Weakly Supervised Dense Video Captioning

Zhiqiang Shen[†,*], Jianguo Li[‡], Zhou Su[‡], Minjun Li[†]
Yurong Chen[‡], Yu-Gang Jiang[†], Xiangyang Xue[†]
[†]Shanghai Key Laboratory of Intelligent Information Processing
School of Computer Science, Fudan University
[‡]Intel Labs China
[†]{zhiqiangshen13, minjunli13, ygj, xyxue}@fudan.edu.cn
[‡]{jianguo.li, zhou.su, yurong.chen}@intel.com

Abstract

This paper focuses on a novel and challenging vision task, dense video captioning, which aims to automatically describe a video clip with multiple informative and diverse caption sentences. The proposed method is trained without explicit annotation of fine-grained sentence to video region-sequence correspondence, but is only based on weak *video-level sentence annotations. It differs from existing video captioning systems in three technical aspects. First, we propose lexical fully convolutional neural networks (Lexical-FCN) with weakly supervised multi-instance multi-label learning to weakly link video regions with lexical labels. Second, we introduce a novel submodular maximization scheme to generate multiple informative and diverse region-sequences based on the Lexical-FCN outputs. A winner-takes-all scheme is adopted to weakly associate sentences to region-sequences in the training phase. Third, a sequence-to-sequence learning based language model is trained with the weakly supervised information obtained through the association process. We show that the proposed method can not only produce informative and diverse dense captions, but also outperform state-of-the-art single video captioning methods by a large margin.*

1. Introduction

Automatically describing images or videos with natural language sentences has recently received significant attention in the computer vision community. For images, researchers have investigated image captioning with one sentence [52, 50, 5, 1, 7, 26, 47] or multiple sentences [17, 16, 33]. For videos, most of the works focused on generating only one caption for a short video clip using methods based on mean pooling of features over frames [49], the soft-attention scheme [53], or visual-semantic embedding between visual feature and language [30]. Some recent works further considered the video temporal structure, such as the sequence-to-sequence learning (S2VT) [48] and hierarchical recurrent neural encoder [29].

Figure 1: Illustration of dense video captioning (*DenseVidCap*). Each region-sequence is highlighted in white bounding boxes along with corresponding predicted sentence in its bottom. The ground-truth sentences are presented on the right.

However, using a single sentence cannot well describe the rich contents within images/videos. The task of dense image captioning is therefore proposed, which aims to generate multiple sentences for different detected object locations in images [16, 17, 19]. However, this setting requires region-level caption annotations for supervised training purpose. As is well-known, videos are much more complex than images since the additional temporal dimension could provide informative contents such as different viewpoints of objects, object motions, procedural events, etc. It is fairly expensive to provide region-sequence level sentence annotations for dense video captioning. The lack of such annotations has largely limited the much-needed progress of dense video captioning. Our work in this paper is motivated by the following two questions. First, most existing datasets have multiple video-level sentence annotations, which usu-

*This work was done when Zhiqiang Shen was an intern at Intel Labs China. Jianguo Li and Yu-Gang Jiang are the corresponding authors.

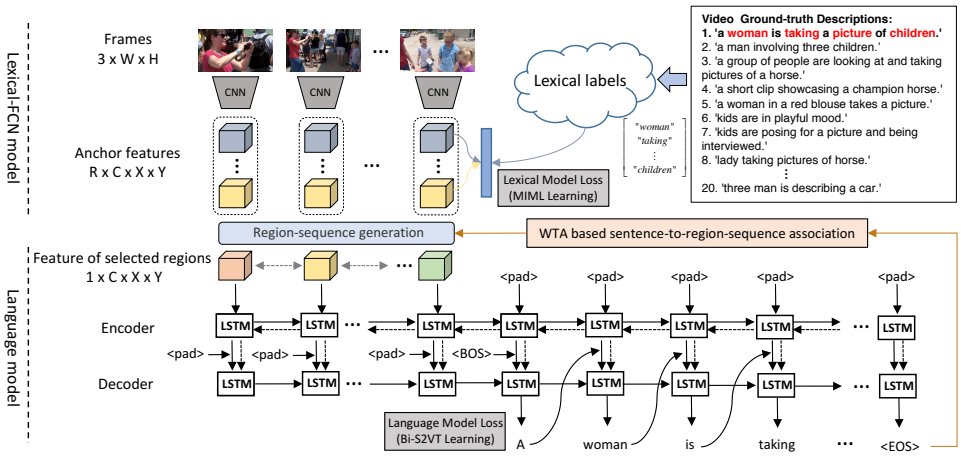


Figure 2: Overview of our *Dense Video Captioning* framework. In the language model, <BOS> denotes the begin-of-sentence tag and <EOS> denotes the end-of-sentence tag. We use zeros as <pad> when there is no input at the time step. Best viewed in color.

ally describe very diverse aspects (regions/segments) of the video clip. However, existing video captioning methods simply represented all sentence descriptions with one global visual representation. This one-to-many mapping is far from accurate. It is thus very interesting to investigate if there is an automatic way to (even weakly) associate sentence to region-sequence. Second, is it possible to perform dense video captioning with those weakly associations (without strong 1-to-1 mapping between sentences and region-sequence) in a weakly supervised fashion?

In this paper, we propose an approach to generate multiple diverse and informative captions by weakly supervised learning from only the video-level sentence annotations. Figure 2 illustrates the architecture of the proposed approach, which consists of three major components: visual sub-model, region-sequence sub-model and language sub-model. The visual sub-model is a lexical-FCN trained with weakly supervised multi-instance multi-label learning, which builds the weak mapping between sentence lexical words and grid regions. The second component solves the region-sequence generation problem. We propose submodular maximization scheme to automatically generate informative and diverse region-sequences based on Lexical-FCN outputs. A winner-takes-all scheme is proposed to weakly associate sentences to region-sequences in the training phase. The third component generates sentence output for each region-sequence with a sequence-to-sequence learning based language model [48]. The main contributions are summarized as follows:

(1) To the best of our knowledge, this is the first work for dense video captioning with only video-level sentence annotations.

(2) We propose a novel dense video captioning approach, which models visual cues with Lexical-FCN, discovers region-sequence with submodular maximization, and decodes language outputs with sequence-to-sequence learning. Although the approach is trained with weakly supervised signal, we show that *informative* and *diverse* captions can be produced.

(3) We evaluate dense captioning results by measuring the performance gap to oracle results, and diversity of the dense captions. The results clearly verify the advantages of the proposed approach. Especially, the best single caption by the proposed approach outperforms the state-of-the-art results on the MSR-VTT challenge by a large margin.

2. Related Work

Multi-sentence description for videos has been explored in various works recently [37, 41, 54, 3, 18]. Most of these works [54, 41, 37] focused on generating a long caption (story-like), which first temporally segmented the video with action localization [41] or different levels of details [37], and then generated multiple captions for those segments and connected them with natural language processing techniques. However, these methods simply considered the temporally segmentation, and ignored the frame-level region attention and the motion-sequence of region-level objects. Yu *et al.* [54] considered both the temporal and spatial attention, but still ignored the association or alignment of the sentences and visual locations. In contrast, this paper tries to exploit both the temporal and spatial region information and further explores the correspondence between sentences and region-sequences for more accurate modeling.

Lexical based CNN model is of great advantages over the ImageNet based CNN model [39] in image/video captioning, since the ImageNet based CNN model only captures a limited number of object concepts, while the lexical based CNN model is able to capture all kinds of semantic concepts (nouns for objects and scenes, adjective for shape and attributes, verb for actions, etc). It is non-trivial

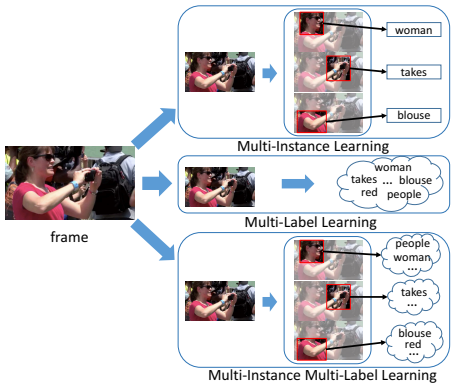


Figure 3: Three paradigms of learning a lexical model.

to adopt/fine-tune the existing ImageNet CNN models with lexical output. Previous works [7, 1, 47, 38, 19] have proposed several ways for this purpose. For instance, [7] adopted a weakly supervised multiple instance learning (MIL) approach [27, 56] to train a CNN based word detector without the annotations of image-region to words correspondence; and [1] applied a multiple label learning (MLL) method to learn the CNN based mapping between visual inputs and multiple concept tags.

Sequence to sequence learning with long short-term memory (LSTM) [13] was initially proposed in the field of machine translation [43]. Venugopalan *et al.* (S2VT) [48] generalized it to video captioning. Compared with contemporaneous works [53, 51, 30] which require additional temporal features from 3D ConvNets [45], S2VT can directly encode the temporal information by using LSTM on the frame sequence, and no longer needs the frame-level soft-attention mechanism [53]. This paper adopts the S2VT model [48] with a bi-directional formulation to improve the encoder quality, which shows better performance than the vanilla S2VT model in our experiments.

3. Approach

Our ultimate goal is to build a system that describes input videos with dense caption sentences. The challenges are two folds. First, we do not have fine-grained training-data annotations which link sentence captions to region-sequences. Second, we must ensure the generated sentences being informative and diverse. As discussed earlier, the proposed approach consists of three components (see Figure 2): lexical-FCN based visual model, region-sequence generation and language model. We elaborate each of them in the following.

3.1. Lexical FCN Model

3.1.1 Multi-instance Multi-label Lexical Model

We adopt multi-instance multi-label learning (MIMLL) to train our lexical model, which could be viewed as a combination of word detection [7] (MIL) and deep lexical classification [1] (MLL). Figure 3 illustrates the comparison of the three methods.

Multi-instance learning [27, 56, 7] assumes that the word label y_i^w is assigned to a bag of instances $\mathbf{X}_i = \{\mathbf{x}_{i1}, \ldots, \mathbf{x}_{ij}\}$ where i is the bag index, $\mathbf{x}_{ij} \in \mathbb{R}^d$ is a d-dimensional feature vector for the j-th instance. The word detection method [7] used fc7 features of VGG-16 as the instance representations. The bag is positive with a word label $y_i^w = 1$ if at least one of the instances in $\mathbf{X}_i$ contains the word w, although it is not exactly known which one contains the word. The bag is negative with label $y_i^w = 0$ if no instance contains the word w.

Multi-label learning assumes that each instance $\mathbf{x}_i$ has multiple word labels: $\mathbf{y}_i = \{y_i^1, \ldots, y_i^{\mathbf{k}}\}$ where $\mathbf{k}$ is the number of labels. For this purpose, we usually train a deep neural network with a sigmoid cross-entropy loss [1].

Multi-instance multi-label learning [57] is a natural generalization of MIL. It takes as input pairs $\{\mathbf{X}_i, \mathbf{y}_i\}$, where each $\mathbf{X}_i$ is a bag of instances labeled with a set of words $\mathbf{y}_i = \{y_i^1, \ldots, y_i^{\mathbf{k}}\}$. In MIMLL, each instance usually has one or multiple word labels. For instance, we can use "woman", "people", "human" or other synonyms in the lexicon to describe a female (see Figure 3 for one example). Now we define the loss function for a bag of instances. As each bag has multiple word labels, we adopt the cross-entropy loss to measure the multi-label errors:

$$L(\mathbf{X}, \mathbf{y}; \theta) = -\frac{1}{N}\sum_{i=1}^{N}[\mathbf{y}_i \cdot \log \hat{\mathbf{p}}_i + (1 - \mathbf{y}_i) \cdot \log(1 - \hat{\mathbf{p}}_i)], \quad (1)$$

where θ is the model parameters, N is the number of bags, $\mathbf{y}_i$ is the label vector for bag $\mathbf{X}_i$, and $\hat{\mathbf{p}}_i$ is the corresponding probability vector. We weakly label the bag as negative when all instances in the bag are negative, and thus use a noisy-OR formulation [12, 27] to combine the probabilities that the individual instances in the bag are negative:

$$\hat{p}_i^w = P(y_i^w = 1|\mathbf{X}_i; \theta) = 1 - \prod_{\mathbf{x}_{ij} \in \mathbf{X}_i}(1 - P(y_i^w = 1|\mathbf{x}_{ij}; \theta)), \quad (2)$$

where $\hat{p}_i^w$ is the probability when word w in the i-th bag is positive. We define a sigmoid function to model the individual word probability:

$$P(y_i^w = 1|\mathbf{x}_{ij}; \theta) = \sigma(\mathbf{w}_w \mathbf{x}_{ij} + \mathbf{b}_w), \quad (3)$$

where $\mathbf{w}_w$ is the weight matrices, $\mathbf{b}_w$ is the bias vector, and $\sigma(x) = 1/(1 + \exp(-x))$ is the logistic function. In our Lexical-FCN model, we use the last pooling layer (pool5 for ResNet-50) as the representation of instance $\mathbf{x}_{ij}$, which will be elaborated in the following sections.

3.1.2 Details of Lexical-FCN

Lexical-FCN model builds the mapping between frame regions and lexical labels. The first step of Lexical-FCN is

to build a lexical vocabulary from the video caption training set. We extract the part-of-speech [44] of each word in the entire training dataset. These words may belong to any part of sentences, including nouns, verbs, adjectives and pronouns. We treat some of the most frequent functional words[1] as stop words, and remove them from the lexical vocabulary. We keep those remaining words appearing at least five times in the MSR-VTT training set, and finally obtain a vocabulary $\mathcal{V}$ with 6,690 words.

The second step of Lexical-FCN is to train the CNN models with MIMLL loss described above. Instead of training from scratch, we start from some state-of-the-art ImageNet models like VGG-16 [42] or ResNet-50 [11], and fine-tune them with the MIMLL loss on the MS-VTT training set. For VGG-16, we re-cast the fully connected layers to convolutions layers to obtain a FCN. For ResNet-50, we remove final softmax layer and keep the last mean pooling layer to obtain a FCN.

3.1.3 Regions from Convolutional Anchors

In order to obtain the dense captions, we need grounding the sentences to sequences of ROI (regions of interest). Early solutions in object detection adopt region proposal algorithms to generate region candidates, and train a CNN model with an additional ROI pooling layer [10, 8, 36]. This cannot be adopted in our case, since we do not have the bounding box ground-truth for any words or concepts required in the training procedure. Instead, we borrow the idea from YOLO [35], and generate coarse region candidates from anchor points of the last FCN layer [24, 7]. In both training and inference phases, we sample the video frames and resize both dimensions to 320 pixels. After feeding forward through the FCN, we get a 4×4 response map (4096 channels for VGG-16 and 2048 channels for ResNet-50). Each anchor point in the response map represents a region in the original frame. Unlike object detection approaches, the bounding-box regression process is not performed here since we do not have the ground-truth bounding boxes. We consider the informative region-sequence generation problem directly starting with these 16 very-coarse grid regions.

3.2. Region-Sequence Generation

Regions between different frames are matched and connected sequentially to produce *region-sequences*. As each frame has 16 coarse regions, even if each video clip is downsampled to 30 frames, we have to face a search space of size 16^{30} for region-sequence generation. This is intractable for common methods even for the training case that has video-level sentence annotations. However, our Lexical-FCN model provides the lexical descriptions for each region at every frame, so that we can consider the problem from a different perspective.

[1]Functional words are 'is', 'are', 'at', 'on', 'in', 'with', 'and' and 'to'.

3.2.1 Problem Formulation

We formulate the region-sequence generation task as a sub-set selection problem [22, 9], in which we start from an empty set, and sequentially add one most informative and coherent region at each frame into the subset, and in the meantime ensure the diversity among different region-sequences. Let $\mathcal{S}_\mathbf{v}$ denote the set of all possible region sequences of video $\mathbf{v}$, $\mathcal{A}$ is a region-sequence sub-set, i.e., $\mathcal{A} \subseteq \mathcal{S}_\mathbf{v}$. Our goal is to select a region-sequence $\mathcal{A}^*$, which optimizes an objective R:

$$\mathcal{A}^* = \arg\max_{\mathcal{A}\subseteq\mathcal{S}_\mathbf{v}} R(\mathbf{x}_\mathbf{v}, \mathcal{A}), \tag{4}$$

where $\mathbf{x}_\mathbf{v}$ are all region feature representations of video $\mathbf{v}$. We define $R(\mathbf{x}_\mathbf{v}, \mathcal{A})$ as linear combination objectives

$$R(\mathbf{x}_\mathbf{v}, \mathcal{A}) = {\mathbf{w}_\mathbf{v}}^T \mathbf{f}(\mathbf{x}_\mathbf{v}, \mathcal{A}), \tag{5}$$

where $\mathbf{f} = [f_{inf}, f_{div}, f_{coh}]^T$, which describe three aspects of the region-sequence, i.e., *informative*, *diverse* and *coherent*. The optimization problem of Eq-4 quickly becomes intractable when $\mathcal{S}_\mathbf{v}$ grows exponentially with the video length. We restrict the objectives $\mathbf{f}$ to be monotone submodular function and $\mathbf{w}_\mathbf{v}$ to be non-negative. This allows us to find a near optimal solution in an efficient way.

3.2.2 Submodular Maximization

We briefly introduce submodular maximization and show how to learn the weights $\mathbf{w}_\mathbf{v}$. A set function is called submodular if it fulfills the *diminishing returns* property. That means, given a function f and arbitrary sets $\mathcal{A} \subseteq \mathcal{B} \subseteq \mathcal{S}_\mathbf{v} \setminus r$, f is submodular if it satisfies:

$$f(\mathcal{A} \cup \{r\}) - f(\mathcal{A}) \geq f(\mathcal{B} \cup \{r\}) - f(\mathcal{B}). \tag{6}$$

Linear combination of submodular functions is still submodular for non-negative weights. For more details, please refer to [28, 22].

Submodular functions have many properties that are similar to convex or concave functions, which are desirable for optimization. Previous works [28, 22, 9] have shown that maximizing a submodular function with a greedy algorithm yields a good approximation to the optimal solution. In this paper, we apply a commonly used cost-effective lazy forward (CELF) method [22] for our purpose. We defined a marginal gain function as

$$\begin{aligned}\mathcal{L}(\mathbf{w}_\mathbf{v}; r) &= R(\mathcal{A}_\mathbf{t-1} \cup \{r\}) - R(\mathcal{A}_\mathbf{t-1}) \\ &= {\mathbf{w}_\mathbf{v}}^T \mathbf{f}(\mathbf{x}_\mathbf{v}, \mathcal{A}_\mathbf{t-1} \cup \{r\}) - {\mathbf{w}_\mathbf{v}}^T \mathbf{f}(\mathbf{x}_\mathbf{v}, \mathcal{A}_\mathbf{t-1}).\end{aligned} \tag{7}$$

The CELF algorithm starts with an empty sequence $\mathcal{A}_0 = \emptyset$, and adds the region r_t at step t into region-sequence which can maximize the marginal gain:

$$\mathcal{A}_\mathbf{t} = \mathcal{A}_\mathbf{t-1} \cup \{r_t\};\ \ r_t = \arg\max_{r\in\mathcal{S}_t} \mathcal{L}(\mathbf{w}_\mathbf{v}; r), \tag{8}$$

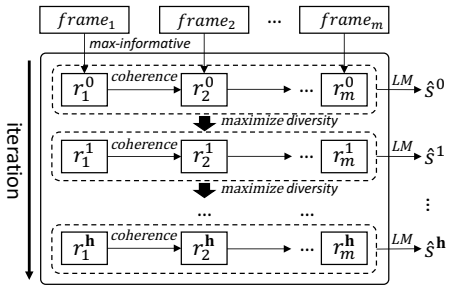


Figure 4: Illustration of region-sequence generation. r_i^j is the j-th region-sequence in i-th frame and '*LM*' denotes language model.

where $\mathcal{S}_t$ means region sets in frame-t.

Given N pairs of known correspondences $\{(\mathbf{r}, \mathbf{s})\}$, we optimize $\mathbf{w_v}$ with the following objective:

$$\min_{\mathbf{w_v} \geq 0} \frac{1}{N} \sum_{i=1}^{N} \max_{r \in \mathbf{r}_i} \mathcal{L}_i(\mathbf{w_v}; r) + \frac{\lambda}{2} \|\mathbf{w_v}\|^2, \tag{9}$$

where the max-term is a generalized hinge loss, which means ground-truth or oracle selected region r should have a higher score than any other regions by some margin.

Our training data do not have $(\mathbf{r}, \mathbf{s})$ pairs, i.e., the sentence to region-sequence correspondence. We solve this problem in a way that is similar to the alternative directional optimization: (1) we initialize $\mathbf{w_v} = \mathbf{1}$ (all elements equals to 1); (2) we obtain a region-sequence with submodular maximization with that $\mathbf{w_v}$; (3) we weakly associate sentence to region-sequence with a winner-takes-all (WTA) scheme (described later); (4) we refine $\mathbf{w_v}$ with the obtained sentence to region-sequence correspondence; (5) we repeat step-2∼4 until $\mathbf{w_v}$ is converged.

The WTA scheme works in four steps when giving a ground-truth sentence $\mathbf{s}$. *First*, we extract the lexical labels from $\mathbf{s}$ based on the vocabulary $\mathcal{V}$, and form a lexical subset $\mathcal{V}_\mathbf{s}$. *Second*, we obtain probability of word $w \in \mathcal{V}_\mathbf{s}$ for the i-th region-sequence by $p_i^w = \max_j p_{ij}^w$, where p_{ij}^w is the probability of word w in the j-th frame, which is in fact from the Lexical-FCN output for each region. *Third*, we threshold p_i^w with a threshold θ, i.e., redefining $p_i^w = 0$ if $p_i^w < \theta$ (θ = 0.1 in our studies). *Last*, we compute the matching score by

$$f_i = \sum_{w \in \mathcal{V}_\mathbf{s};\ p_i^w \geq \theta} p_i^w, \tag{10}$$

and obtain the best region-sequence by $i^* = \arg\max_i f_i$. This objective suggests that we should generate region-sequences having high-scored words in the sentences.

3.2.3 Submodular Functions

Based on the properties of submodular function [25, 28], we describe how to define the three components as follows.

Informativeness of a region-sequence is defined as the sum of each region's informativeness:

$$f_{\text{inf}}(\mathbf{x_v}, \mathcal{A}_t) = \sum\nolimits_w p^w; \quad p^w = \max_{i \in \mathcal{A}_t} p_i^w. \tag{11}$$

If video-level sentence annotations are known either in the training case or by an oracle, we replace the definition with Eq-10, which limits words by the sentence vocabulary $\mathcal{V}_\mathbf{s}$.

Coherence aims to ensure the temporal coherence of the region-sequence, since significant changes of region contents may confuse the language model. Similar to some works in visual tracking [2, 14], we try to select regions with the smallest changes temporally, and define the coherence component as

$$f_{coh} = \sum\nolimits_{r_s \in \mathcal{A}_{t-1}} \langle \mathbf{x}_{r_t}, \mathbf{x}_{r_s} \rangle, \tag{12}$$

where $\mathbf{x}_{r_t}$ is the feature of region r_t at t-th step, $\mathbf{x}_{r_s}$ is one of the region feature in the previous $(t-1)$ steps, and $\langle , \rangle$ means dot-production operation between two normalized feature vectors. In practice, we also limit the search space of region r_t within the 9 neighborhood positions of the region from the previous step.

Diversity measures the degree of difference between a candidate region-sequence and all the existing region-sequences. Suppose $\{p_i^w\}_{i=1}^N$ are the probability distribution of the existing N region-sequences and q^w is the probability distribution of a candidate region-sequence, the diversity is defined with the Kullback-Leibler divergence as

$$f_{div} = \sum_{i=1}^{N} \int_w p_i^w \log \frac{p_i^w}{q^w} dw. \tag{13}$$

We initially pick the most informative region-sequence, and feed it to a language model (*LM*) for sentence output. Then we iteratively pick a region-sequence which maximizes diversity to generate multiple sentence outputs. Figure 4 illustrates our region-sequence generation method. The detailed algorithm is given in the supplementary file.

3.3. Language Models

We model the weakly associated temporal structure between region-sequence and sentence with the sequence-to-sequence learning framework (S2VT) [48], which is an encoder-decoder structure. S2VT encodes visual feature of region-sequences $\vec{V} = (\mathbf{v_1}, \cdots, \mathbf{v_T})$ with LSTM, and decodes the visual representation into a sequence of output words $\vec{u} = (u_1, \cdots, u_S)$. LSTM is used to model a sequence in both the encoder part and the decoder part. As a variant of RNN, LSTM is able to learn long-term temporal information and dependencies that traditional RNN is difficult to capture [13]. Our LSTM implementation is based on [55] with dropout regularization on all LSTM units (dropout ratio 0.9).

We extend the original S2VT with bi-directional encoder, so that the S2VT learning in Figure 2 stacks three LSTM models. The first LSTM encodes forward visual feature sequence $\{\vec{V}\}$, and the second encodes the reverse visual feature sequence $\{\overleftarrow{V}\}$. These two LSTM networks form the encoder part. We will show the benefit of bi-direction LSTM encoding later. The third LSTM decodes

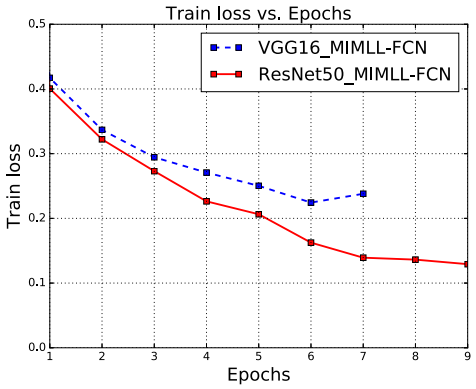


Figure 5: The lexical training loss on the MSR-VTT dataset.

visual codes from both the forward pass and backward pass into sequences of words (sentences).

To further improve accuracy, we propose a category-wise language model extension. Videos may belong to different categories, such as news, sports, etc. Different video category has very different visual patterns and sentence styles. The category-wise language model is defined as

$$\mathbf{s}^* = \arg\max_{\mathbf{s}} P(\mathbf{s}|c, \mathbf{v})P(c|\mathbf{v}), \tag{14}$$

where c is the category label, $\mathbf{v}$ is the video feature representation, and $\mathbf{s}$ is the predicted sentence. $P(\mathbf{s}|c, \mathbf{v})$ is the probability conditional on category c and video $\mathbf{v}$, and $P(c|\mathbf{v})$ is prior confidence of video $\mathbf{v}$ belongs to a category c, which can be obtained from a general video categorization model. The category-wise language model can be viewed as max-a-posterior estimation.

4. Experiments

4.1. Dataset and Implementation Details

We conduct experiments on the MSR-VTT dataset [51], which is a recently released large-scale video caption benchmark. This dataset contains 10,000 video clips (6,513 for training, 497 for validation and 2,990 for testing) from 20 categories, including news, sports, etc. Each video clip is manually annotated with 20 natural sentences. This is currently the largest video captioning dataset in terms of the amounts of sentences and the size of the vocabulary. Although this dataset was mainly used for evaluating single sentence captioning results, we assume that the 20 sentences for each clip contain very diversified annotations and can be used in the task of dense captioning (with some redundancy as will be discussed later).

For the evaluation of single captioning, the authors of this benchmark proposed machine translation based metrics like METEOR [21], BLEU@1-4 [32], ROUGE-L [23] and CIDEr [46]. For dense video captioning results, we propose our own evaluation protocol to justify the results.

All the training and testing are done on an Nvidia TitanX GPU with 12GB memory. Our model is efficient during the testing stage. It can process a 30-frame video clip in about 840ms on the TitanX GPU, including 570ms for CNN feature extraction, 90ms for region-sequence generation, and 180ms for language model.

Model	METEOR	BLEU@4	ROUGE-L	CIDEr
Unidirectional (VGG-16)	25.2	32.7	56.0	31.1
Bi-directional (VGG-16)	25.4	32.8	56.5	32.9
Unidirectional (ResNet-50)	25.7	32.1	56.4	32.5
Bi-directional (ResNet-50)	25.9	33.7	56.9	32.6

Table 1: Single sentence captioning accuracy by bi-/uni-directional encoder on the *validation set* of MSR-VTT.

Model	METEOR	BLEU@4	ROUGE-L	CIDEr
MIL (bi-directional)	23.3	28.7	53.1	24.4
MIMLL (bi-directional)	25.9	33.7	56.9	32.6

Table 2: Single sentence captioning accuracy by MIL and MIMLL on the *validation set* of MSR-VTT.

4.2. Ablation Studies on Single Sentence Captioning

We first evaluate the effect of several design components through single sentence captioning experiments, which produce a caption with the maximal informative score defined by Eq-11 (i.e., $\hat{\mathbf{s}}^0$ in Figure 4).

Effectiveness of Network Structure. We compare VGG-16 and ResNet-50 for the Lexical-FCN model. Due to the GPU memory limitation, we do not try a deeper network like ResNet-152. Figure 5 shows that ResNet-50 achieves better training loss than VGG-16, which is consistent with their results on ImageNet. Table 1 summarizes the single sentence captioning results on the MSR-VTT validation set by two networks. As can be seen, in all the cases, ResNet-50 performs better than VGG-16. Based on these results, we choose ResNet-50 as our network structure in the following studies when there is no explicit statement.

Effectiveness of Bi-directional Encoder. Next we compare the performances of *bi-directional* and *unidirectional* S2VT models for language modeling. Results are also shown in Table 1. It is obvious that *bi-directional* model outperforms *unidirectional* model on all the evaluated metrics. The benefit of *bi-directional* model is not that significant. We conjecture that this is due to the fact that the region-sequences already include enough temporal and local information. Nevertheless, for better accuracy, all the following studies adopt the *bi-directional* model.

Effectiveness of MIMLL. Our Lexical-FCN model is trained on video frames. Compared with image-level lexical learning [7, 1], our setting is much more challenging since the sentences are on the video-level, and it is hard to determine which words correspond to which frames. Here we show the effectiveness of the MIMLL in two aspects. First, we compare the single captioning results by MIMLL and MIL in Table 2. We can see that MIMLL achieves better accuracy than MIL on all the four metrics. Second, we compare the word detection accuracy of MIMLL and MIL. We first compute the max-probability of each word within

the region-sequence. If the max-probability of a word is greater than a threshold (0.5), we claim that the word is detected. We observe that MIMLL is better in detecting accuracy than MIL in this study (43.1% vs 41.3%). Both results demonstrate the effectiveness of the proposed MIMLL for the Lexical-FCN model.

Effectiveness of Category-wise Language Model. All the previous studies are based on language model without using video category information. Here, we study the benefit of the category-wise language model, as defined in Eq-14. Results are shown in the 2nd last and the 3rd last rows in Table 3. We observe that the category-wise language model achieves much better accuracy than that without category-wise modeling. The benefit is due to that category information provides a strong prior about video content.

Comparison with State-of-the-arts. We also compare our single sentence captioning results with the state-of-the-art methods in MSR-VTT benchmark. For better accuracy, this experiment adopts data augmentation during the training procedure, similar to the compared methods. We pre-process each video clip to 30-frames with different sampling strategies (random, uniform, etc), and obtain multiple instances for each video clip.

We first compare our method with mean-pooling [49], soft-attention [53] and S2VT [48] on the validation set of MSR-VTT. All these alternative methods have source codes available for easy evaluation. Results are summarized in Table 3. Our baseline approach (the 2nd last row) is significantly better than these 3 methods. We also compare with the top-4 results from the MSR-VTT challenge in the table, including v2t_navigator [15], Aalto [40], VideoLAB [34] and ruc_uva [6][2], which are all based on features from multiple cues such as action features like C3D and audio features like Bag-of-Audio-Words (BoAW) [31]. Our baseline has on-par accuracy to the state-of-the-art methods. For fair comparison, we integrate C3D action features and audio features together with our lexical features and feed them into the language model. Clearly better results are observed.

In Table 4, we compare our results on the test set of MSR-VTT with the top-4 submissions in the challenge leaderboard, where we can see that similar or better results are obtained in all the four evaluated metrics.

4.3. Evaluation of Dense Captioning Results

The proposed approach can produce a set of region-sequences with corresponding multiple captions for an input video clip. Besides qualitative results in Figure 1 and the supplementary file, we evaluate the results quantitatively in two aspects: 1) performance gap between automatic results and oracle results, and 2) diversity of the dense captions.

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

Model	METEOR	BLEU@4	ROUGE-L	CIDEr
Mean-Pooling [49]	23.7	30.4	52.0	35.0
Soft-Attention [53]	25.0	28.5	53.3	37.1
S2VT [48]	25.7	31.4	55.9	35.2
ruc-uva [6]	27.5	39.4	60.0	48.0
VideoLAB [34]	27.7	39.5	61.0	44.2
Aalto [40]	27.7	41.1	59.6	46.4
v2t_navigator [15]	29.0	43.7	61.4	45.7
Ours w/o category	27.7	39.0	60.1	44.0
Ours category-wise	28.2	40.9	61.8	44.7
Ours + C3D + Audio	**29.4**	**44.2**	**62.6**	**50.5**

Table 3: Comparison with state of the arts on the *validation set* of MSR-VTT dataset. See texts for more explanations.

Model	METEOR	BLEU@4	ROUGE-L	CIDEr
ruc-uva [6]	26.9	38.7	58.7	45.9
VideoLAB [34]	27.7	39.1	60.6	44.1
Aalto [40]	26.9	39.8	59.8	45.7
v2t_navigator [15]	28.2	40.8	60.9	44.8
Ours	**28.3**	**41.4**	**61.1**	**48.9**

Table 4: Comparison with state of the arts on the *test set* of MSR-VTT dataset. See texts for more explanations.

4.3.1 Performance Gap with Oracle Results

We measure the quality of dense captioning results by the performance gap between our automatic results and oracle results. Oracle leverages information from ground-truth sentences to produce the caption results. Oracle information could be incorporated in two settings. *First*, similar to the training phase, during inference oracle uses the ground-truth information to guide sentence to region-sequence association. *Second*, oracle uses the ground-truth sentences to measure the goodness of each caption sentence using metrics like METEOR and CIDEr, and re-ranks the sentences according to their evaluation scores. It is obvious that the oracle results are the upper bound of the automatic method.

Inspired by the evaluation of dense image captioning [16], we use averaged precision (AP) to measure the accuracy of dense video captioning. We compute the precision in terms of all the four metrics (METEOR, BLEU@4, ROUGE-L and CIDEr) for every predicted sentence, and obtain average values of the top-5 and top-10 predicted sentences. The gap of AP values between oracle results and our automatic results will directly reflect the effectiveness of the automatic method.

For our automatic method, the output sentences need to be ranked to obtain the top-5 or top-10 sentences. Similar to [40], we train an evaluator network in a supervised way for this purpose, since submodular maximization does not ensure that sentences are generated in quality decreasing order. Table 5 lists the comparative results on the validation set of MSR-VTT using three strategies: (1) oracle for both sentence to region-sequence association and sentence re-ranking (OSR + ORE in short); (2) dense video captioning + oracle re-ranking (Dense + ORE in short); (3) fully automatic dense video captioning method (DenseVidCap).

Results indicate that the dense video captioning + ora-

Model	METEOR	BLEU@4	ROUGE-L	CIDEr
Averaged Precision of Top-5 Sentences				
OSR + ORE	29.3 (100)	42.3 (100)	64.1 (100)	43.4 (100)
Dense + ORE	28.0 (95.6)	40.8 (96.5)	62.8 (97.9)	41.9 (96.5)
DenseVidCap	26.5 (90.4)	34.8 (82.3)	57.7 (90.0)	37.3 (85.9)
Averaged Precision of Top-10 Sentences				
OSR + ORE	27.9 (100)	38.8 (100)	61.4 (100)	39.1 (100)
Dense + ORE	26.4 (94.6)	36.6 (94.3)	59.5 (96.9)	36.6 (93.6)
DenseVidCap	26.1 (93.5)	33.6 (86.6)	57.1 (93.0)	35.3 (90.3)

Table 5: Averaged precision of the top-5/10 sentences generated on the *validation set* of MSR-VTT. **OSR** means oracle for sentence to region-sequence association, and **ORE** means oracle for sentence re-ranking. The values in the parenthesis indicate the relative percentage (%) to the fully oracle results (OSR+ORE).

cle re-ranking could reach $\geq$95% relative accuracy of the "fully" oracle results (OSR+ORE) on all the metrics for the top-5 sentences, and $\geq$93% relative accuracy to the fully oracle results for the top-10 sentences. The fully automatic method (our DenseVidCap) can consistently achieve more than 82% relative accuracy of the oracle results on both top-5 and top-10 settings. This is very encouraging as the performance gap is not very large, especially considering that our model is trained with weakly annotated data. One important reason that causes the gap is that the evaluator network is not strong enough when compared with oracle re-ranking, which is a direction for further performance improvement.

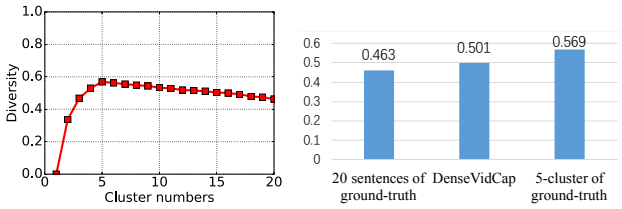
4.3.2 Diversity of Dense Captions

The diversity of the generated captions is critical for dense video captioning. We evaluate diversity from its opposite – the similarity of the captions. A common solution is to determine the similarity between pairs of captions, or between one caption to a set of other captions. Here we consider similarity from the apparent semantic relatedness of the sentences. We use the Latent semantic analysis (LSA) [4] which first generates sentence bag-of-words (BoW) representation, and then maps it to LSA space to represent a sentence. This method has demonstrated its effectiveness in measuring document distance [20]. Based on the representation, we compute cosine similarity between two LSA vectors of sentences. Finally, the diversity is calculated as:

$$D_{div} = \frac{1}{n} \sum_{\mathbf{s}^i, \mathbf{s}^j \in \mathbf{S};\ i \neq j} (1 - \langle \mathbf{s}^i, \mathbf{s}^j \rangle), \tag{15}$$

where $\mathbf{S}$ is the sentence set with cardinality n, and $\langle \mathbf{s}^i, \mathbf{s}^j \rangle$ denotes the cosine similarity between $\mathbf{s}^i$ and $\mathbf{s}^j$.

As aforementioned, we assume that the multiple video-level captions cover diversified aspects of the video content with some redundancy. The diversity metric can be applied in two aspects: evaluating the diversity degree of (1) our dense captioning results and (2) the manually generated captions in the ground-truth. Some of the manually annotated ground-truth sentences on MSR-VTT are redundant. For instance, the captions "a woman is surfing" and "a woman surfing in the ocean" are more or less the same. We remove the redundant captions by clustering on each video caption set with the LSA based representation. Different clustering numbers can lead to different diversity scores. As shown in Figure 6(a), five clusters give the highest diversity score on this dataset.



(a) (b)

Figure 6: (a) Diversity score of clustered ground-truth captions under different cluster numbers; (b) Diversity score comparison of our automatic method (middle) and the ground-truth.

We compare the diversity score of our automatic results with that of the ground-truth sentences in Figure 6(b). As can be seen, our DenseVidCap achieves better diversity score (0.501) than that of the original 20 ground-truth sentences (0.463), but is slightly worse than that of the best of the clustered ground-truth sentences (0.569). Please refer to Figure 1 and the supplementary file for some qualitative dense video captioning results. Both the diversity score and the qualitative results confirm that our proposed approach could produce diversified captioning output.

Through the comparison with the oracle results and the diversity evaluation in this subsection, we have demonstrated that our method can indeed produce good dense captions.

5. Conclusion

We have presented a weakly supervised dense video captioning approach, which is able to generate multiple diversified captions for a video clip with only video-level sentence annotations during the training procedure. Experiments have demonstrated that our approach can produce multiple *informative* and *diversified* captions. Our best single caption output outperforms the state-of-the-art methods on the MSR-VTT challenge with a significant margin. Future work may consider leveraging the context among the dense captions to produce a consistent story for the input video clips.

Acknowledgements

Yu-Gang Jiang and Xiangyang Xue were supported in part by three NSFC projects (#U1611461, #61622204 and #61572138) and a grant from STCSM, Shanghai, China (#16JC1420401).

References

[1] L. Anne Hendricks, S. Venugopalan, and et al. Deep compositional captioning: Describing novel object categories without paired training data. In *CVPR*, 2016. 1, 3, 6

[2] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online multiple instance learning. In *CVPR*, 2009. 5

[3] P. Das, C. Xu, and et al. A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In *CVPR*, 2013. 2

[4] S. Deerwester, S. T. Dumais, and et al. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391, 1990. 8

[5] J. Donahue, L. Anne, and et al. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015. 1

[6] J. Dong, X. Li, and et al. Early embedding and late reranking for video captioning. In *ACM Multimedia Grand Challenge*, 2016. 7

[7] H. Fang, S. Gupta, and et al. From captions to visual concepts and back. In *CVPR*, 2015. 1, 3, 4, 6

[8] R. Girshick. Fast r-cnn. In *ICCV*, 2015. 4

[9] M. Gygli, H. Grabner, and L. Van Gool. Video summarization by learning submodular mixtures of objectives. In *CVPR*, 2015. 4

[10] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *ECCV*, 2014. 4

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4

[12] D. Heckerman. A tractable inference algorithm for diagnosing multiple diseases. *arXiv:1304.1511*, 2013. 3

[13] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8), 1997. 3, 5

[14] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi. Robust online appearance models for visual tracking. *IEEE T PAMI*, 2003. 5

[15] Q. Jin, J. Chen, and et al. Describing videos using multi-modal fusion. In *ACM Multimedia Grand Challenge*, 2016. 7

[16] J. Johnson, A. Karpathy, and L. Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. In *CVPR*, 2016. 1, 7

[17] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 1

[18] M. U. G. Khan, L. Zhang, and Y. Gotoh. Human focused video description. In *ICCV Workshops*, 2011. 2

[19] G. Kulkarni, V. Premraj, and et al. Babytalk: Understanding and generating simple image descriptions. *IEEE T PAMI*, 2013. 1, 3

[20] M. J. Kusner, Y. Sun, and et al. From word embeddings to document distances. In *ICML*, 2015. 8

[21] M. D. A. Lavie. Meteor universal: language specific translation evaluation for any target language. *ACL*, 2014. 6

[22] J. Leskovec, A. Krause, and et al. Cost-effective outbreak detection in networks. In *ACM SIGKDD*, 2007. 4

[23] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *ACL-04 workshop on Text summarization branches out:*, 2004. 6

[24] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 4

[25] L. Lovász. Submodular functions and convexity. In *Mathematical Programming The State of the Art*, pages 235–257. Springer, 1983. 5

[26] J. Mao, W. Xu, Y. Yang, and et al. Deep captioning with multimodal recurrent neural networks (m-rnn). *arXiv:1412.6632*, 2014. 1

[27] O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. In *NIPS*, 1998. 3

[28] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functionsi. *Mathematical Programming*, 14(1):265–294, 1978. 4, 5

[29] P. Pan, Z. Xu, Y. Yang, and et al. Hierarchical recurrent neural encoder for video representation with application to captioning. In *CVPR*, 2016. 1

[30] Y. Pan, T. Mei, T. Yao, and et al. Jointly modeling embedding and translation to bridge video and language. In *CVPR*, 2016. 1, 3

[31] S. Pancoast and M. Akbacak. Softening quantization in bag-of-audio-words. In *IEEE ICASSP*, 2014. 7

[32] K. Papineni, S. Roukos, T. Ward, and et al. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002. 6

[33] B. A. Plummer, L. Wang, C. M. Cervantes, and et al. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015. 1

[34] V. Ramanishka, A. Das, D. H. Park, and et al. Multimodal video description. In *ACM Multimedia Grand Challenge*, 2016. 7

[35] J. Redmon, S. Divvala, and et al. You only look once: Unified, real-time object detection. In *CVPR*, 2016. 4

[36] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 4

[37] A. Rohrbach, M. Rohrbach, and et al. Coherent multi-sentence video description with variable level of detail. In *GCPR*, 2014. 2

[38] A. Rohrbach, M. Rohrbach, and B. Schiele. The long-short story of movie description. In *GCPR*, 2015. 3

[39] O. Russakovsky, J. Deng, H. Su, J. Krause, and et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 2

[40] R. Shetty and J. Laaksonen. Frame-and segment-level features and candidate pool evaluation for video caption generation. *arXiv:1608.04959*, 2016. 7

[41] A. Shin, K. Ohnishi, and et al. Beyond caption to narrative: Video captioning with multiple sentences. *arXiv:1605.05440*, 2016. 2

[42] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014. 4

[43] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, 2014. 3

[44] K. Toutanova, D. Klein, and et al. Feature-rich part-of-speech tagging with a cyclic dependency network. In *NAACL*, 2003. 4

[45] D. Tran, L. Bourdev, and et al. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015. 3

[46] R. Vedantam, C. Lawrence Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, 2015. 6

[47] S. Venugopalan, L. A. Hendricks, and et al. Captioning images with diverse objects. *arXiv:1606.07770*, 2016. 1, 3

[48] S. Venugopalan, M. Rohrbach, and et al. Sequence to sequence-video to text. In *ICCV*, 2015. 1, 2, 3, 5, 7

[49] S. Venugopalan, H. Xu, and et al. Translating videos to natural language using deep recurrent neural networks. In *NAACL*, 2015. 1, 7

[50] O. Vinyals, A. Toshev, and et al. Show and tell: A neural image caption generator. In *CVPR*, 2015. 1

[51] J. Xu, T. Mei, T. Yao, and Y. Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, 2016. 3, 6

[52] K. Xu, J. Ba, and et al. Show, attend and tell: Neural image caption generation with visual attention. *arXiv:1502.03044*, 2(3):5, 2015. 1

[53] L. Yao, A. Torabi, K. Cho, and et al. Describing videos by exploiting temporal structure. In *ICCV*, 2015. 1, 3, 7

[54] H. Yu, J. Wang, and et al. Video paragraph captioning using hierarchical recurrent neural networks. In *CVPR*, 2016. 2

[55] W. Zaremba, I. Sutskever, and O. Vinyals. Recurrent neural network regularization. *arXiv:1409.2329*, 2014. 5

[56] C. Zhang, J. C. Platt, and P. A. Viola. Multiple instance boosting for object detection. In *NIPS*, 2005. 3

[57] Z.-H. Zhou and M.-L. Zhang. Multi-instance multi-label learning with application to scene classification. In *NIPS*, 2006. 3