Temporal Modular Networks for Retrieving Complex Compositional Activities in Videos

Bingbin Liu[1](✉), Serena Yeung[1,2], Edward Chou[1], De-An Huang[1], Li Fei-Fei[1,2], and Juan Carlos Niebles[1,2]

[1] Stanford University, Stanford, CA 94305, USA
bingbin@stanford.edu
[2] Google Cloud AI, Mountain View, CA 94043, USA

Abstract. A major challenge in computer vision is scaling activity understanding to the long tail of complex activities without requiring collecting large quantities of data for new actions. The task of video retrieval using natural language descriptions seeks to address this through rich, unconstrained supervision about complex activities. However, while this formulation offers hope of leveraging underlying compositional structure in activity descriptions, existing approaches typically do not explicitly model compositional reasoning. In this work, we introduce an approach for explicitly and dynamically reasoning about compositional natural language descriptions of activity in videos. We take a modular neural network approach that, given a natural language query, extracts the semantic structure to assemble a compositional neural network layout and corresponding network modules. We show that this approach is able to achieve state-of-the-art results on the DiDeMo video retrieval dataset.

Keywords: Video retrieval · Action recognition · Modular networks

1 Introduction

A fundamental goal of computer vision is understanding rich, diverse and complex activities occurring over time in a dynamic visual world. While there has been significant progress in activity recognition, it is often restricted to a constrained setting with a fixed number of action classes for each particular dataset [1,6,22,25,26,28,33,51,66]. Scaling these recognition models to the long tail of complex activities is still an open problem in this paradigm, as it requires collecting large quantities of data for new action classes and does not explicitly exploit similarity between activities.

To address this problem, a natural solution is to describe complex activity in natural language [5,7,39,44,59]. This allows for supervised labels containing rich, unconstrained information about the activity, and motivates tasks such as video retrieval [16,47,52,55]. This formulation also gives hope of leveraging

V. Ferrari et al. (Eds.): ECCV 2018, LNCS 11207, pp. 569–586, 2018.
https://doi.org/10.1007/978-3-030-01219-9_34

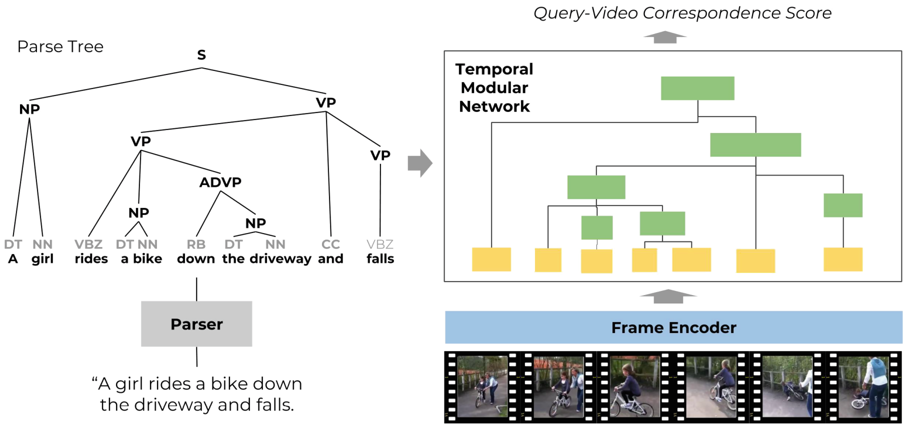

Query-Video Correspondence Score
Parse Tree
S
NP
VP
VP
VP
ADVP
NP
NP
DT NN VBZ DT NN RB DT NN CC VBZ
A girl rides a bike down the driveway and falls
Parser
"A girl rides a bike down the driveway and falls.
Temporal Modular Network
Frame Encoder

Fig. 1. Given a natural language query and video as input, Temporal Modular Networks (TMN) uses the underlying language structure of the query to dynamically assemble a corresponding modular neural network that reasons compositionally over the video to produce a query-video correspondence score.

the underlying structure in the activity description in order to reuse learned sub-concepts across activities. The approach we use endows models with an increasingly compositional structure. For example, a complex concept like "girl riding a bike down a driveway then falling" can be decomposed into two sub-events "riding" and "falling" which can be observed and learned in very different contexts (riding a bike vs. a skateboard, down a driveway vs. down a hill).

In this work, we focus on the *natural language video retrieval task*. Given an input in the form of natural language description, the goal is to retrieve the best matching video. The variety of the language descriptions and visual appearances makes it a challenging task beyond the classification of predefined action categories. Existing video retrieval methods typically learn embedding representations of language and video using recurrent neural networks [9,14,16,61,63] or spatio-temporal convolutions [21,25,54]. While simple and effective, these approaches fail to capture, and more importantly, leverage, the inherently compositional structure of the concepts and fail to properly relate each sub-concept for efficient reasoning. We posit that explicitly modeling compositional structure is key for the generalizability and scalability needed for complex video understanding.

To this end, we introduce a dynamic compositional approach for reasoning about complex natural language descriptions of activity in videos. We draw inspiration from recent successes in visual question answering using compositional models [2,3,17,18,24,58]. Given a natural language query and a video, our approach explicitly makes use of the underlying language structure of the query to dynamically (and hierarchically) assemble a corresponding modular network to reason over the video, and output the correspondence between the query and the video (Fig. 1). More specifically, we use a natural language parser to extract a structure from the description. Using this structure, we construct a hierarchical layout based on which corresponding neural network modules are assembled. Because the modules are reused across different queries, we can jointly learn the module parameters across these queries and their corresponding videos to enable efficient learning and scaling to diverse concepts.

Our contributions are as follow:

- We propose a new model called Temporal Modular Networks that explicitly uses the compositionality in natural languages for temporal reasoning in videos.
- We demonstrate that by leveraging this additional structure, our model is able to achieve state-of-the-art results on DiDeMo [16], a diverse dataset for localizing free-form queries in videos.

2 Related Work

There is a large body of work on the problem of activity recognition in videos [1,6,9,21,22,25,26,35,37,43,45,51,53,54,56,57]. However, the majority of these have focused on recognizing a fixed set of activity classes with large numbers of labeled data [1,6,22,25,26,51], which is not a practical paradigm for scaling to the large number of long-tail and complex activities. In this section, we focus the discussion on work that tackles the challenge of scaling through zero-shot, compositional, and natural language-based approaches.

Zero-Shot Action Recognition. Zero-shot approaches seek to avoid the need of training examples for every class of interest. This is related to our work as a popular approach is to use the word embedding as the representation of the class to achieve zero-shot learning [11]. A popular direction is to leverage links other than visual cues to recognize a large number of novel classes given a smaller number of known ones. [20,29,30] draw links between actions and objects. [65] uses attributes such as duration and dynamics for each verb, and predicts unseen verbs jointly from these attributes and semantic embedding. [41] takes a similar approach, but instead uses a bank of simpler actions to describe more complex meta-actions. Our approach is related to the zero-shot setting in the sense that it can extend to previously unseen descriptions by leveraging the language structure to compose the network, whose base module can also be seen as a zero-shot model for detecting visual concept based on the word.

Compositional Action Recognition. Methods for compositional action recognition have taken the approach of defining actions using a set of atomic actions or objects. This includes interpreting an action as a sequence of poses with a part-based model on body segments [19,31,34], or as composed of a set of action primitives [10,12,13,64]. Compositional action recognition methods are useful specially for instructional videos, with clearly defined instruction sequences that are naturally compositional [38,40,42,67]. For example, Rohrbach et al. [40] applies a hand-centric pose estimation technique to recognize fine-grained activities, using which complex cooking activities are then composed.

Compositionality Through Natural Language. A complementary way to model complex concepts is at the higher level of unconstrained natural language, which is inherently compositional. Related to action recognition, a natural setting is video retrieval [1,6,16,22,25,26,28,33,51,66]. While most of these works use recurrent neural networks for language encoding [14,61,63], more explicit compositional and hierarchical reasoning has recently been used, such as in the setting of visual question-answering in images (VQA [4]). These build off previous work relating language structure to visual scenes in images [48,49]. [60] uses a two-layer stacked attention network, and demonstrates that this hierarchical structure allows the first layer to focus on scattered objects which are then aggregated by the second layer. [32] shares a similar structure, but defines the hierarchy based on the word-phrase-sentence structure of natural languages, and calculates attention at each level independently to avoid error propagation. Xiao [58] follows a parsed language structure more closely, and adds two types of structural losses to constraint attentions at different nodes. Our work builds on these directions of using explicit compositional reasoning based on natural language, and extends to the video domain for the retrieval task.

The idea of leveraging language structure naturally points to related work in natural language processing, such as Recursive Neural Networks [49,50]. While these works have laid the foundation of tree-structured reasoning, our work differs from them in two key aspects. First, our work uses instance-dependent modules that are parameterized by specific queries, while the computation units in recursive neural networks remain the same for all instances. Second, as mentioned earlier, our work focus on the adaptation to the video domain which has remained unexplored. In particular, [49] works on semantic segmentation, and [50] learns compositionally aggregated semantic features, which are setting rather disparate from ours.

Modular Neural Networks. Recently, there have been approaches to image question-answering that model compositionality through dynamic neural network layouts. [3] proposes modular neural networks which composes reusable modules using layouts output by a natural language parser. To overcome the limitations of a fixed parser, [2] reassembles subsets of modules to obtain a list of candidate layouts, from which it selects the best one using reinforcement learning. [17] takes a step further to explore a wider layout space, while still using parser output as “expert policies” for supervised learning at the initial learning stage. Finally, [24] instead learns a program generator to predict the network layout. However, these works work on image-question answering where queries and modules have structures with limited variations, and the images often come from synthetic datasets such as CLEVR [23]). For a more realistic setting, [18] applies compositional modular networks to real-world images with free-form queries, but as a trade-off, it only uses a fixed triplet structure. In contrast, our work adapts the modular approach to the video domain, and works on video retrieval with natural language. In order to handle the diversity in natural language descriptions of complex activity, we leverage a language parser for network structure, and introduce modular network components suitable for handling diverse activity descriptions for videos. To the best of our knowledge, our work is the first to explore dynamic modular networks for free-form, language-based reasoning about videos.

3 Temporal Modular Networks

In this work, we address the natural language video retrieval task. Given an input sentence, the goal is to retrieve the best corresponding video. Our key observation is that there is an underlying structure in the natural language description that plays an essential role in the compositional understanding of the corresponding video. Based on this intuition, we propose Temporal Modular Networks (TMN), a novel framework for compositional reasoning of complex activities in videos that takes a natural language description and a video as input, and outputs scores indicating the correspondence between sub-videos and the description. Our method uses dynamically-assembled neural modular networks to explicitly model the compositional structure of diverse and complex natural language description of the activity, which is in contrast to previous work where language and visual embedding are performed in separation.

In Sect. 3.1, we first describe how we leverage natural language parsing to transform diverse descriptions into tree structures compatible with compositional reasoning. In Sect. 3.2, we then present how, for any given description, we can use these tree structures to dynamically assemble a corresponding modular neural network over the video. The assembled networks explicitly model the compositionality in natural language descriptions, and we refer to these as *Temporal Modular Networks (TMN)*. Finally, in Sect. 3.3 we explain how we jointly learn the module components of TMN given pairs of descriptions and corresponding videos.

3.1 Transforming Phrases into Compositional Structure

Given a natural language description of a complex activity, we need to first decompose this description into a compositional structure. While there exist approaches that model constrained forms of compositional activity description and structure, our goal is to enable reasoning over rich and unconstrained natural language descriptions of activity.

We therefore use a natural language parser to extract structures from arbitrary descriptions. Natural language has inherent structures in the form of word-phrase-sentence hierarchies, and natural language parsers formalize this through parse trees. In particular, we use the Stanford Parser [27], a probabilistic context-free grammar parser, to obtain grammatical relationships between words in the description and to obtain an initial parse tree with part-of-speech (POS) tags.

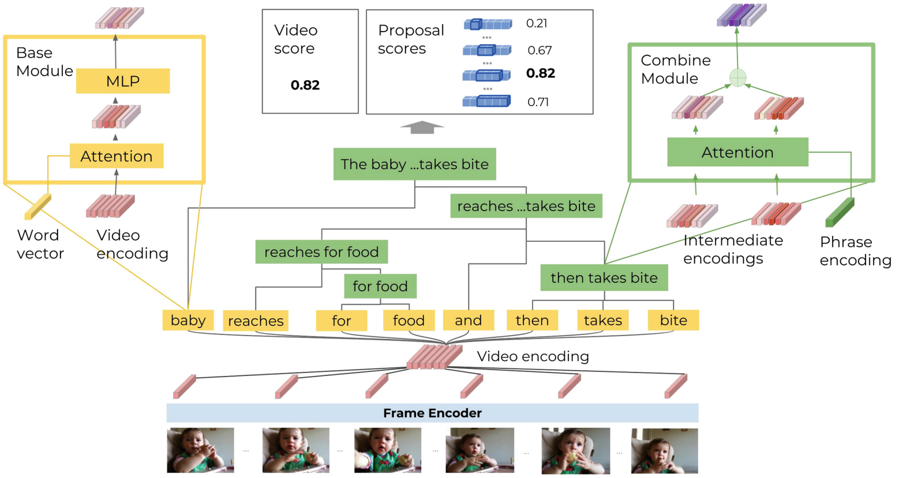


Fig. 2. Temporal modular network architecture. The compositional layout of the network is determined by structure extracted from a natural language parser (Sect. 3.1). *Base modules* (yellow) in the network reason directly over the temporal sequence of frame-level visual encodings for a video, based on node-specific word embeddings. *Combine modules* (green) combine information from child nodes, based on node-specific higher-level phrase embeddings. The output of the top-level combine module is used to produce, depending on the setting, a score corresponding to the strength of the video match with the query, or scores of temporal proposals within the video. See Sect. 3.2 for more details. (Color figure online)

The choice of a constituency parser over a dependency parser comes from the fact that a dependency parser is designed to be invariant to syntactic structure, while a constituency parser captures the syntactic structure which represents the language compositionality that we desire [48]. Sequential events, for example, is not clearly presented in a dependency parse tree. For the description "girl riding a bike then falling" which includes two sequential actions "riding" and "falling", a dependency parser would treat the second action "falling" as a dependent of the first action "riding", resulting in a misleading hierarchy, whereas a constituency parser gives a parallel structure over the two (verb) phrases.

While a parser provides an initial compositional structure, some POS tags neither represent nor relate visual concepts, such as *DT* (determiner) and *RP* (particle). We therefore discard these elements from the parse tree. We furthermore merge tags that differ in tenses or pluralities but belong to the same word class. For example, *VBZ* (verb, third-person singular present) and *VBD* (verb, past tense) are merged as *VB* (verb, base form). Table 1 specifies the POS tag mapping. After merging and discarding, the total number of POS tags appearing in a tree is reduced from 36 to 8.

Table 1. Part-of-speech (POS) tag mapping from those output by the natural language parser those in the processed compositional trees. The original POS tag(s) corresponding to each mapped tag are listed.

Mapped tag	Description	Original tag(s)
CC	Coordinating conjunction	CC
FW	Foreign word	FW
IN	Preposition or subordinating conjunction	IN
JJ	Adjective	JJ, JJR, JJS
NN	Noun	NN, NNS, NNP, NNPS, PRP
RB	Adverb	RB, RBR, RBS
TO	To	TO
VB	Verb	VB, VBD, VBG, VBN, VBP, VBZ

Then, nodes in the resulting tree can be categorized into two types: *base nodes* that correspond to single words in a description, and *combine nodes* which correspond to phrases (sequences of words) and combine its child nodes.

3.2 Dynamically Assembling Compositional Networks over Video

We have described in Sect. 3.1 how we can use natural language parsing to obtain inherent compositional structures from arbitrary descriptions of complex activities. The challenge at hand then becomes how we can use this structure to perform compositional reasoning in videos. Our key insight is that we can leverage this language structure to modularize the corresponding video understanding network for modeling the structure of the activity.

Our modular approach, which we call Temporal Modular Networks (TMN), reasons about a natural language description paired with a video with a dynamically assembled modular network. A set of neural network modules are used to represent nodes in the description's corresponding compositional tree. The complete network connects these composable modules following the tree structure (Fig. 2).

We use two types of modules, namely base modules and combine modules, corresponding respectively to the two types of nodes in the structure described in Sect. 3.1. The lower-level *base modules* reason directly over video features, while higher-level *combine modules* operate on the outputs from child modules. Intuitively, the base module is used to detect atomic visual concepts described by the words, and the combine module learns to gradually combine the visual information flowing from its child modules. Our modular design allows us to share parameters in each type of the modules. Following, we describe base and combine modules in more detail, how they operate over temporal video data, as well as how to obtain correspondence scores between queries (i.e. natural language descriptions) and parts of videos for intra-video retrieval.

Base Modules. Base modules correspond to the base nodes in a compositional tree (Fig. 2). Each base module takes as input a temporal sequence of segment-level visual encoding of a video, $M^{\text{in}} \in \mathbb{R}^{D_v \times n}$, and the word embedding $v_w \in \mathbb{R}^{D_w}$ of a single word corresponding to the module. Here D_v is the dimension of the visual encoding, D_w is the dimension of the word embedding, and n is the length of the temporal sequence. Intuitively, we would like the module to encode the semantic presence of the word in the video. The base module therefore first produces a temporal attention vector based on the word embedding and the visual encoding following [60], and then passes the temporally attended feature map through a multi-layer perceptron. The output feature map M^{out} may be of arbitrary dimension but we choose it to be the same as the input dimension, and formally compute it as:

$$\begin{aligned} h_{\text{att}} &= \tanh(W_v M^{\text{in}} \oplus (W_w v_w + b_w)) \in \mathbb{R}^{k \times n_{seg}} \\ a &= \text{softmax}(W_a h_{\text{att}} + b_a) \in \mathbb{R}^{n} \\ M^{\text{att}} &= a \odot M^{\text{in}} \in \mathbb{R}^{D_v \times n} \\ M^{\text{out}} &= \text{MLP}(M^{\text{att}}) \in \mathbb{R}^{D_v \times n} \end{aligned} \tag{1}$$

Here k is the dimension of the common embedding space which the visual encoding and the word vector are mapped into, and $W_v \in \mathbb{R}^{k \times D_w}$ and $W_w \in \mathbb{R}^{k \times D_v}$ are the embedding matrices of the visual encoding and word vector, respectively. $\oplus$ denotes matrix-vector addition where the vector is added to each column of the matrix. $W_a \in \mathbb{R}^{1 \times k}$ maps h_{att} to a vector of length n, the temporal length of the sequence, which is then normalized by softmax to produce the temporal attention weights. b_w and b_a are bias terms. $\odot$ denotes matrix-vector multiplication that multiplies the i_{th} column of the matrix with the i_{th} entry of the vector. Finally, the attended feature map M^{att} is passed through a multi-layer perceptron to produce the output M^{out}.

Combine Modules. Combine modules correspond to the combine nodes of a compositional tree, whose function is to combine child feature maps to pass the information upwards in the compositional hierarchy. The flexible structure of parse-based compositional trees means that the combine module may have a variable arity (i.e. number in children). This contrasts with previous modular network approaches in settings where the arity of modules is fixed [3,24], or where the number of children is expected to be within a predefined limit [48]. To handle this, the combine modules iteratively combine adjacent child feature maps. Given a pair of child feature maps $M^a, M^b \in \mathbb{R}^{D_v \times n}$, a combine module computes an attention vector $a \in \mathbb{R}^n$ parameterized by the encoding of the module's corresponding natural language phrase, indicating desired relative weighting in combining M^a vs. M^b at each temporal segment. Formally, the output of a combine module with C children is computed iteratively as:

$$\begin{cases} M^{1^*} = M^1 \\ M^{c^*} = a \cdot M^{(c-1)^*} + (1-a) \cdot M^c, 1 < c < C \\ M^{\text{out}} = M^{C^*} = a \cdot M^{(C-1)^*} + (1-a) \cdot M^C \end{cases} \tag{2}$$

Here M^c is the feature map of the c_{th} child, and M^{c^*} is the feature map aggregated over children 1 to c. The output feature map is the aggregated feature map from the last child, i.e. $M^{\text{out}} = M^C$. This iterative formulation allows us to handle a variable module arity.

The attention vector $a \in \mathbb{R}^n$ weighting the combination of two child feature maps $M^a, M^b \in \mathbb{R}^{D_v \times n}$ is computed from the feature maps and the combine module's corresponding phrase encoding $v_p \in \mathbb{R}^{D_p}$ as follows:

$$\begin{aligned} h_p &= W_p v_p + b_p \in \mathbb{R}^{D_v} \\ h_1, h_2 &= h_p^T M^a, h_p^T M^b \in \mathbb{R}^n \\ h_{\text{weight}} &= \text{softmax}([h_1, h_2], \text{dim} = 1) \in \mathbb{R}^{n \times 2} \\ a,\ 1 - a &= h^0_{\text{weight}},\ h^1_{\text{weight}} \in \mathbb{R}^n \end{aligned} \tag{3}$$

where $W_p \in \mathbb{R}^{D_v \times D_p}$ and $b_p \in \mathbb{R}^{D_v}$ are weight and bias terms for embedding the phrase encoding v_p to a common embedding space with the visual encoding. In practice, we use a bag-of-words representation where a phrase encoding is obtained by averaging of the word vectors in the phrase. h_1, $h_2 \in \mathbb{R}^n$ represent affinity scores between the phrase encoding and each dimension of child feature maps M^a and M^b, which are then stacked into a $\mathbb{R}^{n\times 2}$ matrix and normalized per-dimension as h_{weight}. Finally, attention vectors a and $1 - a$, taking from the two columns of h_{weight}, provide the relative weights of M^a and M^b in their combination by each temporal segment.

Query Scores. The output feature map of the highest level combine module is used to compute the correspondence scores between parts of the video and the query through two fully connected layers. The retrieval task we are addressing is the *intra-video setting*, where the goal is to localize the best matching temporal moment within a video. We therefore wish to output scores for each sub-video (temporal proposal) of variable length. Given that the input video has temporal length n, the network will first regress n correspondence scores for each temporal segment, and then combine the scores of consecutive segments to produce $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$ scores for all possible sub-videos. The sub-video with the maximum score is predicted as the best match for intra-video retrieval. Note that when combining the scores, TMN uses the sum rather than the average to avoid outputting scattered segments and to encourage longer sub-videos, which is in a similar spirit to [15] and gives a significant enhancement on rank-5 accuracy. Moreover, the scores may take negative values; thus longer sub-videos are not always more favorable. This scoring scheme can easily generalize the video retrieval task to the *inter-video setting*, where the goal is to retrieve the best matching video from a set of candidate videos. In this case, the correspondence score for a video can simply be chosen as the max score among all sub-videos.

3.3 Training

Our goal is to learn the parameters of the base and combine modules, as well as the scoring layers at the root, which can be jointly learned given pairs of natural language queries and corresponding videos. Training is performed on minibatches of query-video pairs, where one example in the minibatch is the correct pair and the remaining incorrect. In each batch, inter-video negative examples encourage the modules to distinguish between various scene semantics, while intra-video negatives encourage the modules to focus on learning temporal concepts.

The network is trained end-to-end using a ranking loss function, which is defined as

$$\mathcal{L}_{\text{rank}} = \sum_{i \in N} \max(0, s_i - s^* + b) \tag{4}$$

where N is the set of all possible negative clips, s_i is the score of negative clip i, s^* is the predicted score of the ground truth clip, and b is a margin. While the model can also be trained using binary cross-entropy (BCE) loss, ranking loss is more effective for ous intra-video setting. For example, an inter-video negative with unrelated content should be scored lower than an intra-video negative which contains the best matching video segment but is not chosen optimal match by not being temporally tight, which is a nuance that the BCE loss fails to capture.

4 Experiments

We evaluate our approach for compositional reasoning of complex activities on the task of intra-video retrieval. Given an input natural language description, the goal is to locate the best corresponding sub-video. We posit that explicitly modeling the compositional structure is key to the success of this task. Specifically, we show that under the intra-video retrieval setting, the proposed temporal modular networks can achieve state-of-the-art results on DiDeMo dataset [16]. Here intra-video means the retrieval is within a single video, where given an input query-video pair, the network is expected to temporally locate the query within the video. We use this setting since the subjects and scene in a short (here, 25 to 30 seconds long) video are often unchanged, which ensures that the network must indeed learn to perform temporal reasoning, rather than relying on other information such as objects or scene which may contain strong priors [20,29,30], rendering the task more challenging.

4.1 Implementation Details

We represent a video by a temporal sequence of segment-level visual encoding as described in Sect. 4.2. The Stanford Parser [27] is used to obtain the initial parse trees for the compositional structure. For word vectors as part of the base module input, we use the 300-dimensional GloVe [36] vectors pretrained on Common Crawl (42 billion tokens). For the combine modules, a bag-of-words model is used to generate a fixed-size representation of the corresponding phrase. We use Adam optimizer [8] in all experiments with an initial learning rate of 5e−6 and a weight decay varying from 5e−5 to 3e−7.

4.2 Dataset

We use the DiDeMo [16] dataset which consists of 26,892 videos, each of 25 or 30 seconds and randomly selected from YFCC100M [5]. There are 33,005 video-query pairs in the training set, 4180 in the validation and 4021 in the test set. A video may appear in multiple query-video pairs with different queries matched to different sub-videos. DiDeMo is especially suitable for the intra-video setting, since it desirably offers referring expressions temporally aligned with parts of videos, as opposed to [7,39,59,62] where descriptions are at the video level.

For intra-video retrieval, each video in DiDeMo is divided into 6 segments of 5 seconds long each, and the task is to select the sub-video that best matches the query. Each sub-video contains one or more consecutive segments. In total there are 21 possible candidates for each query, corresponding to 6 single-segment sub-videos, 5 two-segment sub-videos, and so on. Performance is measured by rank-1 accuracy ($rank$@1) and rank-5 accuracy ($rank$@5) for prediction confidence, which is the percentage of examples where the best matches are ranked respectively as top 1 or among top 5, as well as segment-level mean intersection-over-union ($miou$) for temporal precision.

Quantitative Results. Table 2 shows results comparing TMN with Moment Context Network (MCN), the state-of-the-art approach introduced in [16]. For fair comparison, we use the same RGB, flow, and fused features as provided by [16]. The features are extracted from VGG [46] *fc7* and are average-pooled over frames to produce a 4096-d vector for each segment. A video is hence represented by a temporal sequence of 6 feature vectors. We do not compare with the temporal endpoint features in [16], as these directly correspond to dataset priors and do not reflect a model's temporal reasoning capability. It can be seen that TMN outperforms MCN [16] across all modalities, with significant improvements on $rank$@1 and $rank$@5 accuracy and comparable performance on mean IoU.

In contrast to MCN which uses an LSTM for language encoding and outputs matching scores based on the distance between language and visual embedding, the explicit compositional modeling in TMN is crucial for performance gain for all types of features. Interestingly, while MCN had noticeably lower performance on RGB features, TMN is able to large bridge the performance gap to optical flow features. Since optical flow provides additional motion information over RGB, this gain highlights TMN's strong ability to perform compositional reasoning over temporal video even when features contain weaker motion information. The combination of both RGB and flow features ("fused") further boosts the performance of TMN as expected. Moreover, when the base module and combine module are sequentially applied to each word, the network functions similarly to a recurrent neural network. Therefore, the performance gain of TMN showcases the importance of an appropriate compositional structure.

Qualitative Results. One advantage of a compositional network is its interpretability. Figure 3 visualizes the hierarchical pattern in the temporal attentions

Table 2. TMN outperforms MCN using RGB, flow, and fused (RGB+flow) features. The significant gain on RGB features in particular shows TMN's ability of temporal reasoning without relying on motion information in features.

Feature	Model	Rank@1	Rank@5	mean IoU
RGB	MCN	13.10	44.82	25.13
	TMN	**18.71**	**72.97**	**30.14**
Flow	MCN	18.35	56.25	31.46
	TMN	**19.90**	**75.14**	**31.95**
Fuse	MCN	19.88	62.39	33.51
	TMN	**22.92**	**76.08**	**35.17**

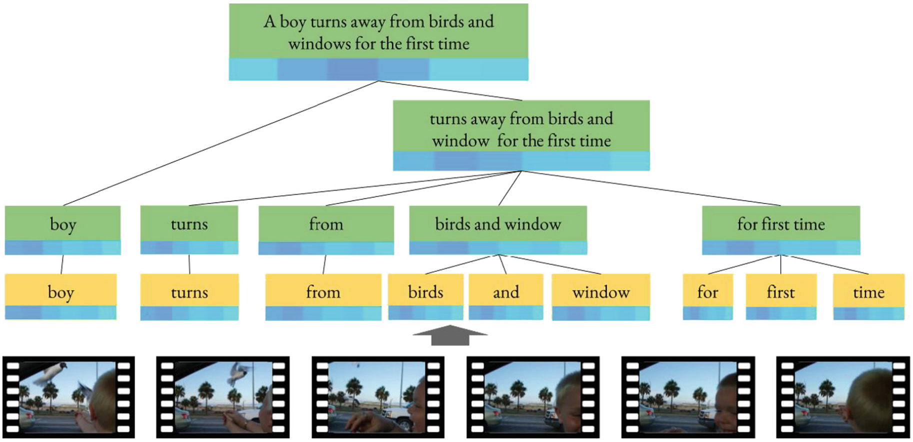
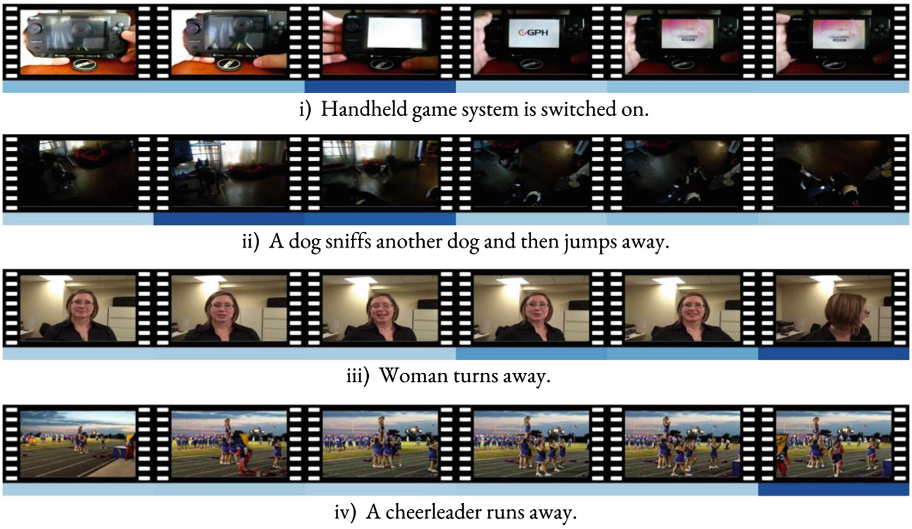

A boy turns away from birds and windows for the first time
turns away from birds and window for the first time
boy
turns
from
birds and window
for first time
boy
turns
from
birds
and
window
for
first
time

Fig. 3. Qualitative example of TMN evaluated on a query-video pair. Attention maps at base and combine nodes are visualized, where the color bars represent attention weights with darker blue indicating higher weights. Attention maps for the base nodes show the activation of a single word, while the attention map in a combine node shows how information from children are aggregated, and how the modules specifically take in the temporal quality ("for the first time") of the phrase encoding in each module.

generated by each combine module, which means the network learns to aggregate information correctly. Figure 4 provides more example outputs. It can be seen that the advantage of TMN is best pronounced for tasks that rely on the temporal dimension.

Ablation Study. We perform ablation studies to investigate the variations in module design, network structures, and loss functions:

- *Type of base modules*: We experimented with two types of base module: the *POS* setting with one base module per POS tag, and the *Single* setting where a single based module is shared across all tags. The *POS* setting may ease



i) Handheld game system is switched on.

ii) A dog sniffs another dog and then jumps away.

iii) Woman turns away.

iv) A cheerleader runs away.

Fig. 4. Example outputs of TMN, where TMN is able to recognize temporal changes such as "*switch on*" and "*turn/run away*", as well as compositional relations such as "*then*"

the learning for TMN by making each module more specialized, whereas the *single* setting allows TMN to learn from larger amounts of data and may help capture patterns existing across POS tags. For example, a single shared module may be similarly parameterized by words with different POS tags but appearing in a similar context. Moreover, using a single module was shown to be more robust since it provides better tolerance towards the parser error, which sometimes mistakenly assigns a noun tag to a singular form verb. The *single* setting was chosen based on our experimental results.

- *Attention in combine module*: In addition to having the combine module selectively attend to different temporal segments based on the phrase encoding, we also considered max pooling as a simplified alternative to combining multiple child feature maps, where the output feature map is element-wise max pooled from all children. This is inspired by the parent-child constraint in [58], where a structure loss is used to penalize the combined feature map from deviating from the union of child feature maps, which is essentially approximating a max pool layer. Formally, the combined feature map is defined such that $\forall i \in \{1 \dots n\},\ j \in \{1 \dots D_v\}$,

$$M^{out}_{i,j} = \max_{c \in C} M^c_{i,j} \tag{5}$$

where C is the set of children and M^c is the feature map of the c_{th} child.

- *Effect of a proper compositional network structure*: We compared three network structures. The first one was without the compositional tree. Since TMN resembles a vanilla RNN when the compositional structure is taken away, the performance gap between MCN [16] and TMN corresponds to the gain of the compositional structure. The other two structures came from a dependency parser and a constituency parser. We found out that structures from both parsers were able to outperform MCN, demonstrating the importance of compositional reasoning. Further, the performance gap between the two parse structures shows the advantage of a proper structure.

- *Choice of loss function*: We trained TMN with both a ranking loss and a binary cross entropy (BCE) loss. The performance gain of ranking loss over BCE loss verifies our hypothesis that the intra-video setting poses additional requirement on temporal localization, which is better coped with relative ranking rather than absolute scores.

Table 3. Ablation study for effectiveness of TMN components: *Lines 1 & 2:* effectiveness of a compositional structure *Lines 3 & 4, 5 & 6*: advantage of ranking loss over BCE loss. *Lines 3 & 5, 4 & 6*: importance of a proper compositional structure.

#id	Model	Rank@1	Rank@5	mean IoU
1	MCN [16] (i.e. no tree structure)	19.88	62.39	33.51
2	const + max pool + rank loss	21.89	75.69	34.24
3	dep + combine attention + BCE loss	20.41	75.38	32.86
4	dep + combine attention + rank loss	21.67	75.98	33.94
5	const + combine attention + BCE loss	21.60	75.81	34.40
6	const + combine attention + rank loss	**22.92**	**76.08**	**35.17**

Table 3 shows ablation results, where *max pool* and *combine attention* analyzes the effect of attention in combine modules, *const* and *dep* refer to structures given by a constituency parser and a dependency parser respectively, and *rank loss* and *BCE loss* compare the choice of loss functions.

5 Conclusions

In this work, we introduced Temporal Modular Networks (TMN), a compositional approach for temporal reasoning in videos through dynamically assembled modular networks. We demonstrated the effectiveness of this approach on the DiDeMo dataset [16] under the intra-video retrieval setting. We believe our work is a first step that highlights the potential of using dynamic compositionality of neural networks to tackle the challenge of scaling video understanding to the large space of complex activities. Future work includes exploring richer modules that can effectively trade-off between handling diverse structure and stronger reasoning about common patterns of activity.

Acknowledgment. We would like to thank members in the Stanford Vision Lab as well as Lu Jiang, Mei Han and Jia Li from Google, for helpful discussions and support. We would also like to thank the anonymous reviewers, whose suggestions and comments have helped to improve the paper.

References

1. Abu-El-Haija, S., et al.: YouTube-8M: a large-scale video classification benchmark. arXiv preprint arXiv:1609.08675 (2016)
2. Andreas, J., Rohrbach, M., Darrell, T., Klein, D.: Learning to compose neural networks for question answering. arXiv preprint arXiv:1601.01705 (2016)
3. Andreas, J., Rohrbach, M., Darrell, T., Klein, D.: Neural module networks. In: CVPR (2016)
4. Antol, S., et al.: VQA: visual question answering. In: ICCV (2015)
5. Thomee, B., et al.: YFCC100M: the new data in multimedia research. Commun. ACM **59**(2), 64–73 (2016)
6. Caba Heilbron, F., Escorcia, V., Ghanem, B., Carlos Niebles, J.: ActivityNet: a large-scale video benchmark for human activity understanding. In: CVPR (2015)
7. Chen, D.L., Dolan, W.B.: Collecting highly parallel data for paraphrase evaluation. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, HLT 2011, vol. 1, pp. 190–200. Association for Computational Linguistics, Stroudsburg (2011). http://dl.acm.org/citation.cfm?id=2002472.2002497
8. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: ICLR (2015)
9. Donahue, J., et al.: Long-term recurrent convolutional networks for visual recognition and description. In: CVPR (2015)
10. Feng, X., Perona, P.: Human action recognition by sequence of movelet codewords. In: Proceedings of First International Symposium on 3D Data Processing Visualization and Transmission, pp. 717–721 (2002). https://doi.org/10.1109/TDPVT.2002.1024148
11. Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Mikolov, T., et al.: Devise: a deep visual-semantic embedding model. In: NIPS (2013)
12. Gaidon, A., Harchaoui, Z., Schmid, C.: Temporal localization of actions with actoms. IEEE TPAMI **35**(11), 2782–2795 (2013)
13. Gu, C., et al.: AVA: a video dataset of spatio-temporally localized atomic visual actions. CoRR abs/1705.08421 (2017). arXiv:1705.08421
14. Guadarrama, S., et al.: YouTube2Text: recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In: ICCV (2013)
15. Han, W., et al.: Seq-NMS for video object detection. arXiv preprint arXiv:1602.08465 (2016)
16. Hendricks, L.A., Wang, O., Shechtman, E., Sivic, J., Darrell, T., Russell, B.C.: Localizing moments in video with natural language. In: ICCV (2017)
17. Hu, R., Andreas, J., Rohrbach, M., Darrell, T., Saenko, K.: Learning to reason: end-to-end module networks for visual question answering. In: ICCV (2017)
18. Hu, R., Rohrbach, M., Andreas, J., Darrell, T., Saenko, K.: Modeling relationships in referential expressions with compositional modular networks. In: CVPR (2017)
19. İkizler, N., Forsyth, D.A.: Searching for complex human activities with no visual examples. IJCV **80**, 337–357 (2008)
20. Jain, M., van Gemert, J.C., Mensink, T., Snoek, C.G.: Objects2action: classifying and localizing actions without any video example. In: ICCV (2015)

21. Ji, S., Xu, W., Yang, M., Yu, K.: 3D convolutional neural networks for human action recognition. TPAMI **35**(1), 221–231 (2013)
22. Jiang, Y.G., et al.: THUMOS challenge: action recognition with a large number of classes (2014). http://crcv.ucf.edu/THUMOS14/
23. Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C.L., Girshick, R.: CLEVR: a diagnostic dataset for compositional language and elementary visual reasoning. In: CVPR, pp. 1988–1997. IEEE (2017)
24. Johnson, J., et al.: Inferring and executing programs for visual reasoning. In: ICCV (2017)
25. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: CVPR (2014)
26. Kay, W., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017)
27. Klein, D., Manning, C.D.: Accurate unlexicalized parsing. In: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics, vol. 1, pp. 423–430. Association for Computational Linguistics (2003)
28. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: ICCV (2011)
29. Li, L.J., Su, H., Fei-Fei, L., Xing, E.P.: Object bank: a high-level image representation for scene classification & semantic feature sparsification. In: NIPS (2010)
30. Li, L.-J., Su, H., Lim, Y., Fei-Fei, L.: Objects as attributes for scene classification. In: Kutulakos, K.N. (ed.) ECCV 2010. LNCS, vol. 6553, pp. 57–69. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-35749-7_5
31. Lillo, I., Niebles, J.C., Soto, A.: Sparse composition of body poses and atomic actions for human activity recognition in RGB-D videos. Image Vis. Comput. **59**, 63–75 (2017)
32. Lu, J., Yang, J., Batra, D., Parikh, D.: Hierarchical question-image co-attention for visual question answering. In: NIPS (2016)
33. Monfort, M., et al.: Moments in time dataset: one million videos for event understanding. arXiv preprint arXiv:1801.03150 (2018)
34. Niebles, J.C., Chen, C.-W., Fei-Fei, L.: Modeling temporal structure of decomposable motion segments for activity classification. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6312, pp. 392–405. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15552-9_29
35. Peng, X., Schmid, C.: Multi-region two-stream R-CNN for action detection. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 744–759. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_45
36. Pennington, J., Socher, R., Manning, C.: GloVe: global vectors for word representation. In: EMNLP (2014)
37. Poppe, R.: A survey on vision-based human action recognition. Image Vis. Comput. **28**(6), 976–990 (2010)
38. Regneri, M., Rohrbach, M., Wetzel, D., Thater, S., Schiele, B., Pinkal, M.: Grounding action descriptions in videos. Trans. Assoc. Comput. Linguist. (2013)
39. Rohrbach, A., Torabi, A., Rohrbach, M., Tandon, N., Pal, C., Larochelle, H., Courville, A., Schiele, B.: Movie description. IJCV **123**(1), 94–120 (2017)
40. Rohrbach, M., et al.: Recognizing fine-grained and composite activities using hand-centric features and script data. IJCV **119**, 346–373 (2016)
41. Sadanand, S., Corso, J.J.: Action bank: a high-level representation of activity in video. In: CVPR (2012)
42. Schiele, B.: A database for fine grained activity detection of cooking activities. In: CVPR (2012)

43. Scovanner, P., Ali, S., Shah, M.: A 3-dimensional sift descriptor and its application to action recognition. In: Proceedings of the 15th ACM International Conference on Multimedia, MM 2007 (2007)
44. Sigurdsson, G.A., Varol, G., Wang, X., Farhadi, A., Laptev, I., Gupta, A.: Hollywood in homes: crowdsourcing data collection for activity understanding. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 510–526. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_31
45. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: NIPS (2014)
46. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. ICLR (2015)
47. Sivic, J., Zisserman, A.: Video Google: a text retrieval approach to object matching in videos. In: ICCV (2003)
48. Socher, R., Karpathy, A., Le, Q.V., Manning, C.D., Ng, A.Y.: Grounded compositional semantics for finding and describing images with sentences. Trans. Assoc. Comput. Linguist. (2014)
49. Socher, R., Lin, C.C., Manning, C., Ng, A.Y.: Parsing natural scenes and natural language with recursive neural networks. In: ICML (2011)
50. Socher, R., et al.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (2013)
51. Soomro, K., Zamir, A.R., Shah, M.: UCF101: a dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
52. Torabi, A., Tandon, N., Sigal, L.: Learning language-visual embedding for movie understanding with natural-language. arXiv preprint arXiv:1609.08124 (2016)
53. Tran, A., Cheong, L.F.: Two-stream flow-guided convolutional attention networks for action recognition. arXiv preprint arXiv:1708.09268 (2017)
54. Tran, D., Bourdev, L.D., Fergus, R., Torresani, L., Paluri, M.: C3D: generic features for video analysis. CoRR abs/1412.0767 (2014). arXiv:1412.0767
55. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: CVPR (2011)
56. Wang, L., et al.: Temporal segment networks: towards good practices for deep action recognition. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 20–36. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_2
57. Weinland, D., Ronfard, R., Boyer, E.: A survey of vision-based methods for action representation, segmentation and recognition. Comput. Vis. Image Underst. **115**, 224–241 (2011)
58. Xiao, F., Sigal, L., Lee, Y.J.: Weakly-supervised visual grounding of phrases with linguistic structures. In: CVPR (2017)
59. Xu, J., Mei, T., Yao, T., Rui, Y.: MSR-VTT: a large video description dataset for bridging video and language. In: CVPR (2016)
60. Yang, Z., He, X., Gao, J., Deng, L., Smola, A.: Stacked attention networks for image question answering. In: CVPR (2016)
61. Yao, L., et al.: Describing videos by exploiting temporal structure. In: ICCV (2015)
62. Yeung, S., Fathi, A., Fei-Fei, L.: VideoSET: video summary evaluation through text. arXiv preprint arXiv:1406.5824 (2014)
63. Yu, H., Wang, J., Huang, Z., Yang, Y., Xu, W.: Video paragraph captioning using hierarchical recurrent neural networks. In: CVPR (2016)
64. Zacks, J.M., Tversky, B.: Event structure in perception and conception. Psychol. Bull. **127**, 3–21 (2001)

65. Zellers, R., Choi, Y.: Zero-shot activity recognition with verb attribute induction. arXiv preprint arXiv:1707.09468 (2017)
66. Zhao, H., Yan, Z., Wang, H., Torresani, L., Torralba, A.: SLAC: a sparsely labeled dataset for action classification and localization. arXiv preprint arXiv:1712.09374 (2017)
67. Zhou, L., Xu, C., Corso, J.J.: ProcNets: learning to segment procedures in untrimmed and unconstrained videos. CoRR abs/1703.09788 (2017)