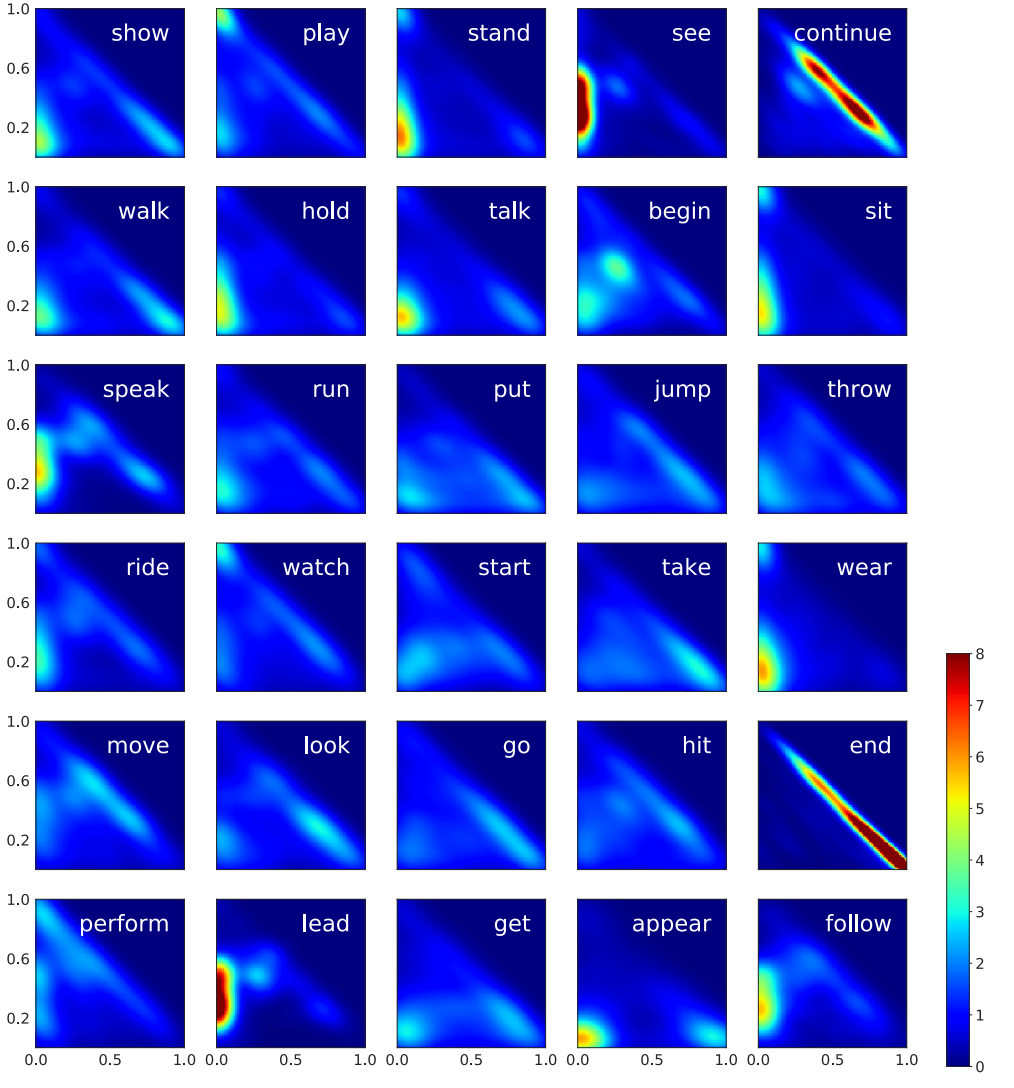


Figure D: Distributions of temporal locations target moments on ActivityNet Captions. The distributions are generated for the verb on top right. Color represents values of probability density functions.

Blind-TAN

To build Blind-TAN, we remove the component to handle input video in 2D-TAN, and instead, add a map $M_p \in \mathbb{R}^{N \times N \times d}$ where N is the number of sampling points of start and end times, and d is the channel size. We set N to 256 for both Charades-STA and ActivityNet Captions, and d to 512 for Charades-STA and 128 for ActivityNet Captions. Note that the prior map is learnable. Query sentence feature is compute by GloVe word embedding and LSTM layers. The query sentence feature and the prior map is fused by the Hadamard product. A multi-layer CNN transform the fused map into 2D score map. The score of a moment

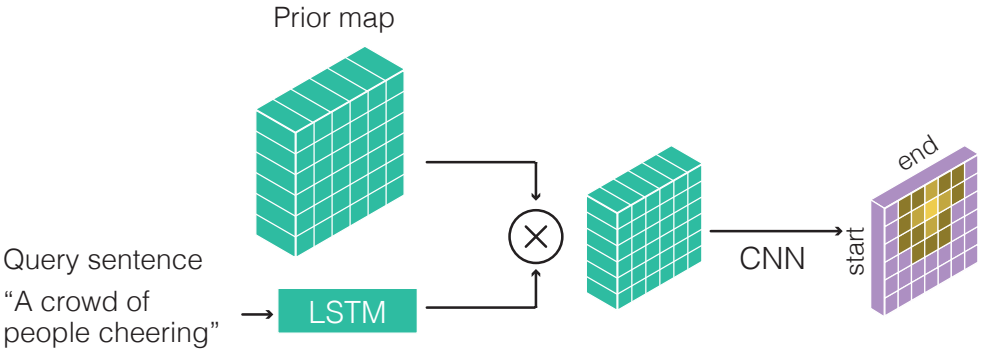


Figure E: Overview of Blind-TAN, which is a reduced version of 2D-TAN [19]. Blind-TAN only takes a query sentence as input and predicts where the target moment is likely to locate.

starting from point i and ent at point j is represented by the score map value at (i, j).

Blind-TAN is trained on pairs of a query sentence and a ground-truth moment location. Videos are not used for training nor testing. The prior map is initialized with random values and updated by backpropagating errors between the predicted score map and ground-truth. The loss function is a weighted binary cross-entropy as in [19].

The effects of randomized videos

We provide detailed results on how randomized videos change output moments. On the randomization test in Sec. 4, the deep models predict moment boundaries using original videos and randomized videos. Let (s_i, e_i) be start and end points of original prediction, and (s'_i, e'_i) be those on randomized video, where all values are normalized to the range $[0, 1]$ by dividing them by the length of the respective video. We compute how randomization affect the prediction by computing $d_i^s = |s_i - s'_i|$ and $d_i^e = |e_i - e'_i|$. Figure F shows the joint distributions of the differences d_i^s and d_i^e, where the Gaussian kernel density estimation is used to generate the distributions; the horizontal and vertical axes are d_i^s and d_i^e, respectively. Except for SCDM on Charades-STA, the models hardly change the output for randomized videos. This demonstrates that 2D-TAN and SCDM trained on ActivityNet Captions actually ignore input videos. Only SCDM on Charades-STA is affected by randomization. However, the start and end times change with similar amounts. That is, the predicted duration of moments are likely to be constant regardless of input videos. This tendency suggests that SCDM guesses the duration of target moments only using priors.

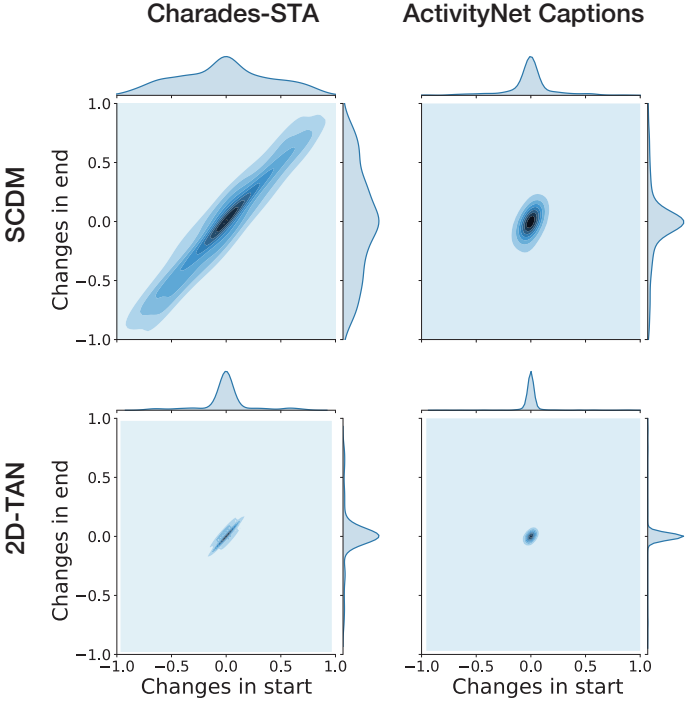


Figure F: Differences between temporal locations of moment predicted using original videos and randomized videos.