Task-Driven Dynamic Fusion: Reducing Ambiguity in Video Description

Xishan Zhang[1,2]*, Ke Gao[1], Yongdong Zhang[1,2], Dongming Zhang[1], Jintao Li[1],and Qi Tian[3]†
[1] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[2] University of Chinese Academy of Sciences, Beijing, China
[3] Department of Computer Science, University of Texas at San Antonio
{zhangxishan,kegao,zhyd,dmzhang,jtli}@ict.ac.cn,qitian@cs.utsa.edu

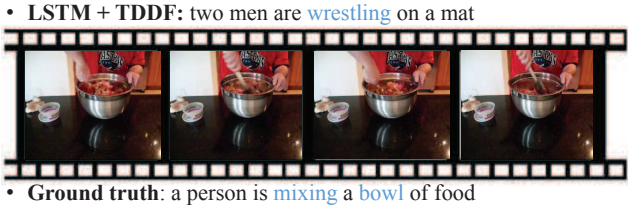
Abstract

Integrating complementary features from multiple channels is expected to solve the description ambiguity problem in video captioning, whereas inappropriate fusion strategies often harm rather than help the performance. Existing static fusion methods in video captioning such as concatenation and summation cannot attend to appropriate feature channels, thus fail to adaptively support the recognition of various kinds of visual entities such as actions and objects. This paper contributes to: 1)The first in-depth study of the weakness inherent in data-driven static fusion methods for video captioning. 2) The establishment of a task-driven dynamic fusion (TDDF) method. It can adaptively choose different fusion patterns according to model status. 3) The improvement of video captioning. Extensive experiments conducted on two well-known benchmarks demonstrate that our dynamic fusion method outperforms the state-of-the-art results on MSVD with METEOR scores 0.333, and achieves superior METEOR scores 0.278 on MSR-VTT-10K. Compared to single features, the relative improvement derived from our fusion method are 10.0% and 5.7% respectively on two datasets.

1. Introduction

Automatically generating natural and accurate language descriptions for videos is one of the ultimate goal of video understanding. Although early work in video captioning borrows insight from image captioning [29], the task is much more challenging due to various objects and complex human actions.

- **Ground truth:** guys on mat wrestling
- **LSTM:** a group of people are dancing on a track
- **LSTM + TDDF:** two men are wrestling on a mat



- **Ground truth**: a person is mixing a bowl of food
- **LSTM:** a person is cooking
- **LSTM + TDDF:** a person is mixing salad in a bowl

Figure 1. Examples of video description generation. **Top**: LSTM misunderstands the action 'wrestling' as 'dancing'. Our LSTM +TDDF correctly recognizes the action 'wrestling'. **Bottom**: LSTM generates vague description 'cooking'. Our LSTM +TDDF generates informative 'mixing', 'salad' and 'bowl'.

Despite large progress in video captioning, existing methods often suffer from description ambiguity, including recognition error and detail deficiency. Taking the LSTM results in Figure. 1 for example, the verb 'wrestling' is incorrectly recognized as 'dancing' in the description of the first video, while details such as 'mixing' and 'bowl' are missing for the second one.

It is well known that different visual cues make different contributions to the recognition of various video content. Integrating complementary features from multiple channels is expected to solve the description ambiguity problem [11, 9, 8, 21, 34, 37, 3, 40, 22, 26, 39]. While different fusion methods such as concatenation and summation have been used in video captioning, the relative increase obtained by fusing multiple-channel visual features is only 0.1%–1.7% [11] or even -0.7% [26]. It reveals that existing visual fusion strategies in video captioning have not made full use of each channel of features and their correlation.

*This work was supported by the National Natural Science Foundation of China (No.61525206, No.61672495, No.61271428, No.61429201), the National Key Research and Development Plan of China under Grant 2016YFB0801203,2016YFB0801200 and Beijing Advanced Innovation Center for Imaging Technology under Grant BAICIT-2016009

†This work was supported in part by ARO grant W911NF-15-1-0290 and Faculty Research Gift Awards by NEC Laboratories of America and Blippar.

We observe that most visual entities in video descriptions could be divided into three categories: 1) appearance-centric 2) motion-centric and 3) correlation-centric. As shown in Figure. 1, in the sentence 'a person is mixing salad in a bowl', the visual entities 'person' and 'mixing' can be easily recognized through appearance and motion features respectively. As to details such as 'salad' which is hard to infer due to clutter, the correlative constraints between motion and appearance make it possible to deduce. Theoretically, feature concatenation [23] is capable of modeling various correlation across features, but in practice the improvement is often limited in video captioning. It mainly because the unbalance distribution of object-related entities and action-related entities in video description. For example, there are 36% nouns and 19% verbs in the training descriptions of MSR-VTT-10K dataset [11], which is also the case in most video captioning datasets. Therefore, in the data-driven fusion method such as feature concatenation, the appearance features are often enhanced, while motion features are suppressed. Such static fusion models cannot adaptively support the recognition of three different kinds of visual entities, which result in description ambiguity, including recognition error and detail deficiency.

To alleviate description ambiguity, we propose a task-driven dynamic fusion approach which can adaptively attend to certain visual cues based on the current model status, so that the generated visual representation will be most relevant to the current word. The fusion model consists of three different fusion patterns, which support the recognition of three kinds of visual entities separately. The proposed fusion approach consists of two steps. 1) Temporal Attention. For different feature channels, we selectively focus on relevant temporal points according to current model status. 2) Dynamic Fusion. Three different fusion patterns are designed to support the recognition of appearance-centric, motion-centric and correlation-centric entities. The fusion model learns to dynamically choose one of the three fusion patterns appropriately according to task status.

In summary, we make the following contributions:

- In-depth study of the weakness inherent in data-driven static fusion methods for video captioning. Existing static fusion methods cannot adaptively support the recognition of various kinds of visual entities, which results in description ambiguity, including recognition error and detail deficiency.

- A task-driven dynamic fusion (TDDF) model is proposed to adaptively choose different fusion patterns according to task status. The dynamic fusion model can attend to certain visual cues that are most relevant to the current word. Through learning correlativity constraints between multiple visual channels, the recognition of all the appearance-centric, motion-centric and correlation-centric entities can be promoted, thus reducing ambiguity in video description.

- Extensive experiments conducted on two well-known video captioning benchmarks, MSVD and MSR-VTT-10K demonstrate that our dynamic fusion method achieves noticeable gains by appropriately integrating multiple-channel features. Compared to single features, the relative improvement derived from our fusion method are 10.0% and 5.7% respectively on two datasets.

2. Related work

Video/Image Captioning: The work on video/image captioning can be divided into two categories: the bottom-up approaches [9, 13, 6] and top-down approaches [18, 37, 35]. The bottom-up approaches first recognize the visual concepts and form them into a description through sentence templates. Appropriate features are used to detect those concepts separately: motion features for actions [28, 16, 7]; different kinds of appearance features for objects, attributes and scenes [32]. Therefore, the correlation between the bottom features and the co-occurrence of the top visual concepts are not fully explored. The top-down approaches are the state-of-the-art ones, which formulate the task into a whole encoder-decoder framework of machine translation. The recognition of visual concepts is implicitly achieved during the sentence generation. Relatively few work in video captioning focuses on the generation of a good task-specific visual representation, except for the recent work of Pan [18]. Pan [18] mainly aimed at capturing temporal information in video representation. Our proposed in-depth study of the fusion of motion and appearance information in video captioning generates a joint representation by promoting individual feature channels and correlating complimentary features according to task status.

Feature Fusion: All the existing feature fusion methods in video caption are static fusion, which means the visual fusion model is not affected by the previous generated target words. The work includes score-level decision fusion [31, 28] and early-stage feature combination [11, 19, 3, 22, 26]. Decision fusion is achieved by averaging a set of network predictors. However, the decision fusion is not data-driven, since discrepant prediction capabilities on different samples of the individual features are neglected. Feature combination is data-driven, which combines motion and appearance features through concatenation, summation or maximization. However, the performance improvement by feature combination is either limited [11](relatively 0.1%–1.7% improvement) or even worse than the single features as reported in many work [3, 22, 26]. Our proposed dynamic fusion model can adaptively choose different fusion patterns according to task status.

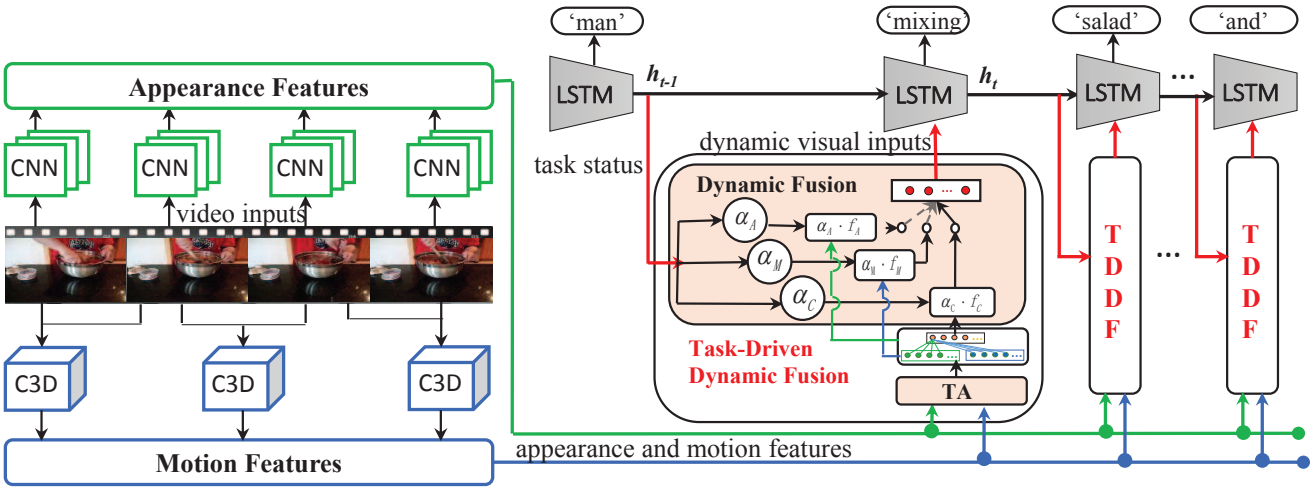


Figure 2. Illustration of task-driven dynamic fusion (TDDF) in video captioning. The blue lines are motion features, the green lines are appearance features, and the red lines are information flows between LSTM and TDDF. Motion features, appearance features and model status information are inputs to TDDF unit. As long as the output word is not EOS (*i.e.* end of sentence), the the encoder part TDDF unit will generates a dynamic visual inputs to each iteration of the LSTM decoder. The details of TDDF are shown in Figure.3.

Attention: Visual attention [36, 35, 37] is widely applied in captioning task to selectively focus on a subset of temporal frames of video or spatial regions of images. The brilliant idea behind attention is to consider task status into the feature encoding part. In attention mechanism, the target word is generated based on the most relevant frames and regions. The relevance is measured by the previous generated existing words, which represent the task status. Our work is closely related to attention mechanism in the sense of task-driven dynamic concentration on the visual feature part. However, there is a significant difference between our dynamic fusion and the wildly used attention mechanism.The attention mechanism deals with homogeneous features extracted from different samples (frames or regions). Our dynamic fusion deals with heterogeneous features even from the same sample. Therefore, in attention, the content of visual features will determine the relevance of it to the sentence context. In dynamic fusion, not the feature content but the kind of the feature will determines its relevance. As a result, the attention mechanism attends to certain visual concepts like the region of a dog or a short clips of running dog. Our dynamic fusion is built upon attention and extends it one step further, which automatically determines whether the appearance of the dog, or the movement of the dog, or the combination should be focused.

3. Video Description with Task-Driven Dynamic Fusion

3.1. Overall Framework

We build our video captioning framework upon the popular ConvNet + LSTM architecture [30, 20, 36, 19], which consists of the encoder part and the decoder part as shown in Figure. 2. The encoder part aims at learning a good visual representation and the decoder part preforms language generation. The video input is represented as a temporal sequence $\mathbf{V} = \{\mathbf{v_1}, ..., \mathbf{v_n}\}$, consisting of motion and appearance features $\mathbf{v_i} = [\mathbf{vm_i}, \mathbf{vs_i}]$ extracted from video frames and clips. The output is the words sequence $\mathbf{Y} = \{y_1, y_2.., y_m\}$ describing the video. In the baseline model [30], the visual features are inputs to the zeroth round of LSTM iteration. However, it is unrealistic to cram the visual information of the whole video into a single vector. Therefore, we follow the implementation of [35, 36] to introduce visual features at each time of the word generation. This requires adding a new visual input path $\varphi_\mathbf{t}(\mathbf{V})$ to the LSTM cell formulated as follows:

$$\mathbf{i_t} = \sigma(W_i\mathbf{E}[y_{t-1}] + U_i\mathbf{h_{t-1}} + A_i\varphi_\mathbf{t}(\mathbf{V}) + \mathbf{b_i}) \quad (1)$$

$$\mathbf{f_t} = \sigma(W_f\mathbf{E}[y_{t-1}] + U_f\mathbf{h_{t-1}} + A_f\varphi_\mathbf{t}(\mathbf{V}) + \mathbf{b_f}) \quad (2)$$

$$\mathbf{o_t} = \sigma(W_o\mathbf{E}[y_{t-1}] + U_o\mathbf{h_{t-1}} + A_o\varphi_\mathbf{t}(\mathbf{V}) + \mathbf{b_o}) \quad (3)$$

$$\mathbf{g_t} = \phi(W_g\mathbf{E}[y_{t-1}] + U_g\mathbf{h_{t-1}} + A_g\varphi_\mathbf{t}(\mathbf{V}) + \mathbf{b_g}) \quad (4)$$

$$\mathbf{c_t} = \mathbf{c_{t-1}} \odot \mathbf{f_t} + \mathbf{i_t} \odot \mathbf{g_t} \quad (5)$$

$$\mathbf{h_t} = \mathbf{o_t} \odot \phi(\mathbf{c_t}) \quad (6)$$

where σ is sigmoid function, ϕ is tanh function, y_{t-1} is the previous word, h_{t-1} is the previous hidden state, and $\varphi_t(V)$ is the dynamically fused visual feature vectors, which enable us to dynamically adjust visual input according to the task status. This will be explained in details in the following sections. The probability distributions over the set of

possible words are obtained through a single hidden layer:

$$\hat{\mathbf{y}}_\mathbf{t} = softmax(U_y\phi(W_y[\mathbf{h_t}, \varphi_t(V), \mathbf{E}[y_{t-1}]]) + \mathbf{d_y}) \quad (7)$$

where $[\mathbf{h_t}, \varphi_t(V), \mathbf{E}[y_{t-1}]]$ denotes the concatenation of the three vectors.

We further enhance the encoder part by adding the task-driven dynamic fusion layer as shown in Figure. 2. First, we extract and select a variable-length motion and appearance features, and generate two channels of video representation through a temporal attention mechanism as explained in Section 3.2. Then, we dynamically combine different feature channels according to the sentence context, as explained in Section 3.3.

3.2. Temporal Attention

In this section, we encode the variable-length video input $\mathbf{V} = \{\mathbf{v_1}, ..., \mathbf{v_n}\}$ into a sentence-length temporal representation $\psi(\mathbf{V}) = \{\psi_1(\mathbf{V}), \psi_2(\mathbf{V}), ..., \psi_m(\mathbf{V})\}$. Each of the $\psi_t(\mathbf{V})$ is the weighted sum of all the n visual features through an attention mechanism. The input video features $\mathbf{v_i}$ contain both appearance feature and motion feature $\mathbf{v_i} = [\mathbf{vm_i}, \mathbf{vs_i}]$.

The traditional dynamic attention strategy [35] does not differentiate the appearance and motion features, where $\psi_t(\mathbf{V}) = \sum_i^n a_i^{(t)}\mathbf{v_i}$, and $a_i^{(t)}$ reflects the relevance of the i-th visual feature to the t-th word y_t. Intuitively, the static appearance features and motion features have different relevance to verbs and nouns, so we assign soft attention to different features separately. In particular, the attention process is applied to static appearance features $\mathbf{VS}^{(t)} = \sum_i^n as_i^{(t)}\mathbf{vs_i}$ and motion features $\mathbf{VM}^{(t)} = \sum_i^n am_i^{(t)}\mathbf{vm_i}$. This sophisticated version of attention allows feature channels have different temporal length n if necessary. Finally, the visual input for the inference of the t-th word y_t is the combination of appearance and motion, $\psi_t(\mathbf{V}) = [\mathbf{VM}^{(t)}, \mathbf{VS}^{(t)}]$.

The attention function is used to calculate $as_i^{(t)}$ and $am_i^{(t)}$, which takes the previous hidden state $\mathbf{h}_{t-1}$ of the LSTM decoder and the i-th temporal features as inputs.

$$as_i^{(t)} = f(\mathbf{h_{t-1}}, \mathbf{vs}_i), \quad (8)$$

$$am_i^{(t)} = f(\mathbf{h_{t-1}}, \mathbf{vm}_i) \quad (9)$$

The attention function f is implemented by a multi-layer perceptron (MLP) as in [33],which has an universal approximation property.

$$f(\mathbf{h_{t-1}}, \mathbf{v}_i) = W_a\phi(W_h\mathbf{h_{t-1}} + W_v\mathbf{v}_i) \quad (10)$$

where W_a,W_h W_v are parameters to be estimated, and W_h are shared by motion and appearance features, while W_v is feature dependent. Once the attention score for all the temporal segments are computed, we normalize them through softmax function $as_i^{(t)} = exp\{as_i^{(t)}\}/\sum_{i=1}^n exp\{as_i^{(t)}\}$.

3.3. Dynamic Fusion

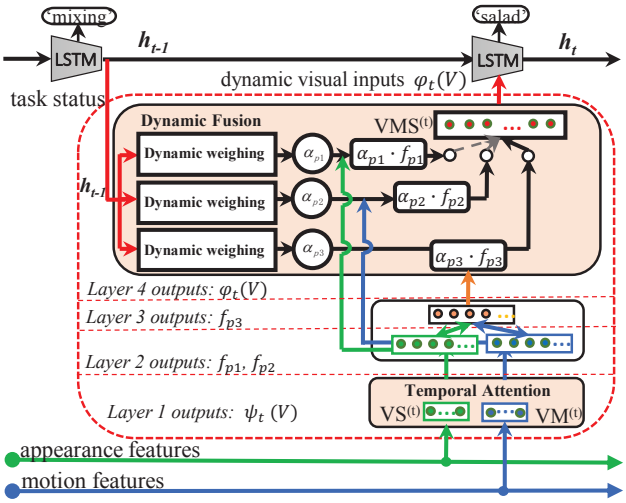


Figure 3. Illustration of the task-driven dynamic fusion unit. Motion features, appearance features and model status information are inputs to TDDF unit. TDDF unit generates a dynamic visual inputs to each iteration of LSTM decoder. There are three pathway: p1 is appearance pathway, p2 is motion pathway and p3 is correlation pathway. Layer 1 performs temporal attention on motion and appearance features separately. Layer 2 performs feature learning. Layer 3 performs concatenation fusion. Layer 4 performs dynamic fusion, choosing appropriate pathway relevant to the current word.

We further dynamically combine motion and appearance features $\mathbf{VM}^{(t)}, \mathbf{VS}^{(t)}$ to get the fused representation $\varphi_\mathbf{t}(\mathbf{V}) = \mathbf{VMS}^{(t)}$. We first introduce two kinds of basic shallow fusion functions: concatenation fusion and sum or max fusion. Then we illustrate the proposed dynamic fusion.

1) Concatenation fusion. The fusion function is

$$\begin{aligned}\mathbf{VMS}^{(t)} &= \mathbf{W_F}([\mathbf{VM}^{(t)}, \mathbf{VS}^{(t)}]^T) \\ &= \mathbf{W_{F_l}}\mathbf{VM}^{(t)} + \mathbf{W_{F_r}}\mathbf{VS}^{(t)}\end{aligned} \quad (11)$$

where motion features and appearance features are concatenated together $[\mathbf{VM}^{(t)}, \mathbf{VS}^{(t)}]^T \in \mathbb{R}^{D_m+D_s}$. The convolves with parameters $\mathbf{W_F} \in \mathbb{R}^{M\times(D_m+D_s)}$ reduce the fused output dimension to M. The concatenation fusion is widely applied in multi-modal learning [17] and recent inception module in GoogLeNet [25, 24]. The concatenation fusion is capable of modeling correlations within and across features. However, the fusion parameters are fixed once learned.

2) Sum or max fusion. The fusion function is the element-wise sum $\mathbf{VMS}^{(t)} = \mathbf{VM}^{(t)} \cdot + \mathbf{VS}^{(t)}$ or the element-wise max $\mathbf{VMS}^{(t)} = \max\{\mathbf{VM}^{(t)}, \mathbf{VS}^{(t)}\}$. These parameter-free fusion functions usually applied to features of same kind so the element-wise addition or max

is reasonable. Sum fusion is applied to the shortcut connection in Residual Network [10] and the combination layers in FractalNet [14]. However, different from concatenation fusion, sum or max fusion can hardly model the correlation between different dimensions of heterogeneous features.

3) Dynamic fusion. We propose a fusion function that is the element-wise weighted-sum of feature channels $\mathbf{VMS}^{(t)} = a_{p_1}^{(t)}\mathbf{VM}^{(t)} \cdot + a_{p_2}^{(t)}\mathbf{VS}^{(t)}$. Therefore, the sum or max fusion can be transformed to a special case of the dynamic weighted-sum fusion. Different from the projected shortcuts in Residual Network [10] where the weights are fixed parameters, $a_{p_1}^{(t)}$ and $a_{p_2}^{(t)}$ are dynamically determined by the task status $\mathbf{h_{t-1}}$. The idea of dynamic fusion is similar to the idea of attention mechanism [33, 36, 35] in the sense that both of them deal with how well the inputs are related to the target words. However, attention mechanism deals with homogeneous features extracted from different samples v_i with $a_{v_i}^{(t)} = f(\mathbf{h_{t-1}}, v_i)$, where the feature content v_i will determine the attention weights. Dynamic fusion deals with heterogeneous feature channel p_i with $a_{p_i} = f(\mathbf{h_{t-1}}, p_i) := f_{p_i}(\mathbf{h_{t-1}})$. The fusion weights are determined by the type of the feature p_i instead of the content of the feature. Similar to the sum fusion, the element-wise weighted-sum hardly models the correlation between multiple features, and it is not reasonable to do element-wise addition for heterogeneous features.

In design of task-driven dynamic fusion (TDDF) unit, we take advantage of the above three kinds of fusion functions, which are related and complimentary. As shown in Figure. 3, the inputs of TDDF unit are motion features $\mathbf{VM}^{(t)}$, appearance features $\mathbf{VS}^{(t)}$, and model status information $\mathbf{h_{t-1}}$. The outputs of TDDF unit are dynamic visual inputs to each iteration of the LSTM decoder. In the beginning, motion features and appearance features separately go through a fully connected feature learning Layer 2 to generate motion pathway and appearance pathway. Through Layer 2, we reduce the original feature dimension to obtain better representations, which are effective to fuse and reasonable to perform the following element-wise addition. Then, the followed concatenation fusion Layer 3 is used to combine the refined motion and appearance features, and generates the correlation pathway. At last, we apply a dynamic fusion Layer 4 on the top of motion, appearance and correlation pathways. The three pathways correspond to three different fusion patterns that are designed to support the recognition of appearance-centric, motion-centric and correlation-centric entities in video description. Dynamic fusion Layer 4 learns to adaptively choose one of the three fusion patterns according to task status through a dynamic weighing mechanism. In particular, the dynamic weights $\mathbf{a}^{(t)}$ for all three pathways are obtained through:

$$\mathbf{s}^{(t)} = softmax(\mathbf{W}_s\mathbf{h_{t-1}} + \mathbf{b}_s), \tag{12}$$

$$\mathbf{c}^{(t)} = \sigma(\mathbf{W}_c\mathbf{h_{t-1}} + \mathbf{b}_c), \tag{13}$$

$$\mathbf{a}^{(t)} = {\mathbf{s}^{(t)}}^T\mathbf{c}^{(t)}, \tag{14}$$

where $\mathbf{s}^{(t)} \in \mathbb{R}^3$ determines the most relevant pathway among all three, and $\mathbf{c}^{(t)} \in \mathbb{R}^3$ determines whether each channel is relevant to the description context $\mathbf{h_{t-1}}$. For the specific feature channel p_i, the weights is:

$$a_{p_i}^{(t)} = f_{p_i}(\mathbf{h_{t-1}}) = s_{p_i}^{(t)} \cdot c_{p_i}^{(t)}, \tag{15}$$

In training, the feature is chosen by minimizing $r(\mathbf{s})$:

$$r(\mathbf{s}) = -\sum_{t}^{m}(\sum_{p_i}(s_{p_i}^{(t)} log(s_{p_i}^{(t)}))), \tag{16}$$

The optimum $\mathbf{s}$ should be one of $[1,0,0]$,$[0,1,0]$ and $[0,0,1]$, which determines the most relevant feature pathway. The result of the sparse $\mathbf{s}^{(t)}$ makes the shortcut between the Layer 2 and Layer 4, which enables to automatically skip feature pathways to promote individual feature channel.

4. Experiments

4.1. Dataset and Evaluation Metrics

Dataset: We conduct the experiments on two video captioning benchmarks: MSVD [1] and MSR-VTT-10K [11]. MSVD [1] consists of 1,970 video clips. Almost all the existing video captioning methods have been tested on this dataset. We adopt the widely-used train and test splits provided by [29, 35], with a training set of 1,200 video clips, a validation set of 100 clips and a test set consisting of the remaining clips. MSR-VTT-10K [11] consists of 10,000 video clips, which is the most challenging dataset for video captioning to date. We used the official split [1] with 6,513 videos for training, 497 for validation and 2,990 for testing. We report the results on both the validation and test splits on MSR-VTT-10K.

Evaluation Metrics: Several standard metrics such as BLEU(precision-based), METEOR(harmonic mean of precision and recall), CIDEr(consensus-based), and ROUGE-L (recall-based) are used for evaluating the video captioning [2]. We utilize the Microsoft COCO evaluation server [2] and report all the four metrics. Among the four metrics, METEOR and CIDer are now regarded as the better ones [27, 18, 28, 37].

4.2. Implementation Details

Features: For appearance features, we adopt 4,096-dimensional fc6 layer from VGG-19 and 1,024-dimensional pool5 layer from GoogLeNet-bu4k [15], a variant of

[1] http://ms-multimedia-challenge.com/

Table 1. Performance evaluation on MSVD

	METEOR	(%↑)	CIDEr	(%↑)	ROUGE-L	(%↑)	BLEU4	(%↑)
VGG	0.302	-	0.563	-	0.675	-	0.416	-
C3D	0.303	-	0.542	-	0.667	-	0.412	-
CON(VGG+C3D)	0.317	(4.6%↑)	0.652	(15.8%↑)	0.680	(0.7%↑)	0.428	(2.9%↑)
MAX-2(VGG+C3D)	0.308	(1.7%↑)	0.558	(-0.9%↑)	0.675	(0%↑)	0.417	(0.2%↑)
SUM-2(VGG+C3D)	0.307	(1.3%↑)	0.654	(16.1%↑)	0.681	(0.9%↑)	0.438	(5.3%↑)
MAX-3(VGG+C3D)	0.313	(3.3%↑)	0.663	(17.7%↑)	0.687	(1.8%↑)	0.452	(8.6%↑)
SUM-3(VGG+C3D)	0.314	(3.6%↑)	0.602	(6.9%↑)	0.684	(1.3%↑)	0.440	(5.8%↑)
TA [35]	0.296	-	0.517	-	-	-	0.419	-
LSTM-E [19]	0.310	(3.7%↑)	-	-	-	-	0.453	(8.6%↑)
h-RNN [37]	0.326	(4.8%↑)	0.658	(6.0%↑)	-	-	**0.499**	(2.2%↑)
HRNE [18]	0.331	-	-	-	-	-	0.438	-
TDDF(VGG+C3D)	**0.333**	**(10.0%↑)**	**0.730**	**(29.7%↑)**	**0.697**	**(3.3%↑)**	0.458	**(10.1%↑)**

*fusion method has relatively (%↑) improvement over the best single features

GoogLeNet [25]. For motion features, we adopt the 4,096-dimensional fc6 layer from C3D [5] pre-trained on Sports-1M video dataset [12]. We take continuous 16 frames as the input short clips for the C3D, similar as [5, 11]. At last, we select 28 equally-spaced frame appearance features and clip motion features as the visual inputs, similar as [35].

Model and Training: The overview of our video captioning architecture is shown in Figure 2. The size of hidden layer in LSTM is 1024. As MSR-VTT-10K is much larger than MSVD, we use two-layer LSTM on MSR-VTT-10K and one-layer LSTM on MSVD. The proposed task-driven dynamic fusion unit is shown in Figure 3, Layer 2 and Layer 3 are fully connected layers with tanh function as activation. The dimensionality of Layer 2 is 1024 for each input feature channel, the dimensionalities of Layer 3 and Layer 4 are 1024 respectively. As for the parameters in temporal feature selection $W_h \in \mathbb{R}^{1024\times1024}$ and $\mathbf{v_a} \in \mathbb{R}^{1024}$. In training, we use the Adadelta algorithm [38] with gradients computed by back propagation algorithm. The model is trained end-to-end by minimizing negative log-likelihood:

$$L = -log(P(\mathbf{Y}|\mathbf{V})) + g(\mathbf{as}) + g(\mathbf{am}) + r(\mathbf{s}) \quad (17)$$

While predicting the word, we regularize the attention weights to enforce the completeness of attention paid to every temporal feature when generating the complete sentence. The regularization function is similar to [35, 33]:

$$g(\mathbf{as}) = -\sum_{t}^{m}(1 - \sum_{j}^{t} as_{ij})^2 \quad (18)$$

4.3. Experimental Results

Baseline Methods: First, we compare our task-driven dynamic visual fusion method (TDDF) with single feature methods, denoted as VGG, GoogLeNet, and C3D. Then, as stated in Section 3.3, our TDDF unit takes advantage of static fusion, so we compare to these methods: concatenation fusion denoted as CON, sum fusion denoted as SUM and max fusion denoted as MAX. CON simply concatenates the features on Layer 2 and feeds it to Layer 4. SUM-2 or MAX-2 adds or maximizes the results on Layer 2. SUM-3 or MAX-3 adds or maximizes the results both on Layer 2 and Layer 3 to form the fused representation.

State-of-the-art Methods: On MSVD, we compare to three methods: TA [35], LSTM-E [19], h-RNN [37] and HRNE [18]. TA is the first work applying temporal attention in video captioning. LSTM-E simultaneously explores the learning of LSTM and visual-semantic embedding. h-RNN explores both temporal and spatial attention in video captioning. HRNE aimes at learning a task-specified video representation for video captioning. h-RNN and HRNE report the best results on MSVD at present. On MSR-VTT-10K, there are relatively less work. We compare to three methods:SA-LSTM [11], C3D+Res [26], v2t_nagvigator [4]. SA-LSTM is the baseline method published along with the MSR-VTT-10K dataset, but it is done on a different split from ours. It uses a two-layer LSTM with temporal attention mechanism. **C3D+Res** [26] investigates multimodal fusion. Though the whole framework incorporates audio modality, we compare with their visual fusion results. v2t_nagvigator [4] is the best result on the leader board [2]. As some work also fuses multiple features and reports the results before and after fusion, we present their relative improvement by fusion methods.

Results on MSVD: We report the results on MSVD in Table. 1. Our task-driven dynamic visual fusion method achieves the best METEOR and CIDEr scores among all the methods. We also report the relative improvement obtained

[2]http://ms-multimedia-challenge.com/leaderboard

Table 2. Performance evaluation on MSR-VTT-10K

	Test split				Valid split			
	BLEU4	METEOR	CIDEr	ROUGE-L	BLEU4	METEOR	CIDEr	ROUGE-L
VGG	0.338	0.263	0.384	0.569	0.330	0.265	0.365	0.564
C3D	0.363	0.263	0.397	0.575	0.340	0.264	0.377	0.569
GoogLeNet	0.328	0.268	0.398	0.559	0.317	0.267	0.389	0.555
CON(GoogLeNet+C3D)	0.368	0.267	0.406	0.583	0.364	0.273	0.392	0.581
SUM-2(GoogLeNet+C3D)	0.340	0.258	0.382	0.570	0.332	0.260	0.371	0.564
MAX-2(GoogLeNet+C3D)	0.353	0.261	0.374	0.584	0.361	0.267	0.381	0.584
v2t_nagvigator [4]	0.408	0.282	0.448	0.609	0.394	0.275	0.480	0.600
C3D+Res [26]	-	-	-	-	0.385	0.267	0.411	0.601
(relative improvement%↑)	-	-	-	-	(-0.1%↑)	(-0.7%↑)	(2.8%↑)	(-0.6%↑)
SA-LSTM(VGG+C3D) [11]*	0.405	0.299	-	-	-	-	-	-
(relative improvement%↑)	(0.9%↑)	(1.7%↑)	-	-	-	-	-	-
TDDF(GoogLeNet+C3D)	0.372	0.277	0.441	0.586	0.367	0.280	0.434	0.587
(relative improvement%↑)	(2.5%↑)	(3.3%↑)	**(10.8%↑)**	(1.9%↑)	**(7.9%↑)**	(4.9%↑)	(11.5%↑)	(2.1%↑)
TDDF(VGG+C3D)	0.373	0.278	0.438	0.592	0.355	0.282	0.427	0.591
(relative improvement%↑)	**(2.7%↑)**	**(5.7%↑)**	(10.3%↑)	**(2.9%↑)**	(4.4%↑)	**(6.4%↑)**	**(13.2%↑)**	**(3.9%↑)**

*tested on different split

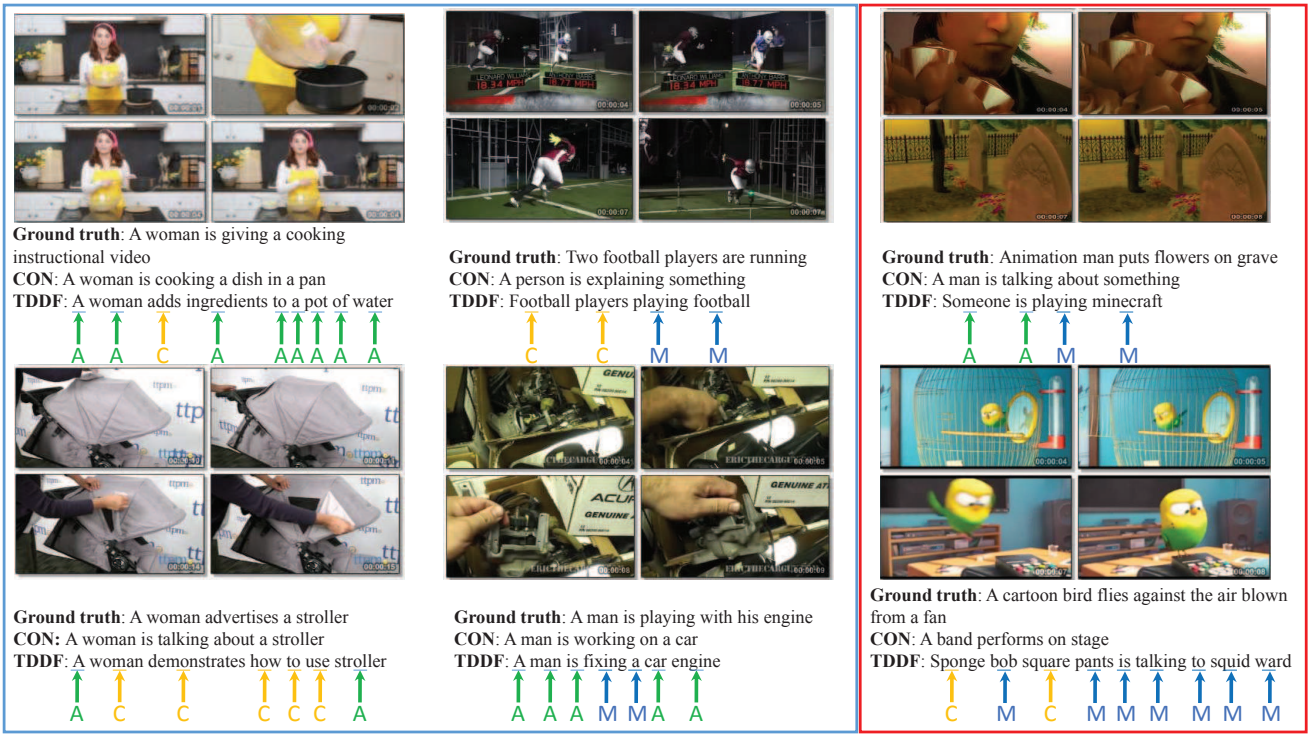


Figure 4. Examples to show a dynamic focus on different feature channels for different words in video captioning. The arrows show which features are used in determining the current words. Blue arrow represents motion features, green arrow represents appearance features and orange arrow represents the combination of motion and appearance. The red box shows the failed cases.

by all the fusion method. Compared to the methods using single features, our method obtains 10% relative improvements in terms of METEOR, and 29.7% relative improvements in terms of CIDEr. These two consensus-based metrics reward a sentence for being similar to the majority of human written descriptions. Our TDDF is capable of adapting and promoting the visual features according to the description context, therefore it generates better visual repre-

sentation suitable for different sentences that share a similar context. The baseline static fusion methods also have improvement over the single feature methods. Our method is better than MAX-2 and SUM-2, suggesting that considering the feature correlation is necessary. Although CON, MAX-3 and SUM-3 considered the feature correlation through a concatenation fusion layer, they still perform worse than our fusion method. The experiments suggest that both the feature correlation and the dynamic selection among features are crucial in our task. TA [35] applied the attention mechanism on the concatenated feature channels, and our method outperforms it by 9.3% relatively, which shows that the fusion strategy after attention mechanism is better. Compared to LSTM-E [19], h-RNN [37] and HRNE [18], our method achieves best results in METEOR and CIDEr. This result confirms the effectiveness of our TDDF. We notice that h-RNN [37] outperforms the others in terms of BLEU. While h-RNN proposed a better language model which can utilize multiple descriptions of a video at one time, our work, however, is focusing on fusing a good visual feature to improve the encoder part, not the language decoder part. Besides, METEOR and CIDEr are considered to be more reliable than BLEU [27, 18, 28, 37]. h-RNN [18] also made a comparison between single feature and fused features. Their relative improvement obtained by the fusion method is less than our fusion method.

Results on MSR-VTT-10K: We report the results on MSR-VTT-10K in Table. 2, shown that our method outperforms the single feature methods by 3.3%–5.7% relatively in terms of METEOR and 10.3% –10.8% relatively in terms of CIDEr. On this challenging dataset, the basic fusion methods CON, MAX-2 and SUM-2 hardly have any improvements over single features, which is consistent with the finding in [11, 3, 22, 26]. This is mainly because the human descriptions of the same video are more diverse on this challenging dataset. Therefore, the fusion methods in the task of video caption is worth exploring. In terms of METEOR and CIDEr, our method outperforms C3D+Res [26] method, which fused the C3D and the appearance feature from residual network [10]. In terms of the precision-based BLEU4 which tries to give more weights on human-like grammatically correct sentences, C3D+Res performs better. Considering the fact that 836 out of 23,667 words in MSR-VTT-10K sentences are misspelled (*e.g.*, 'basktball' and 'peson') [26], it is much more challenging for captioning on this dataset if the misspelling is not corrected. As to SA-LSTM(VGG+C3D) [11] and v2t_nagvigator [4], both of them has a higher performance than our method. For SA-LSTM(VGG+C3D), it is done on a different split from our available data, which makes it inappropriate to compare. For v2t_nagvigator, it utilizes extra data to train action and object detectors and applies theses detectors to pre-process the videos, meanwhile sentence re-ranking methods are also used to post-process the generated sentences. However, we aim to improve the visual encoder part, which has fundamental differences from their method.

Qualitative Analysis: In Figure. 4, we visualize the dynamic weights $\mathbf{s}^{(t)}$ for different feature channels. Though $\mathbf{s}^{(t)}$ is not the final fusion weights $\mathbf{a}^{(t)}$, it serves as a automatic switch to give us intuition in what features the model is focusing on to predict the current word. The blue arrow represents motion inputs, the green arrow represents appearance inputs, and the orange arrow represents the correlated motion and appearance inputs. As seen from Figure. 4, the green arrows (the appearance features) dominate the generation of most words. The nouns 'car' and 'engine' are learned from the appearance features and the verb 'fixing' is learned from the motion features. Moreover, the 'football players' is inferred from the correlated motion and appearance inputs. When there is no visual 'football' in the video, our model pays attention to the action cue to guess the 'football', verifying that the motion features still contribute to the recognition of nouns. We also show that our method is failed to describe some animation films. These failure cases show the limitation of our current method. The predicting of the wrong action 'playing' will pass down misleading context to predict the noun 'minecraft'. Our approach suffers from a known problem as in all the encoder-decoder based video/image captioning method. The problem is that the training process uses the right previous word to generate the next word, while in the testing process the previous word is not guaranteed. This problem is amplified in our work, where our description context information is different during training and testing. A potential cure is to improve the language model in the training, where the model can be devoid of ground truth in sentences generation.

5. Conclusion

Existing static fusion methods cannot adaptively support the recognition of various kinds of visual entities, so the relative increase obtained by fusing multiple-channel visual features is limited. In this paper, we propose a task-driven dynamic visual fusion method for video captioning, which achieves state-of-the-art performance on popular benchmarks. Our method adaptively chooses different fusion patterns according to task status. Three different fusion patterns are designed to support the recognition of three visual entities respectively, including appearance-centric, motion-centric and correlation-centric entities. The dynamic fusion model can attend to certain visual cues that are most relevant to the current word, thus reducing ambiguity in video description. Our task-driven dynamic fusion method can be added on any encoder-decoder based video captioning architecture, so any further improvement on related architectures will promote the overall performance.

References

[1] D. L. Chen and W. B. Dolan. Collecting highly parallel data for paraphrase evaluation. In *ACL*, 2011.

[2] X. Chen, H. Fang, T. Y. Lin, R. Vedantam, S. Gupta, P. Dollar, and C. L. Zitnick. Microsoft coco captions: Data collection and evaluation server. In *arXiv*, 2015.

[3] J. Dong, X. Li, W. Lan, Y. Huo, and C. G. M. Snoek. Early embedding and late reranking for video captioning. In *ACM MM*, 2016.

[4] J. Dong, X. Li, W. Lan, Y. Huo, and C. G. M. Snoek. Early embedding and late reranking for video captioning. In *ACM MM*, 2016.

[5] T. Du, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. C3d: Generic features for video analysis. In *arXiv*, 2014.

[6] H. Fang, J. C. Platt, C. L. Zitnick, G. Zweig, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollar, and J. Gao. From captions to visual concepts and back. In *CVPR*, 2015.

[7] C. Feichtenhofer, A. Pinz, and A. Zisserman. Convolutional two-stream network fusion for video action recognition. In *CVPR*, 2016.

[8] X. Gao, S. C. H. Hoi, Y. Zhang, J. Wan, and J. Li. Soml: Sparse online metric learning with application to image retrieval. *AAAI*, 2014.

[9] S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, S. Venugopalan, R. Mooney, T. Darrell, and K. Saenko. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *ICCV*, 2013.

[10] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[11] T. Y. Y. R. Jun Xu, Tao Mei. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016.

[12] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and F. F. Li. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

[13] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Babytalk: Understanding and generating simple image descriptions. In *CVPR*, 2013.

[14] G. Larsson, M. Maire, and G. Shakhnarovich. Fractalnet: Ultra-deep neural networks without residuals. In *arXiv*, 2016.

[15] P. Mettes, D. C. Koelma, and C. G. M. Snoek. The imagenet shuffle: Reorganized pre-training for video event detection. In *ICMR*, 2016.

[16] Y. H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In *CVPR*, 2015.

[17] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *ICML*, 2011.

[18] P. Pan, Z. Xu, Y. Yang, F. Wu, and Y. Zhuang. Hierarchical recurrent neural encoder for video representation with application to captioning. In *CVPR*, 2016.

[19] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui. Jointly modeling embedding and translation to bridge video and language. In *CVPR*, 2016.

[20] V. Ramanathan, K. Tang, G. Mori, and F. F. Li. Learning temporal embeddings for complex video analysis. In *ICCV*, 2015.

[21] A. Rohrbach, M. Rohrbach, and B. Schiele. The long-short story of movie description. In *arXiv*, 2015.

[22] R. Shetty. Natural language description of images and videos. Master's thesis.

[23] N. Srivastava and R. Salakhutdinov. Multimodal learning with deep boltzmann machines. In *NIPS*, 2012.

[24] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *arXiv*, 2016.

[25] C. Szegedy, W. Liu, Y. Jia, and P. Sermanet. Going deeper with convolutions. In *CVPR*, 2015.

[26] A. D. Vasili Ramanishka. Multimodal video description. In *ACM MM*, 2016.

[27] R. Vedantam, C. L. Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, 2015.

[28] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to sequence – video to text. In *ICCV*, 2015.

[29] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko. Translating videos to natural language using deep recurrent neural networks. In *NAACL-HLT*, 2015.

[30] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, 2015.

[31] M. Wang, L. Song, X. Yang, and C. Luo. A parallel-fusion rnn-lstm architecture for image caption generation. In *ICIP*, 2016.

[32] Q. Wu, C. Shen, A. V. D. Hengel, L. Liu, and A. Dick. What value do explicit high level concepts have in vision to language problems? In *CVPR*, 2016.

[33] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015.

[34] L. Yao, N. Ballas, K. Cho, J. R. Smith, Y. Bengio, L. Yao, N. Ballas, K. Cho, J. R. Smith, and Y. Bengio. Trainable performance upper bounds for image and video captioning. In *arXiv*, 2015.

[35] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. Describing videos by exploiting temporal structure. In *ICCV*, 2015.

[36] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo. Image captioning with semantic attention. In *CVPR*, 2016.

[37] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu. Video paragraph captioning using hierarchical recurrent neural networks. In *CVPR*, 2016.

[38] M. D. Zeiler. Adadelta: An adaptive learning rate method. In *arXiv*, 2012.

[39] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua. Visual translation embedding network for visual relation detection. In *CVPR*, 2017.

[40] H. Zhang, X. Shang, W. Yang, H. Xu, H. Luan, and T.-S. Chua. Online collaborative learning for open-vocabulary visual classifiers. In *CVPR*, 2016.