Video2Subtitle: Matching Weakly-Synchronized Sequences via Dynamic Temporal Alignment

Ben Xue
xueben@pku.edu.cn
Wangxuan Institute of Computer Technology, Peking University, Beijing, China

Chenchen Liu
liuchenchen@pku.edu.cn
Wangxuan Institute of Computer Technology, Peking University, Beijing, China

Yadong Mu*
myd@pku.edu.cn
Wangxuan Institute of Computer Technology, Peking University, Beijing, China

ABSTRACT

This paper investigates a new research task in multimedia analysis, dubbed as Video2Subtitle. The goal of this task is to finding the most plausible subtitle from a large pool for a querying video clip. We assume that the temporal duration of each sentence in a subtitle is unknown. Compared with existing cross-modal matching tasks, the proposed Video2Subtitle confronts several new challenges. In particular, video frames / subtitle sentences are temporally ordered, respectively, yet no precise synchronization is available. This casts Video2Subtitle into a problem of matching weakly-synchronized sequences. In this work, our technical contributions are two-fold. First, we construct a large-scale benchmark for the Video2Subtitle task. It consists of about 100K video clip / subtitle pairs with a full duration of 759 hours. All data are automatically trimmed from conversational sub-parts of movies and youtube videos. Secondly, an ideal algorithm for tackling Video2Subtitle requires both temporal synchronization of the visual / textual sequences, but also strong semantic consistency between two modalities. To this end, we propose a novel algorithm with the key traits of heterogeneous multi-cue fusion and dynamic temporal alignment. The proposed method demonstrates excellent performances in comparison with several state-of-the-art cross-modal matching methods. Additionally, we also depict a few interesting applications of Video2Subtitle, such as re-generating subtitle for given videos.

CCS CONCEPTS

• **Computing methodologies** → **Activity recognition and understanding**; **Visual content-based indexing and retrieval**.

KEYWORDS

Cross-modal matching, temporal alignment, deep neural networks

ACM Reference Format:
Ben Xue, Chenchen Liu, and Yadong Mu. 2022. Video2Subtitle: Matching Weakly-Synchronized Sequences via Dynamic Temporal Alignment. In *Proceedings of the 2022 International Conference on Multimedia Retrieval (ICMR '22), June 27–30, 2022, Newark, NJ, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3512527.3531371

*Corresponding author.

ICMR '22, June 27–30, 2022, Newark, NJ, USA

ACM ISBN 978-1-4503-9238-9/22/06…$15.00
https://doi.org/10.1145/3512527.3531371

1 INTRODUCTION

Recent decade has witnessed the emerging research on a variety of cross-modal matching or semantic analysis tasks between videos and natural language sentences. Examples include video captioning [26], video-question answering [23], video moment retrieval [24] etc. In particular, there are two lines of research that are mostly relevant to the interest task in this paper, including video captioning and lip reading[7, 8, 30]. The former aims to understanding the actions and events in the video through text descriptions (*e.g.*, tagging procedural videos such as learn-to-cook videos with textual event-level descriptions), and the latter attempts to translate lip motions into natural languages.

This paper investigates a new video-oriented research task for cross-modal matching, dubbed as Video2Subtitle. The primary goal of Video2Subtitle is to find a plausible subtitle to match up with a given video clip. Compared with other existing cross-modal matching tasks, there are several new technical challenges in this new Video2Subtitle task. One of the challenges lies in designing discriminative features for videos and subtitles respectively that can faithfully judge whether a video-subtitle pair is *harmoniously* matched or not. We regard two different kinds of information is crucial for this aim. First, given a subtitle (*i.e.*, a collection of temporally-arranged sentences) and a video clip, both subtitle (via text-to-speech) and video frames (via vision-based lip detection) can be mapped to a sequence of lip motion, respectively. For a true video-subtitle matching, these lip motions shall intuitively match well. However, due to homonyms, different pronunciation styles over different speakers or occlusions of lips in frames, it is non-trivial to find the best matched subtitle for a video clip given the partially-observed, often noisy lip-oriented features. In most cases, one may face several subtitles that are all plausible for the same video clip. This spurs us to also harness other types of features to re-rank the subtitles, such that false matching can be largely eliminated. Intuitively, at cognition level, visual object, scene and action information in the videos can be semantically associated with what the speakers are talking about. This frequently occurs in YouTube videos, where people talk about cooking-related objects in a scene of kitchen, or dressing styles during shopping. Figure 1 illustrates such an example. To find a good matching, it is important to ensure the holistic descriptors of video and subtitles are reasonably aligned, otherwise the audience will immediately notice cross-modal disaccord. For instance, it is weird for a video with in-class teaching activities being associated with a news-report-themed subtitle.

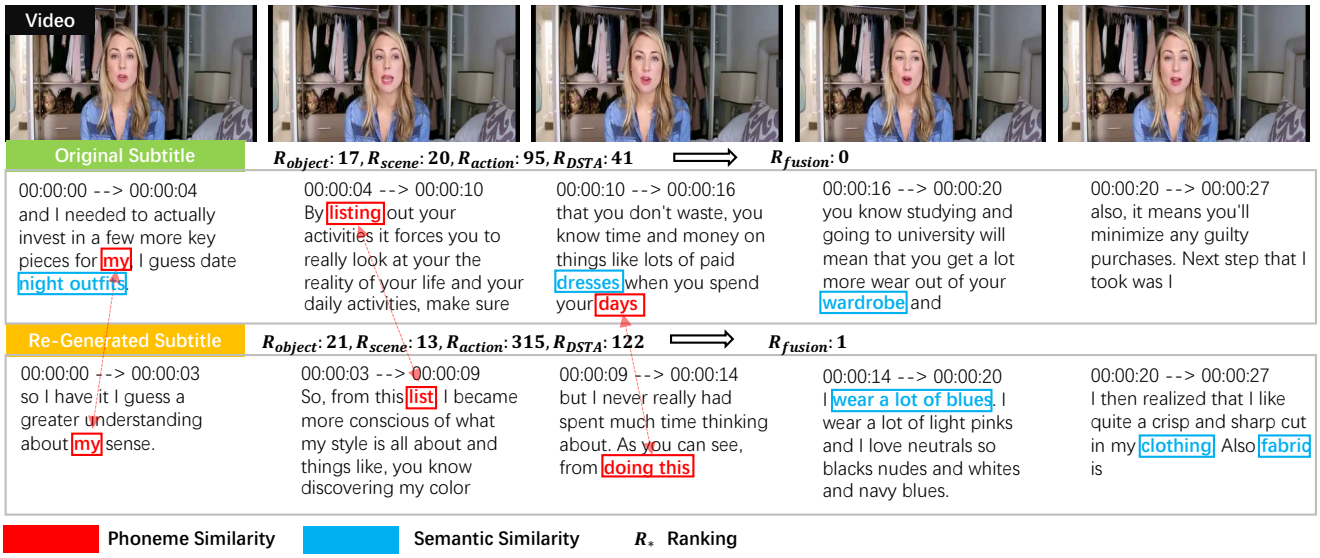


Figure 1: Illustration of the Video2Subtitle task. We propose a novel algorithm which jointly utilizes temporal alignment based on phoneme similarity and fuses different semantic cues (object, scene, action etc). Top / bottom subtitles are ground-truth and the one ranked first by our algorithm, respectively.

Besides the multi-cue fusion issue [29, 41, 43], an additional challenge rises from the fact that Video2Subtitle is fed with weakly-synchronized data. Video frames are sequentially arranged according to their timestamp. A subtitle is often comprised of a set of natural language sentences. The temporal relationship of these sentences can be directly inferred from the order in the subtitle. However, the key information of temporal duration for each sentence is generally unavailable. This collectively defines a sequence matching problem on weak-synchronized data.

Enforcing temporal alignment is thus crucial. To this end, we argue that there are two protocols that the video-subtitle matching should satisfy. (1) The temporal duration of subtitle / video segment with continuous speaking activity should match with each other. To the audience it will be weird for observing a short-duration speech associated with a long sentence in the subtitle, or vice verse. (2) The transform of subtitle to speech implicitly defines a sequence of lip motions, which should be well aligned the speaker's real lip motions visually detected in the videos. Conventionally, dynamic temporal warping (DTW) is a standard tool for conducting the piece-wise temporal alignment between two sequences that do not sync up perfectly. We argue that directly harnessing DTW for the Video2Subtitle task is not an optimal choice. The core dynamic-program update in DTW demands strict temporal order preserving, not allowing any temporal misalignment during the sequence matching. However, the issues of occluded lip and offscreen speech frequently happen in the interested videos. To mitigate it, we propose a new sequence matching algorithm called duration-shifts temporal alignment (DSTA). It consists of modules for detecting video segments with continuous speaking and estimating how long it takes for reading a subtitle sentence. It also adopts a dynamic window so that models can perform temporal forward or backward shifts when updating states to improve the robustness of the algorithm. We also implement DSTA as a differentiable module so that it can be plugged into arbitrary neural backbones and trained in an end-to-end manner.

As another contribution, we establish a large-scale video-subtitle dataset. The data has two main sources, either from commercial movies and YouTube videos, which focus on fictional and real-world conversations respectively. We collect 100,115 dialogue / monologue clips from 629 movies and 4,138 YouTube videos. The chosen videos cover a diverse set of daily-life themes. In contrast, a few previous works [7, 8, 23, 24] collect video-subtitle data from TV series or TV shows. Since episodes from the same series often share similar semantics, they are inadequate in terms of data diversity. For each video in our dataset, accurate tight timestamp for each subtitle sentence is provided for model training purpose.

This paper also depicts a few interesting applications of the proposed Video2Subtitle algorithm, including what we term as *subtitle re-generation*. It is related to DeepFake [19, 32, 34] which synthesizes unseen videos based on provided text, yet operates in an opposite direction (*i.e.*, searching text for given videos). Figure 1 shown such a re-generated subtitle which is top-ranked by our algorithm, based on joint phoneme and semantic similarities. This paves the way of replacing the original subtitle of a video with other ones (such as those from a movie database) for entertainment purpose.

2 RELATED WORK

Video-text matching has been extensively studied in recent years. There are several research thrusts that are particularly related to the main scope of this work.

Video-text datasets. Many video-text datasets flourished in recent years. Some datasets focus on action or event description.

Table 1: Comparison between existing video-subtitle datasets and our proposed new data. "–" implies that the existence of in-screen speaking faces is not explicitly ensured for each video clip.

	#Clips	Durations (in hours)	Subtitles (in lines)	Data sources	Inscreen speaking faces
MovieQA [37]	0.2K	7.7	0.62M (raw)	408 movies	-
TVR [24]	21.7K	461	0.04M (cleaned)	6 TV shows	-
MovieNet [18]	41.3K	214	1.0M (raw)	1,100 movies	-
Ours	100K	759	1.1M (cleaned)	629 movies / 4,138 YouTube videos	Yes

For example, [26, 28, 42] were collected from real-world videos, while [23, 24, 35] were collected from movies and TV series. [12] was collected from cooking videos on YouTube. These videos are all procedural ones, conveying rich scene or action information. Yet, the accompanying text is often post-generated by annotators with pre-defined structure. Some other datasets focus on human language modeling. [7, 8] were well-known datasets for lip reading with strict temporal alignment. However, the visual scenes and human actions in these videos often exhibit limited diversity. [37] mainly aims to evaluate automatic story comprehension thus it is essentially for a QA-like task. [18] is a comprehensive movie dataset with metadata labeling, subtitle is also provided but it is not cleaned to ensure the coherence of speaker and subtitle. A considerable body of movie shots are indeed silent (32% non-conversational subtitles according to our statistics via speaker detection algorithm). [23, 24] generate meaningful conversational clips by human-annotation, but the total duration and subtitle lines are limited. [28] collects huge amount of YouTube videos with subtitle, but they are mostly procedural and narrative videos, with object-centered contents rather than speaking human. A detailed comparison of related datasets can be found in Table 1.

Video-text matching. Extensive works have been devoted to video-text cross modal matching. Canonical correlation analysis (CCA) [38] is a linear method which computes projection matrices for two modals respectively and maximizes linear correlation between them on the projected subspace. [2] replaces the original linear projection with learnable deep layers to obtain non-linear transformations. [43] introduces a joint embedding model which mapped visual and text embedding into the same common space. [44] adopts LSTM [15] and attention [39] module to trace action events in the video. A word-level attention module is used to selectively focus on detected concept words. [13] uses multi-scale sentence embedding strategy. Their follow-up work [25] applies improved triplet ranking loss [14] and obtains better results. [36] merges visual-text features into sequence self-filling tasks similar to [12] and explores video-text relationship in a self supervision style. Aforementioned works perform global pooling temporally when encoding sequence representation, not amenable for temporal alignment. Instead, [5] encodes hierarchical graph representations for sequences and formulates a graph-matching problem. This inspires us to design a network with local alignments because video-subtitle has stronger temporal correlation than video-caption.

Dynamic time warping (DTW) is a popular method for matching temporal sequences. It measures flexible feature similarity under time distortions. However, DTW suffers from pathological alignment problem when matching long-duration sequences. Some methods apply window constraints or temporal clustering to deal with local structures [3, 16, 45, 46], but they are not end-to-end with learnable parameters. Recently a differentiable soft-DTW [10] is introduced to measure sequence similarity. It replaces the original argmax in DTW with a soft-max operator such that gradients can be back-propagated to update the parameters.

3 DATASET

Data collection. We collected video clips from two sources: (*i*) 629 movies released in the past decade, and (*ii*) 4,138 human-related YouTube videos, mainly themed *vlogs, academic talks and interviews*. In detail, the YouTube videos are searched using 321 keywords which have smallest distance to *human* on [1]'s visual knowledge graph. Each keyword contributes about 10 videos. TV series are not included in our data source, since we observe that different episodes from the same series often share similar scenarios, reducing the visual diversity. For the collected videos, each movie lasts about 2 hours on average and YouTube videos are typically shorter than 10 minutes. As stated later, we first adopt sophisticated scheme for identifying conversational clips from these videos. The selected clips are all trimmed within 15-30 seconds, totaling 58,306 clips from movies and 41,809 clips from the YouTube videos. Movie subtitles are gathered from online subtitle community *opensubtitle.org*, and the subtitles of YouTube videos are downloaded directly from the closed-caption (CC) of the original website. The performance evaluation is separately conducted on movie and YouTube videos. Among the video clips, 5,000 clips are kept confidentially as the testing set on both movie and YouTube sets.

Conversational video clip detection. Human-centric conversational clip is of primary attention in constructing the Video2Subtitle dataset. To distinguish interested clips from others, we first separate subtitles into local groups with a duration of 15-30 seconds. Importantly, if the silence time between two adjacent sentences is above specific threshold (set to 4 seconds in practice), these sentences will be treated to belong different local groups. This way aims to avoid conversational topic shifting within a same group.

Afterwards, a face detector is applied on each video frame in order to filter out narrative or non-human procedural video clips. In practice, we uniformly sample T frames from a video clip and count the number of conversational frames[1], denoted by T_{active}. Only clips with a ratio T_{active}/T above some threshold (*e.g.*, 0.65) will be kept for further use. In addition, if a significant portion of the

[1]A video frame is regarded to be *conversational* if at least one face can be visually detected.

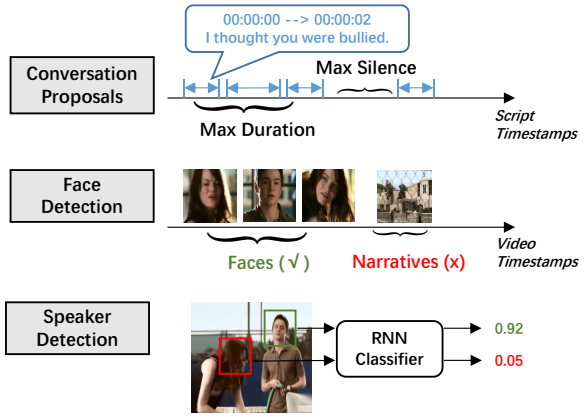


Figure 2: Computational pipeline for conversational video clip detection. See the main text for more details.

video frames contain too many co-occurring faces (*e.g.*, ≥ 3), the clip will be also abandoned in order to reduce the effort on speaker identification. The face detector returns face tubes of multi-frame segments. It is necessary to further inspect the face tubes, such that non-speaking faces will be removed. To this end, we feed each face tube into a RNN network pretrained on a densely labeled human action dataset [9, 21]. The output of last sigmoid layer is used for indicating the speaking activity. if there exist multiple faces in one single frame, we only keep the face with maximal speaking score. Figure 2 summarizes the procedure.

Our data sources provide relatively accurate temporal localization for each subtitle sentence. Thus, for each video clip, it is paired with the automatic annotation $S = \{(Text_i, t_i^{start}, t_i^{end})\}_{i=1}^{M}$, which are essentially M sentences of plain text and the corresponding start-end timestamps.

Global and temporally-stamped features. To obtain comprehensive visual representations, we extract features for both global semantic appearance and temporally-stamped facial motions. For the global appearance, we extract 2,208-D densenet-161 [17] feature at a sampling rate of 8 fps (frame per second) for object and scene information, respectively pretrained on Imagenet [11] and Places365 [47]. For encoding the global motion, we feed 1.5-second segment into I3D [4] network pretrained on Kinetics-400 [20] to obtain a 1,024-D feature vector as action information. For the facial information, we send face tubes into a pretrained lip-reading model SyncNet [30] with 3D-Convolution encoder and the 512-D outputs for each frame are used as temporally-stamped lip features.

For the textual subtitles, we concern both semantic word embedding and pronunciation-related similarity. Glove [33] is adopted to extract a 300-D embedding for each word in the subtitle. All Glove features are eventually temporally pooled (as later discussed in Section 4.2) into a global vector. This allows us to match semantic information between visual and textual modalities. Furthermore, in order to match lip motions and words, a different feature space is required to convey the pronunciation-based difference among words. The scope of conventional word embedding is primarily semantic affinity rather than pronunciation. For example, *cat* and *dog* are semantic neighbors yet clearly have different pronunciations. We propose a two-step process for converting text into pronunciations-based feature. The proposed module first translates text into speech via *text-to-speech* (TTS) toolkit Tacotron [40], and SyncNet [30] is then used to extract temporally-stamped acoustic features.

4 METHOD

This section elaborates on our proposed model for the Video2Subtitle task. It consists of two modules (namely, temporal alignment and global semantic matching), which are detailed in Sections 4.1 and 4.2 respectively. The entire algorithm framework is shown in Figure 3.

4.1 DSTA: Duration-Shifts Temporal Alignment

4.1.1 Forward-backward shifts. The core framework is based on dynamic program (DP). Assume that we need to match a video sequence $V = [v_i; i = 1, ..., l_v]$ and a subtitle sequence $S = [s_i; i = 1, ..., l_s]$. Given the cost matrix $\Delta(V, S) = \delta(v_i, s_j) \in \mathbb{R}^{l_v \times l_s}$ that gauges all pairwise affinities, a standard DTW routine will update a state matrix $R \in \mathbb{R}^{l_v \times l_s}$ according to the following formula:

$$r_{i,j} = \delta(v_i, s_j) + \min\{r_{i-1,j-1}, r_{i-1,j}, r_{i,j-1}\}, \tag{1}$$

where $r_{i,j}$ is the entry in R indexed by (i, j).

Differently, our duration-shifts temporal alignment (DSTA) allows temporal shifts in a time window beyond adjacent frames:

$$r_{i,j} = \delta(v_i, s_j) + \min_{j-k<p<j+k} \{r_{i-1,p}\}, \tag{2}$$

where the parameter k controls the window size of forward-or-backward inspection.

As for the DP procedure, forward shifts ($j - k < p < j$) imply skipping over a non-matched subtitle segment, which is possibly caused by face occlusion. Similarly, backward shifts ($j < p < j + k$) define a regret mechanism which can eliminate misalignment in previous step. The major advantage of such temporal shifts is that it increases the robustness in real world scenarios.

Critically, as the data is weakly-synchronized, the final alignment of sequences is still encouraged to obey the temporal-order constraint (*i.e.*, $p < j$) for most cases. Therefore we introduce an order-sensitive penalty term for backward shifts:

$$\lambda_{p,j} = \max(p - j - m, 0), \tag{3}$$

$$r_{i,j} = \delta(v_i, s_j) + \min_{j-k<p<j+k} \{r_{i-1,p} + \lambda_{ord}\lambda_{p,j}\} \tag{4}$$

where m is a margin parameter that defines the maximum backward steps without any penalty. λ_{ord} is a weighting hyper-parameter. The detailed forwarding procedure is illustrated in Algorithm 1.

4.1.2 Duration-based constraint. When the subtitle sentences are converted into audio signals via *text-to-speech* (TTS), their duration can be roughly estimated since audio data is also time-stamped as videos. The number of syllables one can speak within a time unit can be assumed to approximately obey a Gaussian distribution $\mathcal{N}(\mu, \sigma)$. We argue that when calculating $\delta(v_i, s_j)$, it is important to consider time-related constraint. Given video timestamp i and subtitle timestamp j, we propose a duration-based constraint $\delta_{dur}(i, j)$ as:

$$\delta_{dur}(i, j) = 1 - \exp\left\{-\frac{|j - \omega i|}{2(\eta\sqrt{i})^2}\right\}, \tag{5}$$

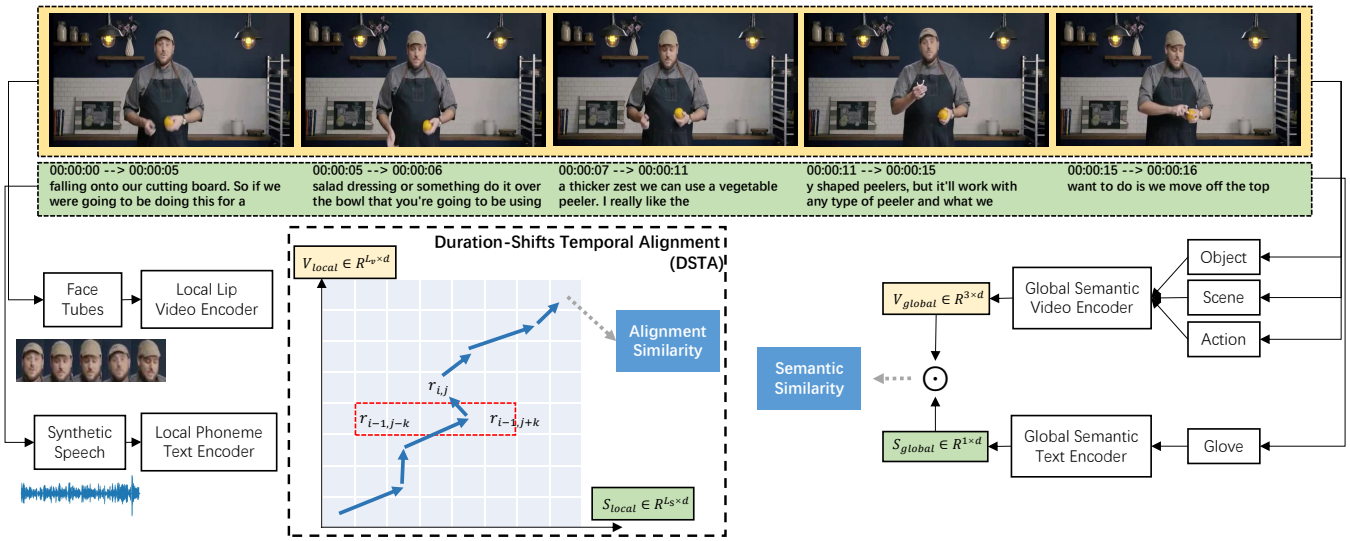


Figure 3: The proposed framework. We design two branches to harness multi-cue fusion. Duration-Shifts Temporal Alignment module (DSTA) in Section 4.1 is used for phonome similarity and duration constraints. Global Semantic Matching module (GSM) in Section 4.2 is used for video theme similarity. The whole pipeline is trained in an end-to-end manner.

where we empirically set mean value $\mu = \omega i$ according to some prior knowledge. ω stands for the average speech rate for an ordinary person. The standard deviation $\sigma = \eta\sqrt{i}$ compensates variable speech rates among different speakers. Generally, the error σ^2 is also proportional to time since the residual accumulates along time.

Note that duration-based constraint leads to a normalized penalty over [0, 1), Putting all together, the distance function $\delta(v_i, s_j)$ is formulated as:

$$\delta(v_i, s_j) = 1 - \left\langle \frac{v_i}{||v_i||}, \frac{s_j}{||s_j||} \right\rangle + \lambda_{dur}\delta_{dur}(i, j), \tag{6}$$

where we adopt cosine similarity in feature space of (v_i, s_j).

4.1.3 Differentiable module design. To compute the distance between v_i and s_j as in Eq. (6), we need to project video and subtitle modalities into a common feature space. With the rapid development of cross-modal deep learning, there are many backbone networks committed to this task. It is highly favored to ensure the sequence matching module in the front-end differentiable, such that the whole pipeline can be trained in an end-to-end fashion.

The main obstacle is the *min* operation in DP. Inspired by soft-DTW [10], we adopt soft minimum $\min^{\gamma}$ to replace hard minimum:

$$\min(a_1, ..., a_n)_{\gamma} = \begin{cases} \min_{i \le n} a_i, & \gamma = 0 \\ -\gamma log \sum_{i=1}^{n} e^{-a_i/\gamma}, & \gamma > 0 \end{cases} \tag{7}$$

Therefore the computation of Eq. (4) now becomes:

$$r_{i,j} = \delta(v_i, s_j) + \min_{j-k<p<j+k}{}^{\gamma}\{r_{i-1,p} + \lambda_{ord}\max(p - j - m, 0)\}. \tag{8}$$

To derive the gradient propagation, we need to apply chain rule:

$$\frac{\partial r_{n,m}}{\partial r_{i,j}} = \sum_{p=j-k}^{j+k} \frac{\partial r_{n,m}}{\partial r_{i+1,p}} \frac{\partial r_{i+1,p}}{\partial r_{i,j}}, \tag{9}$$

where we define the main notation of the backward recursion object $e_{i,j} := \partial r_{n,m}/\partial r_{i,j}$. This object can be computed recursively once we have explict formulation of $\partial r_{i+1,p}/\partial r_{i,j}$. Take the log-sum-exp operation of soft minimum in Eq. (7), let $\lambda_{ord} = 1$, we can get:

$$\frac{\partial r_{i+1,p}}{\partial r_{i,j}} = e^{-r_{i,j}/\gamma} / \sum_{p=j-k}^{j+k} e^{(-r_{i,p}-\lambda_{p,j})/\gamma}. \tag{10}$$

A simplified version is casting lograthm of the derivative:

$$\gamma log \frac{\partial r_{i+1,p}}{\partial r_{i,j}} = \min{}^{\gamma}\{r_{i,p} + \lambda_{p,j}\} - r_{i,j} \tag{11}$$

$$= r_{i+1,j} - \delta(v_{i+1}, s_j) - r_{i,j} - \lambda_{p,j}. \tag{12}$$

Thereby we obtain a recursive backward propagation to compute the entire matrix $E = [e_{i,j}]$. Note that the derivative of cost matrix $\Delta(V, S)$ to video V or subtitle S can be conveniently computed from the definition of distance functions, namely:

$$\nabla_V DSTA_{\gamma}(V, S) = (\frac{\partial \Delta(V, S)}{\partial V})^T E. \tag{13}$$

which is the return value of the entire backward procedure in Algorithm 2.

4.1.4 Implementation of DSTA. To facilitate cross-modal training, we adopt SyncNet [30] as the backbone. As for video modal, it samples RGB frames of face tubes at 24 fps (frame-per-rate) and uses 6 cascaded groups of (Conv3D, ReLU, BatchNorm, MaxPool3D) to extract features. As for the subtitle modal, we first use text-to-speech technology (TTS) to convert natural language sequences into a stream of continuous audio, and then convert wave format into spectrum graph and apply 6 cascaded groups of (Conv2D, ReLU, BatchNorm,MaxPool2D). The model mentioned so far is pretrained on VoxCeleb dataset [31]. Video and subtitle sequences are both projected into a 512-D space $\mathbb{R}^{l \times 512}$.

Algorithm 1 Forward recursion to compute $DSTA_\gamma(V,S)$ and intermediate alignment costs

Inputs: V, S, smoothing $\gamma \geq 0$, distance function δ, order punishment λ, window size $2k+1$
$r_{0,0} = 0; r_{i,0} = r_{0,j} = \infty; i \in [\![n]\!], j \in [\![m]\!]$
for $i = 1, \ldots, n$ **do**
  for $j = i-k, \ldots, i+k$ **do**
    $r_{i,j} = \delta(v_i, s_j) + \min^{\gamma}_{j-k<p<j+k}\{r_{i-1,p} + \lambda_{p,j}\}$
  end for
end for
Output: $(r_{n,m}, R)$

Algorithm 2 Backward recursion to compute $\nabla_V DSTA_\gamma(V,S)$

Inputs: V, S, smoothing $\gamma \geq 0$, distance function δ, order punishment λ, window size $2k+1$
$(\cdot, R) = DSTA_\gamma(V,S), \Delta = [\delta(x_i, y_j)]_{i,j}$
$\delta_{i,m+1} = \delta_{n+1,j} = 0, i \in [\![n]\!], j \in [\![m]\!]$
$e_{i,m+1} = e_{n+1,j} = 0, i \in [\![n]\!], j \in [\![m]\!]$
$r_{i,m+1} = r_{n+1,j} = -\infty, i \in [\![n]\!], j \in [\![m]\!]$
$\delta_{n+1,m+1} = 0, e_{n+1,m+1} = 1, r_{n+1,m+1} = r_{n,m}$
for $i = n, \ldots, 1$ **do**
  for $j = m, \ldots, 1$ **do**
    $c_{i+1,p} = \exp\frac{1}{\gamma}(r_{i+1,j} - \delta(v_{i+1}, s_j) - r_{i,j} - \lambda_{p,j})$
    $e_{i,j} = \sum_{p=j-k}^{j+k} e_{i+1,p} \cdot c_{i+1,p}$
  end for
end for
Output: $\nabla_V DSTA_\gamma(V,S) = (\frac{\partial \Delta(V,S)}{\partial V})^T E$

We further append a simple front-end of 2 groups of (Conv1D, ReLU, BatchNorm, AvgPool1D) to get video and subtitle representations $V \in \mathbb{R}^{l_v \times 512}$ and $S \in \mathbb{R}^{l_s \times 512}$. Then we compute the best matching score $c = DSTA_\gamma(V,S)$ for two sequences. Afterwards, we use ranking loss to maximize the margin between negative pairs and positive pairs:

$$L^{dsta}_{rank} = max(c^{dsta}_{+} - c^{dsta}_{-} - margin, 0). \tag{14}$$

During training, the ground truth alignment for the (V,S) pairs is available. We thus can include extra supervision on the cost matrix $\Delta(V,S)$, setting the entries on alignment path to 1 otherwise 0. Standard cross-entropy loss is adopted to encourage closer distance between aligned frames, leading to a second loss for DSTA:

$$L_{align} = CE\left(\left(\left\langle \frac{v_i}{||v_i||}, \frac{s_j}{||s_j||} \right\rangle + 1\right)/2,\ \mathbb{I}(i=j)\right). \tag{15}$$

4.2 Global Semantic Matching

4.2.1 Temporal aggregation network. A plausible subtitle should also has reasonable semantics with respect to video appearance. Such kind of relationship does not require accurate temporal alignment. A global summary of sequence information is sufficient. We adopt a typical temporal network with GRU [6] to extract temporal information, the output hidden states h_t is then sent to different Conv1D kernels (k=2,3,5) and a global average pooling layer to aggregate information in a multi-scale style. Finally we concatenate these summarized features and use a linear layer to project video and subtitle modalities into a common space, attaining V_{global} and $S_{global} \in \mathbb{R}^{1\times 2048}$. Same as the training of DSTA, we also use a ranking loss similar to the one in Eq. (14):

$$L^{gsm}_{rank} = max(c^{gsm}_{+} - c^{gsm}_{-} - margin, 0). \tag{16}$$

4.2.2 Multi-cue fusion. Note that we have multiple kinds of global visual feature (object, scene, action etc.). A straight-forward solution to fusing these features is applying late fusion, which is simply summing up cosine similarities before retrieval. This brings the final calculation of cross-modal similarity:

$$sim = \sum_{cue} 1 - \left\langle \frac{v_{cue}}{||v_{cue}||}, \frac{s}{||s||} \right\rangle,\ cue = \{object, scene, action\}. \tag{17}$$

Table 2: Retrieval result with temporal alignment module DSTA on YouTube and Movie subsets.

Method	r@1(%)↑	r@5(%)↑	r@10(%)↑	MedRank↓
Youtube				
DTW	1.0	3.7	6.5	392
DSTA	21.8	28.2	31.7	140
soft-DTW[10]	24.0	31.3	35.1	91
soft-DSTA	**36.2**	**45.2**	**49.2**	**12**
Movie				
DTW	0.1	0.2	0.6	972
DSTA	5.4	9.8	12.7	405
soft-DTW[10]	1.8	3.9	5.2	1028
soft-DSTA	**10.3**	**19.3**	**23.8**	**131**

5 EXPERIMENTS

5.1 Evaluation Metric

Since the annotation of a *plausible* video/text pair is based on subjective feelings, we adopt the original subtitle embedded in the video as the ground truth to evaluate quantatives performance. The metric we used is recall of top queries (r@1,5,10) and median rank (MedRank) for the original subtitle.

5.2 Retrieval with temporal alignment

The DSTA module is trained on *Movie* and *YouTube* subset separately. We select 4 video/subtitle pairs in a batch, which makes 16 times forwards of DSTA for 4 positive pairs and 12 negative pairs. During the computation of DSTA, we set window shift parameter k as $||l_v - l_s||$ so that the whole sequence can be counted in the forward loop. As for the order-sensitive penalty in Eq. (3), we set margin $m = 1$ and weight $\lambda_{ord} = 1$. As illustrated in Figure 4, a larger weight of λ_{ord} tends to learn an alignment with few backward shifts, otherwise allowing more misalignment. As for $\delta_{dur}(i,j)$ in Eq. (5), we set the average speech rate $\omega = 0.85$ and error tolerance parameter $\eta = 2$. Figure 5 shows that larger λ_{dur} pushes the alignment towards the diagonal of the cost matrix.

DSTA is trained for 40 epochs, the initial learning rate is 10^{-3} and decays by 0.1 after every 15 epochs. The margin in L^{dsta}_{rank} Eq. (14) is set to 0.1. The backbone is implemented in PyTorch, except that the computation procedure of front-end DSTA is written in Numba [22] for acceleration. we calculate the gradient matrix E in pre-compiled

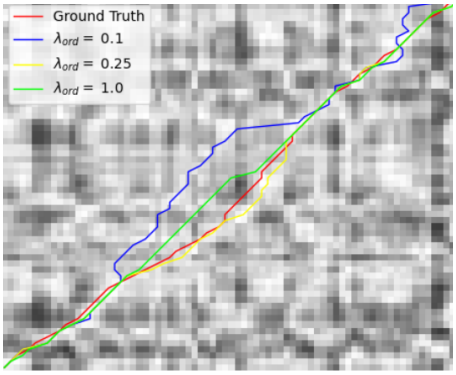


Figure 4: Effect of order-based term in Eq. (4).

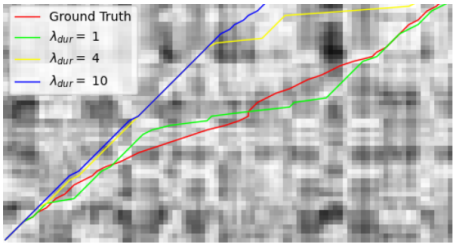


Figure 5: Effect of duration-based term in Eq. (5).

C loop and assign it into the auto-grad procedure of PyTorch. We use DTW and differentiable soft-DTW [10] as the baselines. Both DTW and DSTA are applied on the SynNet [30] pretrained features. As seen in Table 2, DSTA outperformed DTW with 70.5/27.0 for sum of recalls and 252/567 for median rank on both of the subset. We further compare the differentiable module trained with same $\gamma = 0.1$ and extra align loss L_{align} in Eq. (15). The results show that soft-DSTA still outperforms soft-DTW with 40.2/42.5 for sum of recalls and 79/897 for median rank on both subsets.

Figure 6 visualizes the alignments of DTW and DSTA. We highlight the zone near time $[t_1, t_2]$ in dashed box. Note that the similarity value stays identity along V-axis in the matrix. This means the video modal keeps stable temporally and is not very informative, possibly generated by face occlusion. DSTA successfully tackle this zone while DTW deviates the ground truth path seriously. This illustrates the robustness of DSTA when dealing with real-world cross-modal cases.

5.3 Retrieval with multiple cues

We train each cue's model with 40 epochs. The setting of learning rate is similar to DSTA's. The margin used for ranking loss L^{gsm}_{rank} is set to 0.2. The effect of global semantic cues is shown in Table 3. We use state-of-the-art cross modal retrieval model CLIP4clip [27] as the baseline. As seen, the global cue can provide moderate median ranks but relatively low top@(1,5,10) recalls, even using heavy-weight pretrained model [27]. Notably, the collaboration of global cues and soft-DSTA can elevate the performance by large margins from 3.9/2.7 to 43.8/13.6 for top@1 recall and 102/218 to 2/53 for median rank. The weight of each cue is empirically designed as

Table 3: Retrieval result with multi-cue fusion strategy on Youtube and Movie subset.

Cue	r@1(%)↑	r@5(%)↑	r@10(%)↑	MedRank↓
Youtube				
Object(O)	2.5	8.5	13.3	165
Scene(S)	2.9	8.6	13.4	204
Action(A)	2.5	7.7	12.4	202
soft-DSTA(D)	36.2	45.2	49.2	12
O+S+A	3.9	11.2	16.8	102
O+S+A+D	**43.8**	**58.9**	**65.1**	**2**
CLIP4clip[27]	7.4	19.3	27.3	51
Movie				
Object(O)	1.5	5.0	7.8	315
Scene(S)	1.6	4.4	7.0	395
Action(A)	1.1	3.7	6.2	452
soft-DSTA(D)	10.3	19.3	23.8	131
O+S+A	2.7	7.7	11.6	218
O+S+A+D	**13.6**	**24.8**	**30.9**	**53**
CLIP4clip[27]	5.2	11.8	16.3	196

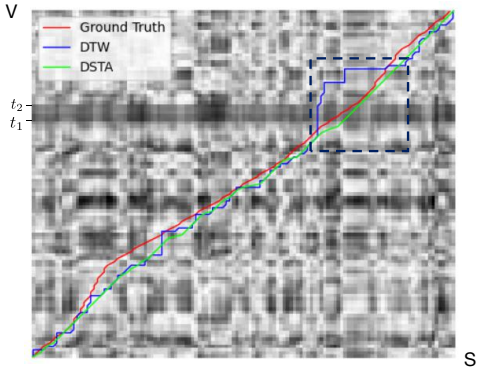
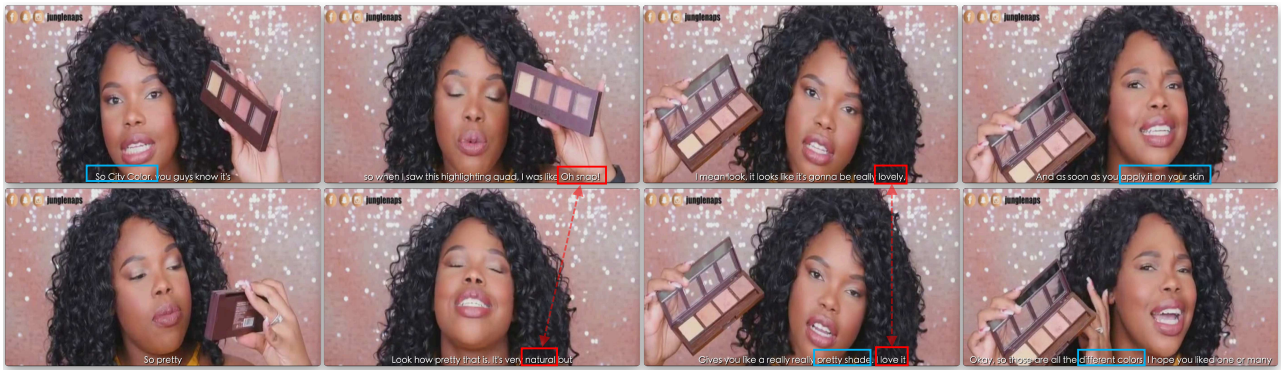


Figure 6: Alignment visualization. Regions in $[t_1, t_2]$ have identity values along V-axis, which means the lip is possibly not moving. DSTA performs more robustly than DTW.

$W_{object} = W_{action} = W_{scene} = 1, W_{DSTA} = 10$ without further fine-tuning.

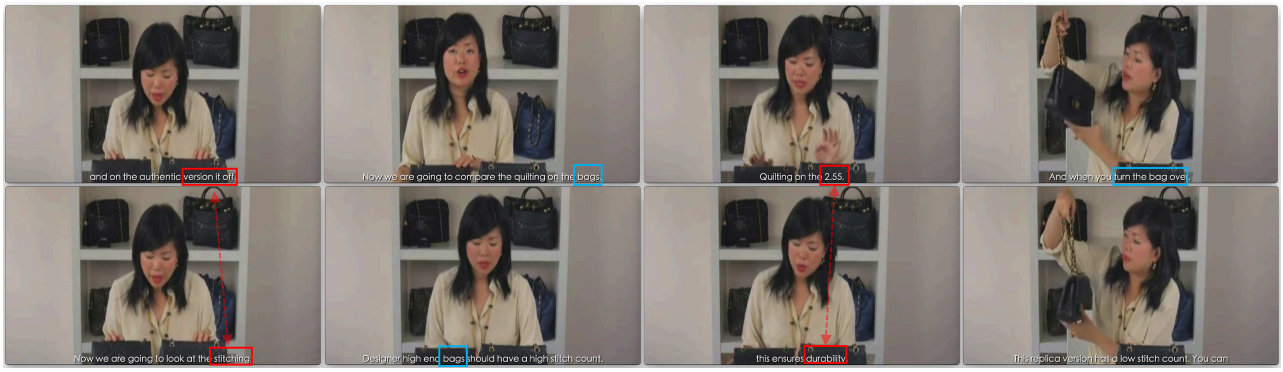
5.4 Visualization of re-generated subtitles

With our multi-cue matching method, the model is capable of finding a new subtitle which has similar semantic meanings and harmonized lip movements. We provide several visualizations of the re-generated subtitle in Figure 7. These cases can successfully find the original subtitle through multi-cue matching. We select the second ranked sample as the re-generated subtitle and demonstrate their detailed rankings of different cues. We can observe each cue plays different roles for different videos. The first case is talking about how to *make up* with *eye shadow*, so action and object scores are important. The second case is talking about business topics like *digital transaction*, thus the text content is abstract and irrelevant with concrete objects in the frame. But according to the scene arrangements (a TV studio), the re-generated subtitle is still



(a) Original: R_{object}:23, R_{scene}:9, R_{action}:0, R_{DSTA}:0 → R_{fusion}:0 **Re-Generated:** R_{object}:1, R_{scene}:21, R_{action}:9, R_{DSTA}:122 → R_{fusion}:1

(b) Original: R_{object}:12, R_{scene}:33, R_{action}:797, R_{DSTA}:14 → R_{fusion}:0 **Re-Generated:** R_{object}:191, R_{scene}:9, R_{action}:257, R_{DSTA}:40 → R_{fusion}:1



(c) Original: R_{object}:3, R_{scene}:181, R_{action}:23, R_{DSTA}:8 → R_{fusion}:0 **Re-Generated:** R_{object}:5, R_{scene}:10, R_{action}:17, R_{DSTA}:259 → R_{fusion}:1

Figure 7: Visualization of re-generated subtitles. Upper rows contain original subtitle and lower rows show the generated subtitle. The effect of temporal alignment DSTA is shown in red box. The effect of semantic similarity is shown in blue box.

constrained to talk about business and economic vocabularies like *group, revenue*. For better evaluating the re-generated subtitles, we also provide the synthesized video clips in the supplemental video presentation material.

6 CONCLUSION

We propose a new task in the domain of cross-modal retrieval, which is called Video2Subtitle. We establish a new benchmark to ensure not only synchronization of cross-modal temporal tracks, but also consistency of semantic information embedded in video and text. With the carefully collected new dataset, we propose a differentiable solution to this task. We hope the Video2Subtitle related techniques could enable some entertainment multimedia applications, *e.g.*, dubbing amateur videos with classic movie lines, transferring subtitles between different characters, etc.

7 ACKNOWLEDGEMENT

This work is supported by Science and Technology Innovation 2030 - New Generation Artificial Intelligence of China (2020AAA0104401) and Beijing Natural Science Foundation (Z190001).

REFERENCES

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. 2016. Youtube-8m: A large-scale video classification benchmark. *CoRR* (2016), 1609.08675.

[2] Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. 2013. Deep canonical correlation analysis. *International Conference on Machine Learning* (2013), 1247–1255.

[3] K Selçuk Candan, Rosaria Rossini, Maria Luisa Sapino, and Xiaolan Wang. 2012. sDTW: computing DTW distances using locally relevant constraints based on salient feature alignments. *International Conference on Very Large Data Bases* (2012).

[4] Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. *IEEE Conference on Computer Vision and Pattern Recognition* (2017), 6299–6308.

[5] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. 2020. Fine-grained video-text retrieval with hierarchical graph reasoning. *IEEE Conference on Computer Vision and Pattern Recognition* (2020), 10638–10647.

[6] Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *Advances in Neural Information Processing Systems Workshop* (2014).

[7] Joon Son Chung and Andrew Zisserman. 2016. Lip reading in the wild. *Asian Conference on Computer Vision* (2016), 87–103.

[8] Joon Son Chung and AP Zisserman. 2017. Lip reading in profile. (2017).

[9] Martin Cooke, Jon Barker, Stuart Cunningham, and Xu Shao. 2006. *The Grid Audio-Visual Speech Corpus.* https://doi.org/10.5281/zenodo.3625687

[10] Marco Cuturi and Mathieu Blondel. 2017. Soft-dtw: a differentiable loss function for time-series. *International Conference on Machine Learning* (2017), 894–903.

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition* (2009), 248–255.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *CoRR* (2018), 1810.04805.

[13] Jianfeng Dong, Xirong Li, Chaoxi Xu, Shouling Ji, Yuan He, Gang Yang, and Xun Wang. 2019. Dual encoding for zero-example video retrieval. *IEEE Conference on Computer Vision and Pattern Recognition* (2019), 9346–9355.

[14] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. 2017. Vse++: Improving visual-semantic embeddings with hard negatives. *CoRR* (2017), 1707.05612.

[15] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. 1999. Learning to forget: Continual prediction with LSTM. (1999).

[16] Jiabo He, Sarah Erfani, Sudanthi Wijewickrema, Stephen O'Leary, and Kotagiri Ramamohanarao. 2020. Segmented Pairwise Distance for Time Series with Large Discontinuities. *International Joint Conference on Neural Networks* (2020), 1–8.

[17] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. *IEEE Conference on Computer Vision and Pattern Recognition* (2017), 4700–4708.

[18] Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. 2020. MovieNet: A Holistic Dataset for Movie Understanding. *European Conference on Computer Vision* (2020).

[19] Justin Johnson, Agrim Gupta, and Li Fei-Fei. 2018. Image generation from scene graphs. *IEEE Conference on Computer Vision and Pattern Recognition* (2018), 1219–1228.

[20] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. 2017. The kinetics human action video dataset. *CoRR* (2017), 1705.06950.

[21] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. 2011. HMDB: A large video database for human motion recognition. *IEEE International Conference on Computer Vision* (2011), 2556–2563.

[22] Siu Kwan Lam, Antoine Pitrou, and Stanley Seibert. 2015. Numba: A LLVM-Based Python JIT Compiler. *Workshop on the LLVM Compiler Infrastructure in HPC*, Article 7 (2015), 6 pages.

[23] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L Berg. 2018. Tvqa: Localized, compositional video question answering. *CoRR* (2018), 1809.01696.

[24] Jie Lei, Licheng Yu, Tamara L Berg, and Mohit Bansal. 2020. TVR: A Large-Scale Dataset for Video-Subtitle Moment Retrieval. *CoRR* (2020), 2001.09099.

[25] Xirong Li, Chaoxi Xu, Gang Yang, Zhineng Chen, and Jianfeng Dong. 2019. W2VV++ Fully Deep Learning for Ad-hoc Video Search. *ACM International Conference on Multimedia* (2019), 1786–1794.

[26] Yuncheng Li, Yale Song, Liangliang Cao, Joel Tetreault, Larry Goldberg, Alejandro Jaimes, and Jiebo Luo. 2016. TGIF: A new dataset and benchmark on animated GIF description. *IEEE Conference on Computer Vision and Pattern Recognition* (2016), 4641–4650.

[27] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. 2021. CLIP4Clip: An Empirical Study of CLIP for End to End Video Clip Retrieval. *CoRR* (2021), 2104.08860.

[28] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. 2019. Howto100M: Learning a text-video embedding by watching hundred million narrated video clips. *IEEE International Conference on Computer Vision* (2019), 2630–2640.

[29] Niluthpol Chowdhury Mithun, Juncheng Li, Florian Metze, and Amit K Roy-Chowdhury. 2018. Learning joint embedding with multimodal cues for cross-modal video-text retrieval. *ACM International Conference on Multimedia Retrieval* (2018), 19–27.

[30] Arsha Nagrani, Joon Son Chung, Samuel Albanie, and Andrew Zisserman. 2020. Disentangled Speech Embeddings using Cross-Modal Self-Supervision. *International Conference on Acoustics, Speech, and Signal Processing* (2020).

[31] A. Nagrani, J. S. Chung, and A. Zisserman. 2017. VoxCeleb: a large-scale speaker identification dataset. *Conference of the International Speech Communication Association* (2017).

[32] Tae-Hyun Oh, Tali Dekel, Changil Kim, Inbar Mosseri, William T Freeman, Michael Rubinstein, and Wojciech Matusik. 2019. Speech2face: Learning the face behind a voice. *IEEE Conference on Computer Vision and Pattern Recognition* (2019), 7539–7548.

[33] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. *Conference on Epirical Methods in Natural Language Processing* (2014), 1532–1543.

[34] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. 2016. Generative adversarial text to image synthesis. *CoRR* (2016), 1605.05396.

[35] Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Christopher Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. 2017. Movie description. *International Journal of Computer Vision* 123, 1 (2017), 94–120.

[36] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019. Videobert: A joint model for video and language representation learning. *IEEE International Conference on Computer Vision* (2019), 7464–7473.

[37] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2016. MovieQA: Understanding Stories in Movies through Question-Answering. *IEEE Conference on Computer Vision and Pattern Recognition* (2016).

[38] Bruce Thompson. 2005. Canonical correlation analysis. *Encyclopedia of statistics in behavioral science* (2005).

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* (2017), 5998–6008.

[40] Yuxuan Wang, RJ Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, Zhifeng Chen, Samy Bengio, et al. 2017. Tacotron: Towards end-to-end speech synthesis. *Conference of the International Speech Communication Association* (2017).

[41] Zuxuan Wu, Yu-Gang Jiang, Xi Wang, Hao Ye, and Xiangyang Xue. 2016. Multi-stream multi-class fusion of deep networks for video classification. *ACM International Conference on Multimedia* (2016), 791–800.

[42] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. *IEEE Conference on Computer Vision and Pattern Recognition* (2016), 5288–5296.

[43] Ran Xu, Caiming Xiong, Wei Chen, and Jason J Corso. 2015. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. *AAAI Conference on Artificial Intelligence* (2015).

[44] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. 2017. End-to-end concept word detection for video captioning, retrieval, and question answering. *IEEE Conference on Computer Vision and Pattern Recognition* (2017), 3165–3173.

[45] Zheng Zhang, Romain Tavenard, Adeline Bailly, Xiaotong Tang, Ping Tang, and Thomas Corpetti. 2017. Dynamic time warping under limited warping path length. *Information Sciences* 393 (2017), 91–107.

[46] Jiaping Zhao and Laurent Itti. 2018. shapedtw: Shape dynamic time warping. *Pattern Recognition* 74 (2018), 171–184.

[47] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2017. Places: A 10 million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2017).