Multi-Task Video Captioning with Video and Entailment Generation

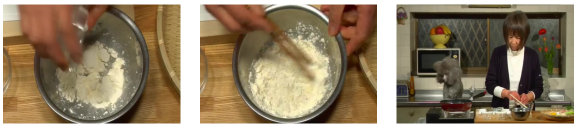
Ramakanth Pasunuru and **Mohit Bansal**
UNC Chapel Hill
{ram, mbansal}@cs.unc.edu

Abstract

Video captioning, the task of describing the content of a video, has seen some promising improvements in recent years with sequence-to-sequence models, but accurately learning the temporal and logical dynamics involved in the task still remains a challenge, especially given the lack of sufficient annotated data. We improve video captioning by sharing knowledge with two related directed-generation tasks: a temporally-directed unsupervised video prediction task to learn richer context-aware video encoder representations, and a logically-directed language entailment generation task to learn better video-entailing caption decoder representations. For this, we present a many-to-many multi-task learning model that shares parameters across the encoders and decoders of the three tasks. We achieve significant improvements and the new state-of-the-art on several standard video captioning datasets using diverse automatic and human evaluations. We also show mutual multi-task improvements on the entailment generation task.

1 Introduction

Video captioning is the task of automatically generating a natural language description of the content of a video, as shown in Fig. 1. It has various applications such as assistance to a visually impaired person and improving the quality of online video search or retrieval. This task has gained recent momentum in the natural language processing and computer vision communities, esp. with the advent of powerful image processing features as well as sequence-to-sequence LSTM models. It is also a step forward from static image captioning, because in addition to modeling the spatial visual features, the model also needs to learn the temporal across-frame action dynamics and the logical storyline language dynamics.



Ground truth: A person is mixing powdered ingradients with water.
A woman is mixing flour and water in a bowl.
Our model: A person is mixing ingredients in a bowl.

Figure 1: A video captioning example from the YouTube2Text dataset, with the ground truth captions and our many-to-many multi-task model's predicted caption.

Previous work in video captioning (Venugopalan et al., 2015a; Pan et al., 2016b) has shown that recurrent neural networks (RNNs) are a good choice for modeling the temporal information in the video. A sequence-to-sequence model is then used to 'translate' the video to a caption. Venugopalan et al. (2016) showed linguistic improvements over this by fusing the decoder with external language models. Furthermore, an attention mechanism between the video frames and the caption words captures some of the temporal matching relations better (Yao et al., 2015; Pan et al., 2016a). More recently, hierarchical two-level RNNs were proposed to allow for longer inputs and to model the full paragraph caption dynamics of long video clips (Pan et al., 2016a; Yu et al., 2016).

Despite these recent improvements, video captioning models still suffer from the lack of sufficient temporal and logical supervision to be able to correctly capture the action sequence and story-dynamic language in videos, esp. in the case of short clips. Hence, they would benefit from incorporating such complementary directed knowledge, both visual and textual. We address this by jointly training the task of video captioning with two related directed-generation tasks: a temporally-

Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pages 1273–1283
Vancouver, Canada, July 30 - August 4, 2017. ©2017 Association for Computational Linguistics
https://doi.org/10.18653/v1/P17-1117

directed unsupervised video prediction task and a logically-directed language entailment generation task. We model this via many-to-many multi-task learning based sequence-to-sequence models (Luong et al., 2016) that allow the sharing of parameters among the encoders and decoders across the three different tasks, with additional shareable attention mechanisms.

The unsupervised video prediction task, i.e., video-to-video generation (adapted from Srivastava et al. (2015)), shares its encoder with the video captioning task's encoder, and helps it learn richer video representations that can predict their temporal context and action sequence. The entailment generation task, i.e., premise-to-entailment generation (based on the image caption domain SNLI corpus (Bowman et al., 2015)), shares its decoder with the video captioning decoder, and helps it learn better video-entailing caption representations, since the caption is essentially an entailment of the video, i.e., it describes subsets of objects and events that are logically implied by or follow from the full video content). The overall many-to-many multi-task model combines all three tasks.

Our three novel multi-task models show statistically significant improvements over the state-of-the-art, and achieve the best-reported results (and rank) on multiple datasets, based on several automatic and human evaluations. We also demonstrate that video captioning, in turn, gives mutual improvements on the new multi-reference entailment generation task.

2 Related Work

Early video captioning work (Guadarrama et al., 2013; Thomason et al., 2014; Huang et al., 2013) used a two-stage pipeline to first extract a subject, verb, and object (S,V,O) triple and then generate a sentence based on it. Venugopalan et al. (2015b) fed mean-pooled static frame-level visual features (from convolution neural networks pre-trained on image recognition) of the video as input to the language decoder. To harness the important frame sequence temporal ordering, Venugopalan et al. (2015a) proposed a sequence-to-sequence model with video encoder and language decoder RNNs.

More recently, Venugopalan et al. (2016) explored linguistic improvements to the caption decoder by fusing it with external language models. Moreover, an attention or alignment mechanism was added between the encoder and the decoder to learn the temporal relations (matching) between the video frames and the caption words (Yao et al., 2015; Pan et al., 2016a). In contrast to static visual features, Yao et al. (2015) also considered temporal video features from a 3D-CNN model pretrained on an action recognition task.

To explore long range temporal relations, Pan et al. (2016a) proposed a two-level hierarchical RNN encoder which limits the length of input information and allows temporal transitions between segments. Yu et al. (2016)'s hierarchical RNN generates sentences at the first level and the second level captures inter-sentence dependencies in a paragraph. Pan et al. (2016b) proposed to simultaneously learn the RNN word probabilities and a visual-semantic joint embedding space that enforces the relationship between the semantics of the entire sentence and the visual content. Despite these useful recent improvements, video captioning still suffers from limited supervision and generalization capabilities, esp. given the complex action-based temporal and story-based logical dynamics that need to be captured from short video clips. Our work addresses this issue by bringing in complementary temporal and logical knowledge from video prediction and textual entailment generation tasks (respectively), and training them together via many-to-many multi-task learning.

Multi-task learning is a useful learning paradigm to improve the supervision and the generalization performance of a task by jointly training it with related tasks (Caruana, 1998; Argyriou et al., 2007; Kumar and Daumé III, 2012). Recently, Luong et al. (2016) combined multi-task learning with sequence-to-sequence models, sharing parameters across the tasks' encoders and decoders. They showed improvements on machine translation using parsing and image captioning. We additionally incorporate an attention mechanism to this many-to-many multi-task learning approach and improve the multimodal, temporal-logical video captioning task by sharing its video encoder with the encoder of a video-to-video prediction task and by sharing its caption decoder with the decoder of a linguistic premise-to-entailment generation task.

Image representation learning has been successful via supervision from very large object-labeled datasets. However, similar amounts of supervision are lacking for video representation learning. Srivastava et al. (2015) address this by propos-

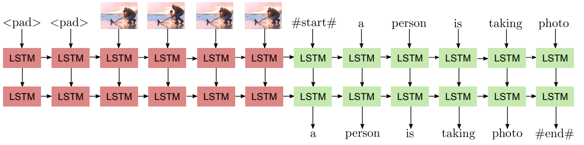


Figure 2: Baseline sequence-to-sequence model for video captioning: standard encoder-decoder LSTM-RNN model.

ing unsupervised video representation learning via sequence-to-sequence RNN models, where they reconstruct the input video sequence or predict the future sequence. We model video generation with an attention-enhanced encoder-decoder and harness it to improve video captioning.

The task of recognizing textual entailment (RTE) is to classify whether the relationship between a premise and hypothesis sentence is that of entailment (i.e., logically follows), contradiction, or independence (neutral), which is helpful for several downstream NLP tasks. The recent Stanford Natural Language Inference (SNLI) corpus by Bowman et al. (2015) allowed training end-to-end neural networks that outperform earlier feature-based RTE models (Lai and Hockenmaier, 2014; Jimenez et al., 2014). However, directly generating the entailed hypothesis sentences given a premise sentence would be even more beneficial than retrieving or reranking sentence pairs, because most downstream generation tasks only come with the source sentence and not pairs. Recently, Kolesnyk et al. (2016) tried a sequence-to-sequence model for this on the original SNLI dataset, which is a single-reference setting and hence restricts automatic evaluation. We modify the SNLI corpus to a new multi-reference (and a more challenging zero train-test premise overlap) setting, and present a novel multi-task training setup with the related video captioning task (where the caption also entails a video), showing mutual improvements on both the tasks.

3 Models

We first discuss a simple encoder-decoder model as a baseline reference for video captioning. Next, we improve this via an attention mechanism. Finally, we present similar models for the unsupervised video prediction and entailment generation tasks, and then combine them with video captioning via the many-to-many multi-task approach.

3.1 Baseline Sequence-to-Sequence Model

Our baseline model is similar to the standard machine translation encoder-decoder RNN model (Sutskever et al., 2014) where the final state of the encoder RNN is input as an initial state to the decoder RNN, as shown in Fig. 2. The RNN is based on Long Short Term Memory (LSTM) units, which are good at memorizing long sequences due to forget-style gates (Hochreiter and Schmidhuber, 1997). For video captioning, our input to the encoder is the video frame features[1] $\{f_1, f_2, ..., f_n\}$ of length n, and the caption word sequence $\{w_1, w_2, ..., w_m\}$ of length m is generated during the decoding phase. The distribution of the output sequence w.r.t. the input sequence is:

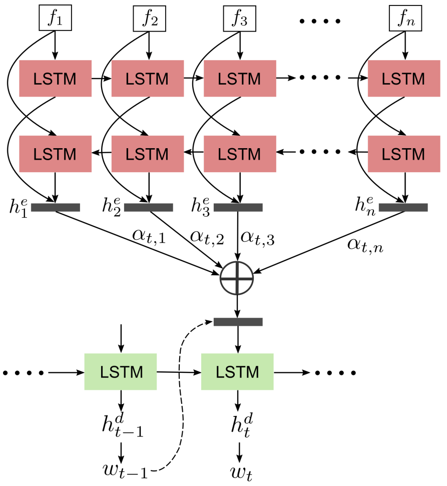


Figure 3: Attention-based sequence-to-sequence baseline model for video captioning (similar models also used for video prediction and entailment generation).

$$p(w_1, ..., w_m | f_1, ..., f_n) = \prod_{t=1}^{m} p(w_t | h_t^d) \qquad (1)$$

where h_t^d is the hidden state at the t^{th} time step of the decoder RNN, obtained from h_{t-1}^d and w_{t-1} via the standard LSTM-RNN equations. The distribution $p(w_t|h_t^d)$ is given by *softmax* over all the words in the vocabulary.

3.2 Attention-based Model

Our attention model architecture is similar to Bahdanau et al. (2015), with a bidirectional LSTM-RNN as the encoder and a unidirectional LSTM-RNN as the decoder, see Fig. 3. At each time step t, the decoder LSTM hidden state h_t^d is a non-linear recurrent function of the previous decoder hidden state h_{t-1}^d, the previous time-step's generated word w_{t-1}, and the context vector c_t:

$$h_t^d = S(h_{t-1}^d, w_{t-1}, c_t) \qquad (2)$$

[1] We use several popular image features such as VGGNet, GoogLeNet and Inception-v4. Details in Sec. 4.1.

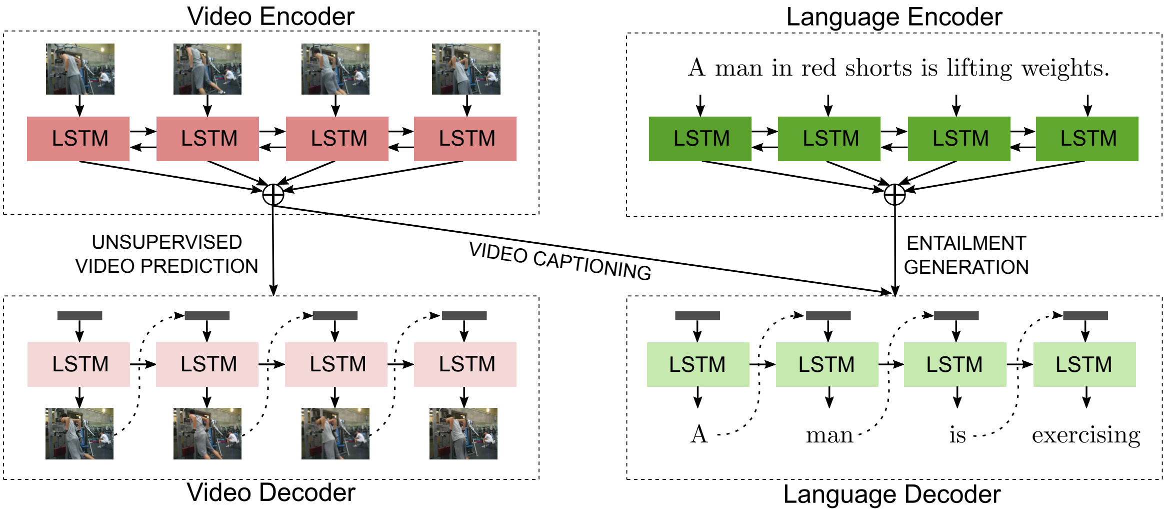


Figure 4: Our many-to-many multi-task learning model to share encoders and decoders of the video captioning, unsupervised video prediction, and entailment generation tasks.

where c_t is a weighted sum of encoder hidden states $\{h_i^e\}$:

$$c_t = \sum_{i=1}^{n} \alpha_{t,i} h_i^e \tag{3}$$

These attention weights $\{\alpha_{t,i}\}$ act as an alignment mechanism by giving higher weights to certain encoder hidden states which match that decoder time step better, and are computed as:

$$\alpha_{t,i} = \frac{exp(e_{t,i})}{\sum_{k=1}^{n} exp(e_{t,k})} \tag{4}$$

where the attention function $e_{t,i}$ is defined as:

$$e_{t,i} = w^T tanh(W_a^e h_i^e + W_a^d h_{t-1}^d + b_a) \tag{5}$$

where w, W_a^e, W_a^d, and b_a are learned parameters. This attention-based sequence-to-sequence model (Fig. 3) is our enhanced baseline for video captioning. We next discuss similar models for the new tasks of unsupervised video prediction and entailment generation and then finally share them via multi-task learning.

3.3 Unsupervised Video Prediction

We model unsupervised video representation by predicting the sequence of future video frames given the current frame sequence. Similar to Sec. 3.2, a bidirectional LSTM-RNN encoder and an LSTM-RNN decoder is used, along with attention. If the frame level features of a video of length n are $\{f_1, f_2, ..., f_n\}$, these are divided into two sets such that given the current frames $\{f_1, f_2, .., f_k\}$ (in its encoder), the model has to predict (decode) the rest of the frames $\{f_{k+1}, f_{k+2}, .., f_n\}$. The motivation is that this helps the video encoder learn rich temporal representations that are aware of their action-based context and are also robust to missing frames and varying frame lengths or motion speeds. The optimization function is defined as:

$$\underset{\phi}{\text{minimize}} \sum_{t=1}^{n-k} ||f_t^d - f_{t+k}||_2^2 \tag{6}$$

where ϕ are the model parameters, f_{t+k} is the true future frame feature at decoder time step t and f_t^d is the decoder's predicted future frame feature at decoder time step t, defined as:

$$f_t^d = S(h_{t-1}^d, f_{t-1}^d, c_t) \tag{7}$$

similar to Eqn. 2, with h_{t-1}^d and f_{t-1}^d as the previous time step's hidden state and predicted frame feature respectively, and c_t as the attention-weighted context vector.

3.4 Entailment Generation

Given a sentence (premise), the task of entailment generation is to generate a sentence (hypothesis) which is a logical deduction or implication of the premise. Our entailment generation model again uses a bidirectional LSTM-RNN encoder and LSTM-RNN decoder with an attention mechanism (similar to Sec. 3.2). If the premise s^p is a sequence of words $\{w_1^p, w_2^p, ..., w_n^p\}$ and the hypothesis s^h is $\{w_1^h, w_2^h, ..., w_m^h\}$, the distribution of the entailed hypothesis w.r.t. the premise is:

$$p(w_1^h, ..., w_m^h | w_1^p, ..., w_n^p) = \prod_{t=1}^{m} p(w_t^h | h_t^d) \tag{8}$$

where the distribution $p(w_t^h | h_t^d)$ is again obtained via softmax over all the words in the vocabulary and the decoder state h_t^d is similar to Eqn. 2.

3.5 Multi-Task Learning

Multi-task learning helps in sharing information between different tasks and across domains. Our primary aim is to improve the video captioning model, where visual content translates to a textual form in a directed (entailed) generation way. Hence, this presents an interesting opportunity to share temporally and logically directed knowledge with both visual and linguistic generation tasks. Fig. 4 shows our overall many-to-many multi-task model for jointly learning video captioning, unsupervised video prediction, and textual entailment generation. Here, the video captioning task shares its video encoder (parameters) with the encoder of the video prediction task (one-to-many setting) so as to learn context-aware and temporally-directed visual representations (see Sec. 3.3).

Moreover, the decoder of the video captioning task is shared with the decoder of the textual entailment generation task (many-to-one setting), thus helping generate captions that can 'entail', i.e., are logically implied by or follow from the video content (see Sec. 3.4).[2] In both the one-to-many and the many-to-one settings, we also allow the attention parameters to be shared or separated. The overall many-to-many setting thus improves both the visual and language representations of the video captioning model.

We train the multi-task model by alternately optimizing each task in mini-batches based on a mixing ratio. Let α_v, α_f, and α_e be the number of mini-batches optimized alternately from each of these three tasks – video captioning, unsupervised video future frames prediction, and entailment generation, resp. Then the mixing ratio is defined as $\frac{\alpha_v}{(\alpha_v+\alpha_f+\alpha_e)} : \frac{\alpha_f}{(\alpha_v+\alpha_f+\alpha_e)} : \frac{\alpha_e}{(\alpha_v+\alpha_f+\alpha_e)}$.

4 Experimental Setup

4.1 Datasets

Video Captioning Datasets We report results on three popular video captioning datasets. First, we use the YouTube2Text or MSVD (Chen and Dolan, 2011) for our primary results, which contains 1970 YouTube videos in the wild with several different reference captions per video (40 on average). We also use MSR-VTT (Xu et al., 2016) with $10,000$ diverse video clips (from a video search engine) – it has $200,000$ video clip-sentence pairs and around 20 captions per video; and M-VAD (Torabi et al., 2015) with $49,000$ movie-based video clips but only 1 or 2 captions per video, making most evaluation metrics (except paraphrase-based METEOR) infeasible. We use the standard splits for all three datasets. Further details about all these datasets are provided in the supplementary.

Video Prediction Dataset For our unsupervised video representation learning task, we use the UCF-101 action videos dataset (Soomro et al., 2012), which contains $13,320$ video clips of 101 action categories, and suits our video captioning task well because it also contains short video clips of a single action or few actions. We use the standard splits – further details in supplementary.

Entailment Generation Dataset For the entailment generation encoder-decoder model, we use the Stanford Natural Language Inference (SNLI) corpus (Bowman et al., 2015), which contains human-annotated English sentence pairs with classification labels of entailment, contradiction and neutral. It has a total of $570,152$ sentence pairs out of which $190,113$ correspond to true entailment pairs, and we use this subset in our multi-task video captioning model. For improving video captioning, we use the same training/validation/test splits as provided by Bowman et al. (2015), which is $183,416$ training, $3,329$ validation, and $3,368$ testing pairs (for the entailment subset).

However, for the entailment generation multi-task results (see results in Sec. 5.3), we modify the splits so as to create a multi-reference setup which can afford evaluation with automatic metrics. A given premise usually has multiple entailed hypotheses but the original SNLI corpus is set up as single-reference (for classification). Due to this, the different entailed hypotheses of the same premise land up in different splits of the dataset (e.g., one in train and one in test/validation) in many cases. Therefore, we regroup the premise-entailment pairs and modify the split as follows: among the $190,113$ premise-entailment pairs subset of the SNLI corpus, there are $155,898$ unique premises; out of which $145,822$ have only one hy-

[2] Empirically, logical entailment helped captioning more than simple fusion with language modeling (i.e., partial sentence completion with no logical implication), because a caption also entails a video in a logically-directed sense and hence the entailment generation task matches the video captioning task better than language modeling. Moreover, a multi-task setup is more suitable to add directed information such as entailment (as opposed to pretraining or fusion with only the decoder). Details in Sec. 5.1.

pothesis and we make this the training set, and the rest of them ($10,076$) have more than one hypothesis, which we randomly shuffle and divide equally into test and validation sets, so that each of these two sets has approximately the same distribution of the number of reference hypotheses per premise.

These new validation and test sets hence contain premises with multiple entailed hypotheses as ground truth references, thus allowing for automatic metric evaluation, where differing generations still get positive scores by matching one of the multiple references. Also, this creates a more challenging dataset for entailment generation because of zero premise overlap between the training and val/test sets. We will make these split details publicly available.

Pre-trained Visual Frame Features For the three video captioning and UCF-101 datasets, we fix our sampling rate to $3fps$ to bring uniformity in the temporal representation of actions across all videos. These sampled frames are then converted into features using several state-of-the-art pre-trained models on ImageNet (Deng et al., 2009) – VGGNet (Simonyan and Zisserman, 2015), GoogLeNet (Szegedy et al., 2015; Ioffe and Szegedy, 2015), and Inception-v4 (Szegedy et al., 2016). Details of these feature dimensions and layer positions are in the supplementary.

4.2 Evaluation (Automatic and Human)

For our video captioning as well as entailment generation results, we use four diverse automatic evaluation metrics that are popular for image/video captioning and language generation in general: METEOR (Denkowski and Lavie, 2014), BLEU-4 (Papineni et al., 2002), CIDEr-D (Vedantam et al., 2015), and ROUGE-L (Lin, 2004). Particularly, METEOR and CIDEr-D have been justified to be better for generation tasks, because CIDEr-D uses consensus among the (large) number of references and METEOR uses soft matching based on stemming, paraphrasing, and WordNet synonyms. We use the standard evaluation code from the Microsoft COCO server (Chen et al., 2015) to obtain these results and also to compare the results with previous papers.[3]

We also present human evaluation results based on relevance (i.e., how related is the generated caption w.r.t. the video contents such as actions, objects, and events; or is the generated hypothesis entailed or implied by the premise) and coherence (i.e., a score on the logic, readability, and fluency of the generated sentence).

4.3 Training Details

We tune all hyperparameters on the dev splits: LSTM-RNN hidden state size, learning rate, weight initializations, and mini-batch mixing ratios (tuning ranges in supplementary). We use the following settings in all of our models (unless otherwise specified): we unroll video encoder/decoder RNNs to 50 time steps and language encoder/decoder RNNs to 30 time steps. We use a 1024-dimension RNN hidden state size and 512-dim vectors to embed visual features and word vectors. We use Adam optimizer (Kingma and Ba, 2015). We apply a dropout of 0.5. See subsections below and supp for full details.

5 Results and Analysis

5.1 Video Captioning on YouTube2Text

Table 1 presents our primary results on the YouTube2Text (MSVD) dataset, reporting several previous works, all our baselines and attention model ablations, and our three multi-task models, using the four automated evaluation metrics. For each subsection below, we have reported the important training details inline, and refer to the supplementary for full details (e.g., learning rates and initialization).

Baseline Performance We first present all our baseline model choices (ablations) in Table 1. Our baselines represent the standard sequence-to-sequence model with three different visual feature types as well as those with attention mechanisms. Each baseline model is trained with three random seed initializations and the average is reported (for stable results). The final baseline model $\otimes$ instead uses an ensemble (E), which is a standard denoising method (Sutskever et al., 2014) that performs inference over ten randomly initialized models, i.e., at each time step t of the decoder, we generate a word based on the avg. of the likelihood probabilities from the ten models. Moreover, we use beam search with size 5 for all baseline models. Overall, the final baseline model with Inception-v4 features, attention, and 10-ensemble performs

[3] We use avg. of these four metrics on validation set to choose the best model, except for single-reference M-VAD dataset where we only report and choose based on METEOR.

Models	METEOR	CIDEr-D	ROUGE-L	BLEU-4
PREVIOUS WORK				
LSTM-YT (V) (Venugopalan et al., 2015b)	26.9	-	-	31.2
S2VT (V + A) (Venugopalan et al., 2015a)	29.8	-	-	-
Temporal Attention (G + C) (Yao et al., 2015)	29.6	51.7	-	41.9
LSTM-E (V + C) (Pan et al., 2016b)	31.0	-	-	45.3
Glove + DeepFusion (V) (E) (Venugopalan et al., 2016)	31.4	-	-	42.1
p-RNN (V + C) (Yu et al., 2016)	32.6	65.8	-	49.9
HNRE + Attention (G + C) (Pan et al., 2016a)	33.9	-	-	46.7
OUR BASELINES				
Baseline (V)	31.4	63.9	68.0	43.6
Baseline (G)	31.7	64.8	68.6	44.1
Baseline (I)	33.3	75.6	69.7	46.3
Baseline + Attention (V)	32.6	72.2	69.0	47.5
Baseline + Attention (G)	33.0	69.4	68.3	44.9
Baseline + Attention (I)	33.8	77.2	70.3	49.9
Baseline + Attention (I) (E) $\otimes$	35.0	84.4	71.5	52.6
OUR MULTI-TASK LEARNING MODELS				
$\otimes$ + Video Prediction (1-to-M)	35.6	88.1	72.9	54.1
$\otimes$ + Entailment Generation (M-to-1)	35.9	88.0	72.7	54.4
$\otimes$ + Video Prediction + Entailment Generation (M-to-M)	**36.0**	**92.4**	**72.8**	**54.5**

Table 1: Primary video captioning results on Youtube2Text (MSVD), showing previous works, our several strong baselines, and our three multi-task models. Here, V, G, I, C, A are short for VGGNet, GoogLeNet, Inception-v4, C3D, and AlexNet visual features; E = ensemble. The multi-task models are applied on top of our best video captioning baseline $\otimes$, with an ensemble. All the multi-task models are statistically significant over the baseline (discussed inline in the corresponding results sections).

well (and is better than all previous state-of-the-art), and so we next add all our novel multi-task models on top of this final baseline.

Multi-Task with Video Prediction (1-to-M) Here, the video captioning and unsupervised video prediction tasks share their encoder LSTM-RNN weights and image embeddings in a one-to-many multi-task setting. Two important hyperparameters tuned (on the validation set of captioning datasets) are the ratio of encoder vs decoder frames for video prediction on UCF-101 (where we found that 80% of frames as input and 20% for prediction performs best); and the mini-batch mixing ratio between the captioning and video prediction tasks (where we found $100 : 200$ works well). Table 1 shows a statistically significant improvement[4] in all metrics in comparison to the best baseline (non-multitask) model as well as w.r.t. all previous works, demonstrating the effectiveness of multi-task learning for video captioning with video prediction, even with unsupervised signals.

Multi-Task with Entailment Generation (M-to-1) Here, the video captioning and entailment generation tasks share their language decoder LSTM-RNN weights and word embeddings in a many-to-one multi-task setting. We observe that a mixing ratio of $100 : 50$ alternating mini-batches (between the captioning and entailment tasks) works well here. Again, Table 1 shows statistically significant improvements[5] in all the metrics in comparison to the best baseline model (and all previous works) under this multi-task setting. Note that in our initial experiments, our entailment generation model helped the video captioning task significantly more than the alternative approach of simply improving fluency by adding (or deep-fusing) an external language model (or pre-trained word embeddings) to the decoder (using both in-domain and out-of-domain language models), again because a caption also 'entails' a video in a logically-directed sense and hence this matches our captioning task better (also see results of Venugopalan et al. (2016) in Table 1).

Multi-Task with Video and Entailment Generation (M-to-M) Combining the above one-to-many and many-to-one multi-task learning models, our full model is the 3-task, many-to-many model (Fig. 4) where both the video encoder and the language decoder of the video captioning model are shared (and hence improved) with that of the unsupervised video prediction and entailment generation models, respectively.[6] A mixing ratio of $100 : 100 : 50$ alternate mini-batches

[4] Statistical significance of $p < 0.01$ for CIDEr-D and ROUGE-L, $p < 0.02$ for BLEU-4, $p < 0.03$ for METEOR, based on the bootstrap test (Noreen, 1989; Efron and Tibshirani, 1994) with 100K samples.

[5] Statistical significance of $p < 0.01$ for all four metrics.

[6] We found the setting with unshared attention parameters to work best, likely because video captioning and video prediction prefer very different alignment distributions.

Models	M	C	R	B
Venugopalan (2015b)*	23.4	-	-	32.3
Yao et al. (2015)*	25.2	-	-	35.2
Xu et al. (2016)	25.9	-	-	36.6
Rank1: v2t_navigator	28.2	44.8	**60.9**	**40.8**
Rank2: Aalto	26.9	45.7	59.8	39.8
Rank3: VideoLAB	27.7	44.1	60.6	39.1
Our Model (**New Rank1**)	**28.8**	**47.1**	60.2	**40.8**

Table 2: Results on MSR-VTT dataset on the 4 metrics. *Results are reimplementations as per Xu et al. (2016). We also report the top 3 leaderboard systems – our model achieves the new rank 1 based on their ranking method.

Models	METEOR
Yao et al. (2015)	5.7
Venugopalan et al. (2015a)	6.7
Pan et al. (2016a)	6.8
Our M-to-M Multi-Task Model	**7.4**

Table 3: Results on M-VAD dataset.

of video captioning, unsupervised video prediction, and entailment generation, resp. works well. Table 1 shows that our many-to-many multi-task model again outperforms our strongest baseline (with statistical significance of $p < 0.01$ on all metrics), as well as all the previous state-of-the-art results by large absolute margins on all metrics. It also achieves significant improvements on some metrics over the one-to-many and many-to-one models.[7] Overall, we achieve the best results to date on YouTube2Text (MSVD) on all metrics.

5.2 Video Captioning on MSR-VTT, M-VAD

In Table 2, we also train and evaluate our final many-to-many multi-task model on two other video captioning datasets (using their standard splits; details in supplementary). First, we evaluate on the new MSR-VTT dataset (Xu et al., 2016). Since this is a recent dataset, we list previous works' results as reported by the MSR-VTT dataset paper itself.[8] We improve over all of these significantly. Moreover, they maintain a leaderboard[9] on this dataset and we also report the top 3 systems from it. Based on their ranking method, our multi-task model achieves the new rank 1 on this leaderboard. In Table 3, we further evaluate our model on the challenging movie-based M-VAD dataset, and again achieve improvements over all previous work (Venugopalan et al., 2015a; Pan et al., 2016a; Yao et al., 2015).[10]

Models	M	C	R	B
Entailment Generation	28.0	108.4	59.7	36.6
+Video Caption (M-to-1)	**28.7**	**114.5**	**60.8**	**38.9**

Table 4: Entailment generation results with the four metrics.

5.3 Entailment Generation Results

Above, we showed that the new entailment generation task helps improve video captioning. Next, we show that the video captioning task also inversely helps the entailment generation task. Given a premise, the task of entailment generation is to generate an entailed hypothesis. We use only the entailment pairs subset of the SNLI corpus for this, but with a multi-reference split setup to allow automatic metric evaluation and a zero train-test premise overlap (see Sec. 4.1). All the hyperparameter details (again tuned on the validation set) are presented in the supplementary. Table 4 presents the entailment generation results for the baseline (sequence-to-sequence with attention, 3-ensemble, beam search) and the multi-task model which uses video captioning (shared decoder) on top of the baseline. A mixing ratio of 100 : 20 alternate mini-batches of entailment generation and video captioning (resp.) works well.[11] The multi-task model achieves stat. significant ($p < 0.01$) improvements over the baseline on all metrics, thus demonstrating that video captioning and entailment generation both mutually help each other.

5.4 Human Evaluation

In addition to the automated evaluation metrics, we present pilot-scale human evaluations on the YouTube2Text (Table 1) and entailment generation (Table 4) results. In each case, we compare our strongest baseline with our final multi-task model by taking a random sample of 200 generated captions (or entailed hypotheses) from the test set and removing the model identity to anonymize the two models, and ask the human evaluator to choose the better model based on *relevance* and *coherence* (described in Sec. 4.2). As shown in Table 5, the multi-task models are always better than the strongest baseline for both video captioning and entailment generation, on both relevance

[7] Many-to-many model's improvements have a statistical significance of $p < 0.01$ on all metrics w.r.t. baseline, and $p < 0.01$ on CIDEr-D w.r.t. both one-to-many and many-to-one models, and $p < 0.04$ on METEOR w.r.t. one-to-many.

[8] In their updated supplementary at `https://www.microsoft.com/en-us/research/wp-content/uploads/2016/10/cvpr16.supplementary.pdf`

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

[10] Following previous work, we only use METEOR because M-VAD only has a single reference caption per video.

[11] Note that this many-to-one model prefers a different mixing ratio and learning rate than the many-to-one model for improving video captioning (Sec. 5.1), because these hyperparameters depend on the primary task being improved, as also discussed in previous work (Luong et al., 2016).

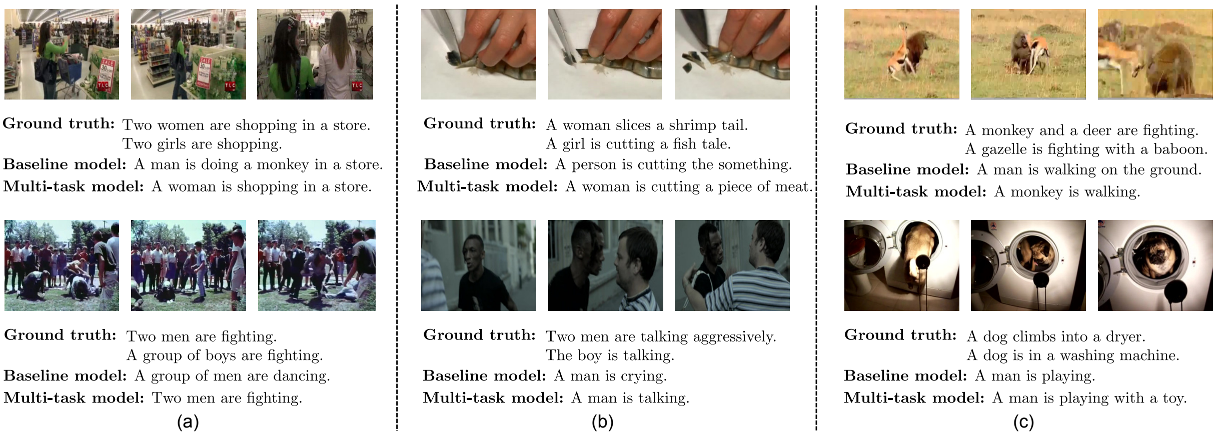

Ground truth: Two women are shopping in a store.
Two girls are shopping.
Baseline model: A man is doing a monkey in a store.
Multi-task model: A woman is shopping in a store.
Ground truth: Two men are fighting.
A group of boys are fighting.
Baseline model: A group of men are dancing.
Multi-task model: Two men are fighting.
(a)
Ground truth: A woman slices a shrimp tail.
A girl is cutting a fish tale.
Baseline model: A person is cutting the something.
Multi-task model: A woman is cutting a piece of meat.
Ground truth: Two men are talking aggressively.
The boy is talking.
Baseline model: A man is crying.
Multi-task model: A man is talking.
(b)
Ground truth: A monkey and a deer are fighting.
A gazelle is fighting with a baboon.
Baseline model: A man is walking on the ground.
Multi-task model: A monkey is walking.
Ground truth: A dog climbs into a dryer.
A dog is in a washing machine.
Baseline model: A man is playing.
Multi-task model: A man is playing with a toy.
(c)

Figure 5: Examples of generated video captions on the YouTube2Text dataset: (a) complex examples where the multi-task model performs better than the baseline; (b) ambiguous examples (i.e., ground truth itself confusing) where multi-task model still correctly predicts one of the possible categories (c) complex examples where both models perform poorly.

	YouTube2Text		Entailment	
	Relev.	Coher.	Relev.	Coher.
Not Distinguish.	65.0%	93.0%	73.5%	94.5%
Baseline Wins	14.0%	1.0%	12.5%	1.5%
Multi-Task Wins	**21.0%**	**6.0%**	**15.0%**	**4.0%**

Table 5: Human evaluation on captioning and entailment.

Given Premise	Generated Entailment
a man on stilts is playing a tuba for money on the boardwalk	a man is playing an instrument
a girl looking through a large telescope on a school trip	a girl is looking at something
several young people sit at a table playing poker	people are playing a game
the stop sign is folded up against the side of the bus	the sign is not moving
a blue and silver monster truck making a huge jump over crushed cars	a truck is being driven

Table 6: Examples of our multi-task model's generated entailment hypotheses given a premise.

and coherence, and with similar improvements (2-7%) as the automatic metrics (shown in Table 1).

5.5 Analysis

Fig. 5 shows video captioning generation results on the YouTube2Text dataset where our final M-to-M multi-task model is compared with our strongest attention-based baseline model for three categories of videos: (a) complex examples where the multi-task model performs better than the baseline; (b) ambiguous examples (i.e., ground truth itself confusing) where multi-task model still correctly predicts one of the possible categories (c) complex examples where both models perform poorly. Overall, we find that the multi-task model generates captions that are better at both temporal action prediction and logical entailment (i.e., correct subset of full video premise) w.r.t. the ground truth captions. The supplementary also provides ablation examples of improvements by the 1-to-M video prediction based multi-task model alone, as well as by the M-to-1 entailment based multi-task model alone (over the baseline).

On analyzing the cases where the baseline is better than the final M-to-M multi-task model, we find that these are often scenarios where the multi-task model's caption is also correct but the baseline caption is a bit more specific, e.g., "a man is holding a gun" vs "a man is shooting a gun".

Finally, Table 6 presents output examples of our entailment generation multi-task model (Sec. 5.3), showing how the model accurately learns to produce logically implied subsets of the premise.

6 Conclusion

We presented a multimodal, multi-task learning approach to improve video captioning by incorporating temporally and logically directed knowledge via video prediction and entailment generation tasks. We achieve the best reported results (and rank) on three datasets, based on multiple automatic and human evaluations. We also show mutual multi-task improvements on the new entailment generation task. In future work, we are applying our entailment-based multi-task paradigm to other directed language generation tasks such as image captioning and document summarization.

Acknowledgments

We thank the anonymous reviewers for their helpful comments. This work was partially supported by a Google Faculty Research Award, an IBM Faculty Award, a Bloomberg Data Science Research Grant, and NVidia GPU awards.

References

Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. 2007. Multi-task feature learning. In *NIPS*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *EMNLP*.

Rich Caruana. 1998. Multitask learning. In *Learning to learn*, Springer, pages 95–133.

David L Chen and William B Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, pages 190–200.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft COCO captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325* .

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *CVPR*. IEEE, pages 248–255.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *EACL*.

Bradley Efron and Robert J Tibshirani. 1994. *An introduction to the bootstrap*. CRC press.

Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2013. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *CVPR*. pages 2712–2719.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Haiqi Huang, Yueming Lu, Fangwei Zhang, and Songlin Sun. 2013. A multi-modal clustering method for web videos. In *International Conference on Trustworthy Computing and Services*. pages 163–169.

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*.

Sergio Jimenez, George Duenas, Julia Baquero, Alexander Gelbukh, Av Juan Dios Bátiz, and Av Mendizábal. 2014. UNAL-NLP: Combining soft cardinality features for semantic textual similarity, relatedness and entailment. In *In SemEval*. pages 732–742.

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

Vladyslav Kolesnyk, Tim Rocktäschel, and Sebastian Riedel. 2016. Generating natural language inference chains. *arXiv preprint arXiv:1606.01404* .

Abhishek Kumar and Hal Daumé III. 2012. Learning task grouping and overlap in multi-task learning. In *ICML*.

Alice Lai and Julia Hockenmaier. 2014. Illinois-LH: A denotational and distributional approach to semantics. *Proc. SemEval* 2:5.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 workshop*. volume 8.

Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task sequence to sequence learning. In *ICLR*.

Eric W Noreen. 1989. *Computer-intensive methods for testing hypotheses*. Wiley New York.

Pingbo Pan, Zhongwen Xu, Yi Yang, Fei Wu, and Yueting Zhuang. 2016a. Hierarchical recurrent neural encoder for video representation with application to captioning. In *CVPR*. pages 1029–1038.

Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. 2016b. Jointly modeling embedding and translation to bridge video and language. In *CVPR*. pages 4594–4602.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*. pages 311–318.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *ICLR*.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402* .

Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. 2015. Unsupervised learning of video representations using lstms. In *ICML*. pages 843–852.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*. pages 3104–3112.

Christian Szegedy, Sergey Ioffe, and Vincent Vanhoucke. 2016. Inception-v4, inception-resnet and the impact of residual connections on learning. In *CoRR*.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *CVPR*. pages 1–9.

Jesse Thomason, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Raymond J Mooney. 2014. Integrating language and vision to generate natural language descriptions of videos in the wild. In *COLING*.

Atousa Torabi, Christopher Pal, Hugo Larochelle, and Aaron Courville. 2015. Using descriptive video services to create a large data source for video annotation research. *arXiv preprint arXiv:1503.01070* .

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. CIDEr: Consensus-based image description evaluation. In *CVPR*. pages 4566–4575.

Subhashini Venugopalan, Lisa Anne Hendricks, Raymond Mooney, and Kate Saenko. 2016. Improving lstm-based video description with linguistic knowledge mined from text. In *EMNLP*.

Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2015a. Sequence to sequence-video to text. In *CVPR*. pages 4534–4542.

Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko. 2015b. Translating videos to natural language using deep recurrent neural networks. In *NAACL HLT*.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*. pages 5288–5296.

Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. 2015. Describing videos by exploiting temporal structure. In *CVPR*. pages 4507–4515.

Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. 2016. Video paragraph captioning using hierarchical recurrent neural networks. In *CVPR*.