Improving LSTM-based Video Description with Linguistic Knowledge Mined from Text

Subhashini Venugopalan
UT Austin
vsub@cs.utexas.edu

Lisa Anne Hendricks
UC Berkeley
lisa_anne@berkeley.edu

Raymond Mooney
UT Austin
mooney@cs.utexas.edu

Kate Saenko
Boston University
saenko@bu.edu

Abstract

This paper investigates how linguistic knowledge mined from large text corpora can aid the generation of natural language descriptions of videos. Specifically, we integrate both a neural language model and distributional semantics trained on large text corpora into a recent LSTM-based architecture for video description. We evaluate our approach on a collection of Youtube videos as well as two large movie description datasets showing significant improvements in grammaticality while modestly improving descriptive quality.

1 Introduction

The ability to automatically describe videos in natural language (NL) enables many important applications including content-based video retrieval and video description for the visually impaired. The most effective recent methods (Venugopalan et al., 2015a; Yao et al., 2015) use recurrent neural networks (RNN) and treat the problem as machine translation (MT) from video to natural language. Deep learning methods such as RNNs need large training corpora; however, there is a lack of high-quality paired video-sentence data. In contrast, raw text corpora are widely available and exhibit rich linguistic structure that can aid video description. Most work in statistical MT utilizes both a language model trained on a large corpus of monolingual target language data as well as a translation model trained on more limited parallel bilingual data. This paper explores methods to incorporate knowledge from language corpora to capture general linguistic regularities to aid video description.

This paper integrates linguistic information into a video-captioning model based on Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) RNNs which have shown state-of-the-art performance on the task. Further, LSTMs are also effective as language models (LMs) (Sundermeyer et al., 2010). Our first approach (early fusion) is to pre-train the network on plain text before training on parallel video-text corpora. Our next two approaches, inspired by recent MT work (Gulcehre et al., 2015), integrate an LSTM LM with the existing video-to-text model. Furthermore, we also explore replacing the standard one-hot word encoding with distributional vectors trained on external corpora.

We present detailed comparisons between the approaches, evaluating them on a standard Youtube corpus and two recent large movie description datasets. The results demonstrate significant improvements in grammaticality of the descriptions (as determined by crowdsourced human evaluations) and more modest improvements in descriptive quality (as determined by both crowdsourced human judgements and standard automated comparison to human-generated descriptions). Our main contributions are 1) multiple ways to incorporate knowledge from external text into an existing captioning model, 2) extensive experiments comparing the methods on three large video-caption datasets, and 3) human judgements to show that external linguistic knowledge has a significant impact on grammar.

2 LSTM-based Video Description

We use the successful S2VT video description framework from Venugopalan et al. (2015a) as our

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1961–1966,
Austin, Texas, November 1-5, 2016. ©2016 Association for Computational Linguistics

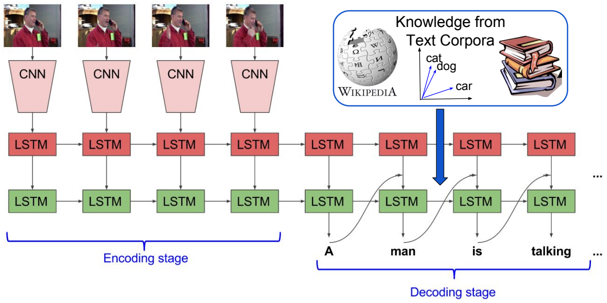


Figure 1: The S2VT architecture encodes a sequence of frames and decodes them to a sentence. We propose to add knowledge from text corpora to enhance the quality of video description.

underlying model and describe it briefly here. S2VT uses a sequence to sequence approach (Sutskever et al., 2014; Cho et al., 2014) that maps an input $\vec{x} = (x_1, \ldots, x_T)$ video frame feature sequence to a fixed dimensional vector and then decodes this into a sequence of output words $\vec{y} = (y_1, \ldots, y_N)$.

As shown in Fig. 1, it employs a stack of two LSTM layers. The input $\vec{x}$ to the first LSTM layer is a sequence of frame features obtained from the penultimate layer (fc_7) of a Convolutional Neural Network (CNN) after the ReLu operation. This LSTM layer encodes the video sequence. At each time step, the hidden control state h_t is provided as input to a second LSTM layer. After viewing all the frames, the second LSTM layer learns to decode this state into a sequence of words. This can be viewed as using one LSTM layer to model the visual features, and a second LSTM layer to model language conditioned on the visual representation. We modify this architecture to incorporate linguistic knowledge at different stages of the training and generation process. Although our methods use S2VT, they are sufficiently general and could be incorporated into other CNN-RNN based captioning models.

3 Approach

Existing visual captioning models (Vinyals et al., 2015; Donahue et al., 2015) are trained solely on text from the caption datasets and tend to exhibit some linguistic irregularities associated with a restricted language model and a small vocabulary. Here, we investigate several techniques to integrate prior linguistic knowledge into a CNN/LSTM-based network for video to text (S2VT) and evaluate their effectiveness at improving the overall description.

Early Fusion. Our first approach (*early fusion*), is to pre-train portions of the network modeling language on large corpora of raw NL text and then continue "fine-tuning" the parameters on the paired video-text corpus. An LSTM model learns to estimate the probability of an output sequence given an input sequence. To learn a language model, we train the LSTM layer to predict the next word given the previous words. Following the S2VT architecture, we embed one-hot encoded words in lower dimensional vectors. The network is trained on web-scale text corpora and the parameters are learned through backpropagation using stochastic gradient descent.[1] The weights from this network are then used to *initialize* the embedding and weights of the LSTM layers of S2VT, which is then trained on video-text data. This trained LM is also used as the LSTM LM in the late and deep fusion models.

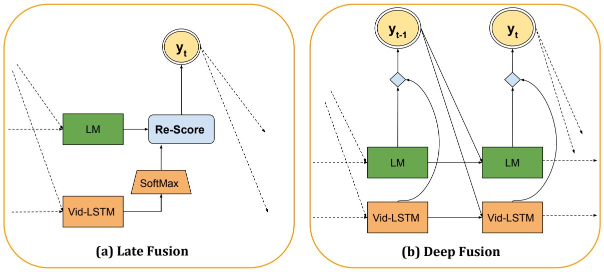
Late Fusion. Our late fusion approach is similar to how neural machine translation models incorporate a trained language model during decoding. At each step of sentence generation, the video caption model proposes a distribution over the vocabulary. We then use the language model to re-score the final output by considering the weighted average of the sum of scores proposed by the LM as well as the S2VT video-description model (VM). More specifically, if y_t denotes the output at time step t, and if p_{VM} and p_{LM} denote the proposal distributions of the video captioning model, and the language models respectively, then for all words $y' \in V$ in the vocabulary we can recompute the score of each new word, $p(y_t = y')$ as:

$$\alpha \cdot p_{VM}(y_t = y') + (1 - \alpha) \cdot p_{LM}(y_t = y') \quad (1)$$

Hyper-parameter α is tuned on the validation set.

Deep Fusion. In the deep fusion approach (Fig. 2), we integrate the LM a step deeper in the generation process by concatenating the hidden state of the language model LSTM (h_t^{LM}) with the hidden state of the S2VT video description model (h_t^{VM}) and use the combined latent vector to predict the output word. This is similar to the technique proposed by Gulcehre et al. (2015) for incorporating language models trained on monolingual corpora for machine translation. However, our approach differs in two

[1] The LM was trained to achieve a perplexity of 120



(a) Late Fusion

(b) Deep Fusion

Figure 2: Illustration of our late and deep fusion approaches to integrate an independently trained LM to aid video captioning. The deep fusion model learns jointly from the hidden representations of the LM and S2VT video-to-text model (Vid-LSTM), whereas the late fusion re-scores the softmax output of the video-to-text model.

key ways: (1) we only concatenate the hidden states of the S2VT LSTM and language LSTM and do not use any additional context information, (2) we fix the weights of the LSTM language model but train the full video captioning network. In this case, the probability of the predicted word at time step t is:

$$p(y_t|\vec{y}_{<t}, \vec{x}) \propto \exp(\mathrm{Wf}(h_t^{VM}, h_t^{LM}) + b) \qquad (2)$$

where $\vec{x}$ is the visual feature input, W is the weight matrix, and b the biases. We avoid tuning the LSTM LM to prevent overwriting already learned weights of a strong language model. But we train the full video caption model to incorporate the LM outputs while training on the caption domain.

Distributional Word Representations. The S2VT network, like most image and video captioning models, represents words using a 1-of-N (one hot) encoding. During training, the model learns to embed "one-hot" words into a lower 500d space by applying a linear transformation. However, the embedding is learned only from the limited and possibly noisy text in the caption data. There are many approaches (Mikolov et al., 2013; Pennington et al., 2014) that use large text corpora to learn vector-space representations of words that capture fine-grained semantic and syntactic regularities. We propose to take advantage of these to aid video description. Specifically, we replace the embedding matrix from one-hot vectors and instead use 300-dimensional GloVe vectors (Pennington et al., 2014) pre-trained on 6B tokens from Gigaword and Wikipedia 2014. In addition to using the distributional vectors for the input, we also explore variations where the model predicts both the one-hot word (trained on the $\mathrm{softmax}$ loss), as well as predicting the distributional vector from the LSTM hidden state using Euclidean loss as the objective. Here the output vector (y_t) is computed as $y_t = (W_g h_t + b_g)$, and the loss is given by:

$$\mathbb{L}(y_t, w_{glove}) = \|(W_g h_t + b_g) - w_{glove}\|^2 \qquad (3)$$

where h_t is the LSTM output, w_{glove} is the word's GloVe embedding and W, b are weights and biases. The network then essentially becomes a multi-task model with two loss functions. However, we use this loss only to influence the weights learned by the network, the predicted word embedding is not used.

Ensembling. The overall loss function of the video-caption network is non-convex, and difficult to optimize. In practice, using an ensemble of networks trained slightly differently can improve performance (Hansen and Salamon, 1990). In our work we also present results of an ensemble by averaging the predictions of the best performing models.

4 Experiments

Datasets. Our language model was trained on sentences from Gigaword, BNC, UkWaC, and Wikipedia. The vocabulary consisted of 72,700 most frequent tokens also containing GloVe embeddings. Following the evaluation in Venugopalan et al. (2015a), we compare our models on the Youtube dataset (Chen and Dolan, 2011), as well as two large movie description corpora: MPII-MD (Rohrbach et al., 2015) and M-VAD (Torabi et al., 2015).

Evaluation Metrics. We evaluate performance using machine translation (MT) metrics METEOR (Denkowski and Lavie, 2014) and BLEU (Papineni et al., 2002) to compare the machine-generated descriptions to human ones. For the movie corpora which have just a single description we use only METEOR which is more robust.

Human Evaluation. We also obtain human judgements using Amazon Turk on a random subset of 200 video clips for each dataset. Each sentence was rated by 3 workers on a Likert scale of 1 to 5 (higher is better) for relevance and grammar. No video was provided during grammar evaluation. For movies, due to copyright, we only evaluate on grammar.

Model	METEOR	B-4	Relevance	Grammar
S2VT	29.2	37.0	2.06	3.76
Early Fusion	29.6	37.6	-	-
Late Fusion	29.4	37.2	-	-
Deep Fusion	29.6	39.3	-	-
Glove	30.0	37.0	-	-
Glove+Deep				
- Web Corpus	30.3	38.1	2.12	4.05*
- In-Domain	30.3	38.8	2.21*	4.17*
Ensemble	**31.4**	**42.1**	**2.24***	**4.20***

Table 1: Youtube dataset: METEOR and BLEU@4 in %, and human ratings (1-5) on relevance and grammar. Best results in bold, * indicates significant over S2VT.

4.1 Youtube Video Dataset Results

Comparison of the proposed techniques in Table 1 shows that Deep Fusion performs well on both METEOR and BLEU; incorporating Glove embeddings substantially increases METEOR, and combining them both does best. Our final model is an ensemble (weighted average) of the Glove, and the two Glove+Deep Fusion models trained on the external and in-domain COCO (Lin et al., 2014) sentences. We note here that the state-of-the-art on this dataset is achieved by HRNE (Pan et al., 2015) (METEOR 33.1) which proposes a superior visual processing pipeline using attention to encode the video.

Human ratings also correlate well with the METEOR scores, confirming that our methods give a modest improvement in descriptive quality. However, incorporating linguistic knowledge significantly[2] improves the grammaticality of the results, making them more comprehensible to human users.

Embedding Influence. We experimented multiple ways to incorporate word embeddings: *(1) GloVe input:* Replacing one-hot vectors with GloVe on the LSTM input performed best. *(2) Fine-tuning:* Initializing with GloVe and subsequently fine-tuning the embedding matrix reduced validation results by 0.4 METEOR. *(3) Input and Predict.* Training the LSTM to accept and predict GloVe vectors, as described in Section 3, performed similar to (1). All scores reported in Tables 1 and 2 correspond to the setting in (1) with GloVe embeddings only as input.

Model	MPII-MD		M-VAD	
	METEOR	Grammar	METEOR	Grammar
S2VT†	6.5	2.6	6.6	2.2
Early Fusion	6.7	-	6.8	-
Late Fusion	6.5	-	6.7	-
Deep Fusion	6.8	-	6.8	-
Glove	6.7	3.9*	6.7	3.1*
Glove+Deep	6.8	**4.1***	6.7	**3.3***

Table 2: Movie Corpora: METEOR (%) and human grammar ratings (1-5, higher is better). Best results in bold, * indicates significant over S2VT.

S2VT: Someone sits in the bed.
Glove: Someone sits on the couch and watches her phone.
Glove+Deep: Someone sits on the couch, watching her, her feet on her lap.
GT: Someone drops the flowers and kisses someone.

Figure 3: Two frames from a clip. Models generate visually relevant sentences but differ from groundtruth (GT).

4.2 Movie Description Results

Results on the movie corpora are presented in Table 2. Both MPII-MD and M-VAD have only a single ground truth description for each video, which makes both learning and evaluation very challenging (E.g. Fig.3). METEOR scores are fairly low on both datasets since generated sentences are compared to a single reference translation. S2VT† is a re-implementation of the base S2VT model with the new vocabulary and architecture (embedding dimension). We observe that the ability of external linguistic knowledge to improve METEOR scores on these challenging datasets is small but consistent. Again, human evaluations show significant (with $p < 0.0001$) improvement in grammatical quality.

5 Related Work

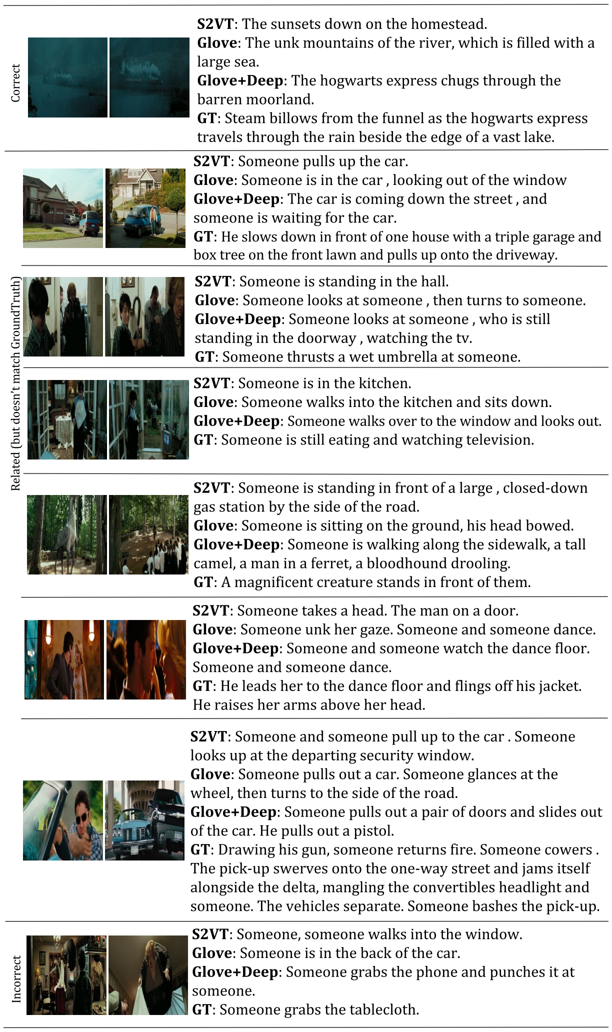
Following the success of LSTM-based models on Machine Translation (Sutskever et al., 2014; Bahdanau et al., 2015), and image captioning (Vinyals et al., 2015; Donahue et al., 2015), recent video description works (Venugopalan et al., 2015b; Venugopalan et al., 2015a; Yao et al., 2015) propose CNN-RNN based models that generate a vector representation for the video and "decode" it using an LSTM sequence model to generate a description. Venugopalan et al. (2015b) also incorporate external data such as images with captions to improve

[2] Using the Wilcoxon Signed-Rank test, results were significant with $p < 0.02$ on relevance and $p < 0.001$ on grammar.

video description, however in this work, our focus is on integrating external linguistic knowledge for video captioning. We specifically investigate the use of distributional semantic embeddings and LSTM-based language models trained on external text corpora to aid existing CNN-RNN based video description models.

LSTMs have proven to be very effective language models (Sundermeyer et al., 2010). Gulcehre et al. (2015) developed an LSTM model for machine translation that incorporates a monolingual language model for the target language showing improved results. We utilize similar approaches (late fusion, deep fusion) to train an LSTM for translating video to text that exploits large monolingual-English corpora (Wikipedia, BNC, UkWac) to improve RNN based video description networks. However, unlike Gulcehre et al. (2015) where the monolingual LM is used only to tune specific parameters of the translation network, the key advantage of our approach is that the output of the monolingual language model is used (as an input) when training the full underlying video description network.

Contemporaneous to us, Yu et al. (2015), Pan et al. (2015) and Ballas et al. (2016) propose video description models focusing primarily on improving the video representation itself using a hierarchical visual pipeline, and attention. Without the attention mechanism their models achieve METEOR scores of 31.1, 32.1 and 31.6 respectively on the Youtube dataset. The interesting aspect, as demonstrated in our experiments (Table 1), is that the contribution of language alone is considerable and only slightly less than the visual contribution on this dataset. Hence, it is important to focus on both aspects to generate better descriptions.



S2VT: The sunsets down on the homestead.
Glove: The unk mountains of the river, which is filled with a large sea.
Glove+Deep: The hogwarts express chugs through the barren moorland.
GT: Steam billows from the funnel as the hogwarts express travels through the rain beside the edge of a vast lake.

S2VT: Someone pulls up the car.
Glove: Someone is in the car , looking out of the window
Glove+Deep: The car is coming down the street , and someone is waiting for the car.
GT: He slows down in front of one house with a triple garage and box tree on the front lawn and pulls up onto the driveway.

S2VT: Someone is standing in the hall.
Glove: Someone looks at someone , then turns to someone.
Glove+Deep: Someone looks at someone , who is still standing in the doorway , watching the tv.
GT: Someone thrusts a wet umbrella at someone.

S2VT: Someone is in the kitchen.
Glove: Someone walks into the kitchen and sits down.
Glove+Deep: Someone walks over to the window and looks out.
GT: Someone is still eating and watching television.

S2VT: Someone is standing in front of a large , closed-down gas station by the side of the road.
Glove: Someone is sitting on the ground, his head bowed.
Glove+Deep: Someone is walking along the sidewalk, a tall camel, a man in a ferret, a bloodhound drooling.
GT: A magnificent creature stands in front of them.

S2VT: Someone takes a head. The man on a door.
Glove: Someone unk her gaze. Someone and someone dance.
Glove+Deep: Someone and someone watch the dance floor. Someone and someone dance.
GT: He leads her to the dance floor and flings off his jacket. He raises her arms above her head.

S2VT: Someone and someone pull up to the car . Someone looks up at the departing security window.
Glove: Someone pulls out a car. Someone glances at the wheel, then turns to the side of the road.
Glove+Deep: Someone pulls out a pair of doors and slides out of the car. He pulls out a pistol.
GT: Drawing his gun, someone returns fire. Someone cowers . The pick-up swerves onto the one-way street and jams itself alongside the delta, mangling the convertibles headlight and someone. The vehicles separate. Someone bashes the pick-up.

S2VT: Someone, someone walks into the window.
Glove: Someone is in the back of the car.
Glove+Deep: Someone grabs the phone and punches it at someone.
GT: Someone grabs the tablecloth.

Figure 4: Representative frames from clips in the movie description corpora. S2VT is the baseline model, Glove indicates the model trained with input Glove vectors, and Glove+Deep uses input Glove vectors with the Deep Fusion approach. GT indicates groundtruth sentence.

6 Conclusion

This paper investigates multiple techniques to incorporate linguistic knowledge from text corpora to aid video captioning. We empirically evaluate our approaches on Youtube clips as well as two movie description corpora. Our results show significant improvements on human evaluations of grammar while modestly improving the overall descriptive quality of sentences on all datasets. While the proposed techniques are evaluated on a specific video-caption network, they are generic and can be applied to many captioning models. The code and models are shared on `http://vsubhashini.github.io/language_fusion.html`.

Acknowledgements

This work was supported by NSF awards IIS-1427425 and IIS-1212798, and ONR ATL Grant N00014-11-1-010, and DARPA under AFRL grant FA8750-13-2-0026. Raymond Mooney and Kate Saenko also acknowledge support from a Google grant. Lisa Anne Hendricks is supported by the National Defense Science and Engineering Graduate (NDSEG) Fellowship.

References

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

[Ballas et al.2016] Nicolas Ballas, Li Yao, Chris Pal, and Aaron C. Courville. 2016. Delving deeper into convolutional networks for learning video representations. *ICLR*.

[Chen and Dolan2011] David Chen and William Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *ACL*.

[Cho et al.2014] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. *Syntax, Semantics and Structure in Statistical Translation*, page 103.

[Denkowski and Lavie2014] Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *EACL*.

[Donahue et al.2015] Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*.

[Gulcehre et al.2015] C. Gulcehre, O. Firat, K. Xu, K. Cho, L. Barrault, H.C. Lin, F. Bougares, H. Schwenk, and Y. Bengio. 2015. On using monolingual corpora in neural machine translation. *arXiv preprint arXiv:1503.03535*.

[Hansen and Salamon1990] L. K. Hansen and P. Salamon. 1990. Neural network ensembles. *IEEE TPAMI*, 12(10):993–1001, Oct.

[Hochreiter and Schmidhuber1997] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8).

[Lin et al.2014] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV*.

[Mikolov et al.2013] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *NIPS*.

[Pan et al.2015] Pingbo Pan, Zhongwen Xu, Yi Yang, Fei Wu, and Yueting Zhuang. 2015. Hierarchical recurrent neural encoder for video representation with application to captioning. *CVPR*.

[Papineni et al.2002] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*.

[Pennington et al.2014] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014)*, 12:1532–1543.

[Rohrbach et al.2015] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. 2015. A dataset for movie description. In *CVPR*.

[Sundermeyer et al.2010] M. Sundermeyer, R. Schluter, and H. Ney. 2010. Lstm neural networks for language modeling. In *INTERSPEECH*.

[Sutskever et al.2014] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*.

[Torabi et al.2015] 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:1503.01070v1*.

[Venugopalan et al.2015a] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. 2015a. Sequence to sequence - video to text. *ICCV*.

[Venugopalan et al.2015b] 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*.

[Vinyals et al.2015] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. *CVPR*.

[Yao et al.2015] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. 2015. Describing videos by exploiting temporal structure. *ICCV*.

[Yu et al.2015] Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. 2015. Video paragraph captioning using hierarchical recurrent neural networks. *CVPR*.