SEEING AND HEARING TOO: AUDIO REPRESENTATION FOR VIDEO CAPTIONING

Shun-Po Chuang, Chia-Hung Wan, Pang-Chi Huang, Chi-Yu Yang, Hung-Yi Lee

National Taiwan University
Graduate Institute of Communication Engineering

ABSTRACT

Video captioning has been widely researched. Most related work takes into account only visual content in generating descriptions. However, auditory content such as human speech or environmental sounds contains rich information for describing scenes, but has yet to be widely explored for video captions. Here, we experiment with different ways to use this auditory content in videos, and demonstrate improved caption generation in terms of popular evaluation methods such as BLEU, CIDEr, and METEOR. We also measure the semantic similarities between generated captions and human-provided ground truth using sentence embeddings, and find that good use of multi-modal contents helps the machine to generate captions that are more semantically related to the ground truth. When analyzing the generated sentences, we find some ambiguous situations for which visual-only models yield incorrect results but that are resolved by approaches that take into account auditory cues.

Index Terms— Video caption generation

1. INTRODUCTION

Video captioning, in which the machine generates one or more sentences that describe the content of a video clip, is a critical step towards machine intelligence; useful applications include video retrieval, automatic video subtitling, and blind navigation. Video captioning has been widely studied; most related work exploits only visual contents. However, as humans take in an environment not only by seeing but also by hearing, we believe that the auditory content of video contains rich information for video captioning. For example, by seeing only it is difficult to differentiate between ‘talking’ and ‘singing’; with hearing, however, these can be easily distinguished. Likewise, it can be hard to decide whether two people are ’talking’ or ‘arguing’ given only visual cues; using voice loudness, though, the two situations can be easily separated. Moreover, it is possible that the source of the sound does not appear in the video, in which case the machine cannot determine the existence of the sound source without hearing. Hence for machine captioning of videos we should consider both visual and auditory cues. There have been attempts to use auditory content to improve video captioning [1, 2], but auditory contents are not always shown to be helpful [3].

In this paper, we attempt to improve video caption generation by using a variety of ways to represent auditory content. In addition to MFCC features, which have been used in previous work [1, 2, 3], we also use ASR system output and audio cues extracted by deep neural networks such as Audio Word2Vec [4] and SoundNet [5]. We compare the performance of different audio representations under different video caption generation models. Using common evaluation measures like BLEU [6], CIDEr [7], ROUGE_L [8], and METEOR [8], we evaluate the difference between generated captions and human-labeled sentences textually. In addition to evaluating results textually, we further evaluate results at the semantic level by computing similarities using sentence embeddings [9].

2. RELATED WORKS

For video captioning, there are two main types of methods: template-based methods [10, 11, 12, 13] and sequence learning methods [14, 15, 16, 17, 18, 19, 20, 21]. For template-based methods, objects are first recognized, and the names of the detected objects are filled into predefined language templates to generate captions. The diversity of the generated sentences depends heavily on the number of predefined templates. Sequence learning methods generate sentences with more flexible syntactical structure using a sequence-to-sequence model, which learns the probability of a word sequence given a video clip using the encoder-decoder architecture. In the encoder-decoder architecture, the encoder takes a video clip as input and encodes it to the embedding space, after which the decoder decodes the embedding vector into descriptions, word by word.

Most video captioning relies on visual contents. Few examples exploit auditory contents: Ramanishka et al. propose a multi-modal video description model [1] which uses category labels and audio in the form of MFCCs. Qin Jin et al. also consider multiple modes in video captioning [2]; they represent audio using an acoustic codebook and then concatenate it with visual features. Hori et al. [3] propose expanding the attention model to selectively attend to multi-modal fea-

This work was sponsored by MediaTek Inc., Hsin-chu, Taiwan and Ministry of Science and Technology, R.O.C.

978-1-5090-4788-8/17/$31.00 ©2017 IEEE

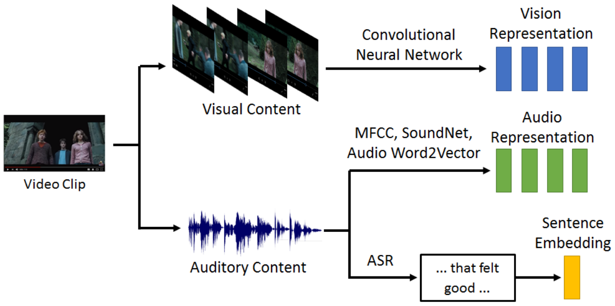


Fig. 1: *Video captioning features*

tures. MFCCs are also used to represent audio, and are used as inputs to a biLSTM which is jointly learned with the whole video caption model.

In video captioning, visual features are usually extracted by a pre-trained deep convolution neural network such as VGG-net [22], Alexnet [23], or GoogleNet [24]; audio is typically simply represented by MFCCs. Although there are several ways to pre-train a deep learning model for audio feature extraction [4, 5, 25, 26, 27], these approaches have not been exploited for video captioning. Chung et al. propose Audio Word2Vec, an unsupervised method for audio representation that uses a sequence-to-sequence auto-encoder [4]. Aytar et al. propose SoundNet, a deep convolution neural network for audio feature extraction [5]. By transferring knowledge from models trained on ImageNet [28] and Place [29], SoundNet features detect environmental sounds and object sounds. Here we enhance video captioning by using features extracted by Audio Word2Vec and SoundNet.

3. APPROACH

3.1. Feature Representation

The features used to represent a video clip here are shown in Fig. 1. Visual contents are represented as a sequence of vectors extracted by a CNN. We focus on exploiting the auditory contents: in addition to mel-frequency cepstral coefficients (MFCCs), we use Audio Word2Vec and SoundNet for audio representations. We also consider ASR transcriptions. More details are shown below.

Audio Word2Vec. Audio Word2Vec [4] encodes an audio segment into a fixed-length vector learned from audio data without human annotation using a sequence-to-sequence audoencoder (SA). The SA consists of two LSTMs: one for encoding and the other for decoding. The encoder reads an audio segment represented as an acoustic feature (e.g., MFCC) sequence $X = (x_1, x_2, x_3...x_T)$ and maps it to a fixed-length vector z, after which the decoder maps the fixed-length vector z to another sequence $Y = (y_1, y_2, y_3...y_T)$. The encoder and decoder are jointly trained to minimize the difference between sequences X and Y, as measured by the mean squared error $\sum_{t=1}^{T} ||x_t - y_t||^2$. Because the input X can be reconstructed from the fixed-length vector z, it is taken as the meaningful representation of the input sequence X. Also, because the SA training target is the network input, Audio Word2Vec needs no labeled data for training. Here, the auditory content of a video clip is segmented into equal-length audio segments, each of which is represented by a fixed-length vector z. Hence, for Audio Word2Vec, the auditory content is also represented as a sequence of vectors.

SoundNet. SoundNet [5] is a deep CNN for natural sound recognition. By transferring learning from the CNNs for object and scene recognition, SoundNet learns to classify objects and scenes using only auditory contents. As the hidden layer output of SoundNet can be used to represent a short raw waveform, we represent the auditory content of a video clip by a sequence of SoundNet vectors.

Sentence Embeddings for ASR transcriptions. Human speech within video clips is helpful for machines to generate more specific descriptions. Therefore, we utilize automatic speech recognition (ASR) to transcribe human voices in videos, and further use the sentence embedding technique [30] to represent these transcriptions as fixed-length vectors.

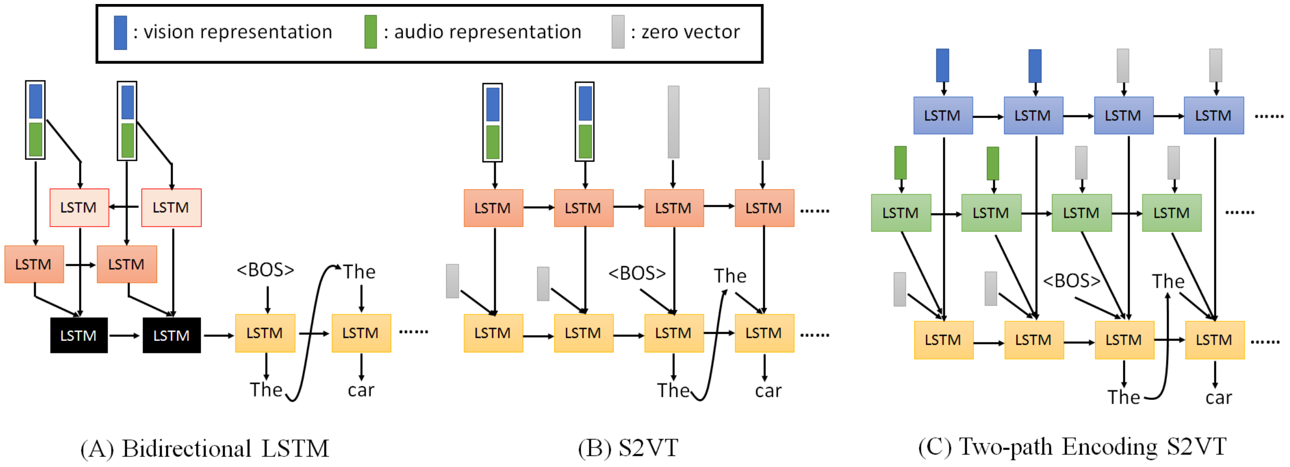
3.2. Model Architecture

The existing models [17, 20] are modified to integrate visual content with auditory content. The three model architectures used here are shown in Fig. 2 and respectively described in Sections 3.2.1, 3.2.2, and 3.2.3. In Fig. 2, the feature sequence extracted from the visual content is the blue vectors; the green vectors are either MFCCs or Audio Word2Vec or SoundNet features. Here we assume the number of blue and green vectors is equal for a given video clip (in Section 4.2 we show how to achieve this). In Section 3.2.4 we describe how to use ASR sentence embedding outputs.

3.2.1. Bidirectional LSTM

The first model in Fig.2 (A) is modified from that proposed by Bin et al. [20]. The input is the concatenation of visual features with MFCCs, or features from Audio Word2Vec or SoundNet. The model encodes the input with a bidirectional-LSTM (light and dark orange blocks in Fig.2 (A)). The outputs of bidirectional-LSTM are further processed by a forward LSTM (the black blocks), whose final output is the whole video representation[1]. In the decoding stage, the LSTM to generate the description (the yellow blocks) takes the whole video representation as input, and produces as output the description, word by word until the EOS (end of sentence) token is generated.

[1] In the original paper [20], the outputs of bidirectional-LSTM were concatenated with original visual features as the input of another LSTM to generate the final video representation; in our preliminary experiments, however, this did not prove helpful.



(A) Bidirectional LSTM

(B) S2VT

(C) Two-path Encoding S2VT

Fig. 2: *Model architecture for video captioning using both visual and auditory content*

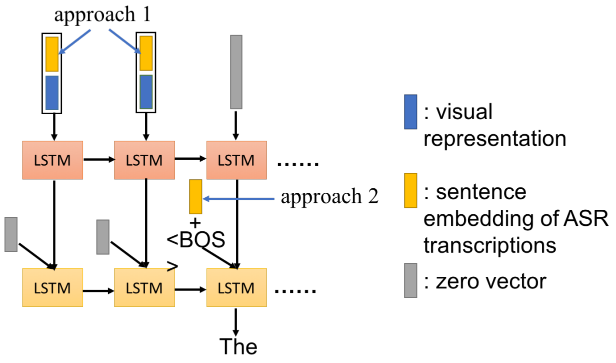


Fig. 3: The S2VT model used to describe two approaches to using ASR transcriptions. The same idea can be used for all models in Fig. 2.

3.2.2. *S2VT*

Shown in Fig. 2 (B), our second model is the S2VT model [17], a two-layer LSTM, the first layer for processing audio-visual content (the upper blocks in orange), and the second layer for generating captions (the lower blocks in yellow). The S2VT-based caption generation process is composed of an encoding stage (the first two time steps in Fig. 2 (B)) and a decoding stage (the remaining steps). During encoding, the concatenated visual and audio features are sequentially fed into the upper LSTM; the upper LSTM outputs are concatenated with zero vectors according to the vocabulary size to form the input of the lower LSTM. During decoding, the upper LSTM input is replaced by zero vectors, and the lower LSTM input is the concatenation of the upper LSTM output and a one-hot encoding of the word generated in the last time step. For the first decoding time step, the input is the BOS (beginning of sentence) token. The lower LSTM output is the words in the generated caption. We collect the words in sequence until the EOS token is generated.

3.2.3. *Two-path Encoding S2VT*

In Section 3.2.2, the upper LSTM in S2VT processes both visual and audio information. Here we modify the S2VT model to reduce the upper LSTM's workload: the visual and acoustic features are encoded by separate LSTMs (blue and green blocks), after which the outputs of the two LSTMs are concatenated as the input of the lower LSTM.

3.2.4. *Exploiting ASR Transcriptions*

As the sentence embedding of the ASR transcription is a single vector as opposed to a vector sequence, it is used differently from other audio representations. In Fig. 3 we use the S2VT model to demonstrate the two approaches for sentence embedding. In the first approach, we duplicate the sentence embedding, and concatenate it with the visual features. In the second approach, to more directly influence the caption generation process, we concatenate the sentence embedding with the one-hot encoding of the BOS token.

4. EXPERIMENTAL SETUP

4.1. Dataset

In the experiments we used Microsoft Research's Video to Text (MSR-VTT) Corpus [16], which contains 7010 video clips for training and 2990 clips for testing. Each clip corresponds to 20 natural language descriptions labeled by AMT workers. Because some clips are not available, and some have no audio, we use only 5928 clips for training and 2623 for testing. To facilitate reproduction of this work, we provide training and testing JSON files containing the video IDs and corresponding captions used here[2].

[2] https://github.com/alex82528/video_captioning_data

Table 1: Evaluation scores. Pairwised t-test with significance level at 0.05 was performed over the overall results. The superscripts * indicate significantly better than the result in rows ”Visual-Only”.

(A) Bi-LSTM		BLEU@1	BLEU@2	BLEU@3	BLEU@4	CIDEr	ROUGH-L	METEOR
(A-1)	Visual-Only	0.636	0.483	0.360	0.260	0.252	**0.517**	0.223
(A-2)	MFCC	0.649*	0.482	0.355	0.255	0.249	0.510	0.221
(A-3)	A2V-Speech	0.646	0.485	0.356	0.254	0.241	0.509	0.220
(A-4)	A2V-Video	0.641	0.486	**0.362**	**0.261**	**0.258**	0.515	0.220
(A-5)	SoundNet	**0.653***	**0.492**	**0.362**	0.260	0.247	0.516	**0.226**
(A-6)	ASR-Con	0.649*	0.487	0.360	0.259	0.245	0.513	0.222
(A-7)	ASR-Dec	0.600	0.442	0.320	0.224	0.220	0.499	0.210
(B) S2VT		BLEU@1	BLEU@2	BLEU@3	BLEU@4	CIDEr	ROUGH-L	METEOR
(B-1)	Visual-Only	0.666	0.513	0.386	0.279	0.260	0.529	0.226
(B-2)	MFCC	0.676	0.527	0.403	0.297	0.282	0.534*	0.231*
(B-3)	A2V-Speech	0.675*	0.525*	0.399*	0.293*	0.290*	0.537*	0.232*
(B-4)	A2V-Video	0.679*	0.527	0.399	0.292	0.289*	0.537*	0.233*
(B-5)	SoundNet	**0.682***	**0.536***	**0.412***	**0.305***	0.279*	**0.539***	**0.233***
(B-6)	ASR-Con	0.673	0.521	0.393	0.289*	**0.288***	0.534	0.232*
(B-7)	ASR-Dec	0.681*	0.530	0.405*	0.297	0.265	0.536*	0.231*
(C) 2-S2VT		BLEU@1	BLEU@2	BLEU@3	BLEU@4	CIDEr	ROUGH-L	METEOR
(C-1)	Visual-Only	0.666	0.513	0.386	0.279	0.260	0.529	0.226
(C-2)	MFCC	0.624	0.458	0.329	0.225	0.170	0.489	0.204
(C-3)	A2V-Speech	0.660	0.493	0.362	0.254	0.224	0.515	0.218
(C-4)	A2V-Video	0.664	0.515	0.392	0.289	**0.280***	**0.536***	0.231*
(C-5)	SoundNet	0.681*	0.525	0.394	0.286	0.260	0.531	**0.232***
(C-6)	ASR-Con	**0.688***	**0.530**	0.397	0.285	0.267	0.534	0.231
(C-7)	ASR-Dec	0.681*	**0.530**	**0.405***	**0.297**	0.265	**0.536***	0.231*

4.2. Feature Representation

Visual Content Representation. We extract visual features via the *fc7* layer of the VGG 19-layer model [22], which is composed of 4096-dimensional vectors. Following the settings in previous work [17], in each video clip, visual features are extracted only from 80 sampled frames, resulting in 80 feature vectors for each clip. These frames are sampled by the same time interval in each clip, but the lengths of the time intervals are different for different clips. Here we focus on auditory features, and thus do not use other visual features such as optical flow or C3D [1, 3, 17, 19]. Because 80 feature vectors are extracted from the visual content, the auditory content of a video clip is also represented by 80 vectors, regardless of the representation approach, except for the sentence embedding of the ASR system output, which is a single vector.

MFCC: 39-dimensional MFCCs are extracted using the Kaldi toolkit [31], and cepstral mean and variance normalization are applied. For each clip, we sample 80 MFCC features using the same time interval for caption generation.

Audio Word2Vec: We first segment the audio of each clip into 80 equal-length segments. As in previous work [4], we use a sequence-to-sequence autoencoder to encode each segment into a 300-dimensional vector. This Audio Word2Vec model was learned either from the Librispeech ASR corpus [32] or from audio content in the training video clips in the MSR-VTT corpus; thus we have two sets of Audio Word2Vec in the following experiments. This enables us to observe the influence of the Audio Word2Vec training data domain on video captioning. The Librispeech ASR corpus contains approximately 1000 hours of English speech data. By training on this corpus, Audio Word2Vec maintains the characteristics of human speech [4]. As the MSR-VTT video clips contain sound other than human speech, the feature vectors extracted from Audio Word2Vec as learned from MSR-VTT may reflect information other than human speech.

SoundNet: Trained on over two million Flickr videos, over a year's length of continuous natural sound and video, SoundNet has learned a good representation of audio from a large corpus of unlabeled video. 5-layer and 8-layer models are available [29]. We use the *pool5* layer in the 8-layer model, which performed best in classification tasks [29]. A sequence of 256-dimensional vectors is extracted from each video clip. Because the number of feature vectors extracted by SoundNet can be fewer or greater than 80, we downsample or upsample the features accordingly. Each dimension in a feature vector extracted by SoundNet corresponds to a specific pattern. Because the feature vectors are the outputs of a pooling layer, and the ReLU activation function is used, all values in the extracted vectors are non-negative. As we suspect that each non-negative value reveals the existence of a specific pattern, more non-negative values in the vectors suggests richer information in the corresponding time

Table 2: Ensembled results of different audio representations using S2VT model. Results in row (A) are those in Table 1, row (B-5).

	BLEU@1	BLEU@2	BLEU@3	BLEU@4	CIDEr	ROUGH-L	METEOR
(A) Soundnet	0.682	0.536	0.412	0.305	0.279	0.539	0.233
(B) +A2V-Speech	0.698	0.553	0.430	0.323	0.315	0.552	0.241
(C) +ASR-Con	0.701	0.557	0.432	0.327	0.319	0.553	0.244
(D) +A2V-Speech+ASR-Con	**0.703**	**0.559**	**0.436**	**0.331**	**0.331**	**0.557**	**0.245**

span; we thus define the importance of a feature vector as the summation of the elements in the feature. When a clip has $80 + K$ frames, we dump the least important K frames; when a clip has $80 - K$ frames, we duplicate the most important K frames[3].

Sentence Embedding: We used the Microsoft Azure Bing Speech API to generate ASR transcriptions, and applied a sentence embedding model [30] to encode it. Trained on English tweets, the model outputs a 700-dimensional vector given a sentence. For clips not containing speech, the ASR system generates no results; in this case, we represent the sentence with a zero vector.

4.3. Parameter Settings

All LSTM were initialized with a uniform distribution in range of -0.1 to 0.1. S2VT-based models used 256-dimensional LSTMs, and 512-dimensional LSTMs were used in the bidirectional LSTM-based model in the following experiments[4]. The vocabulary size was set to 3000 and we did not use a pretrained language model during the decoding stage. We used a linear transformation to reduce the one-hot representation of a word to a 300-dimensional vector as the LSTM input; the transformation weights were trained jointly with the model. We trained models for 200 epochs with a batch size of 100 and the Adam optimizer.

4.4. Evaluation Methods

Metrics BLEU, METEOR, ROUGH-L and CIDEr were used[5]. These metrics are widely utilized for natural language generation tasks such as machine translation.

We also propose a new evaluation measure. We trained a Sent2Vec model on the testing set of the MSR-VTT Corpus, after which we encoded the ground truth descriptions and generated descriptions as 700-dimension vectors, and then computed their cosine similarities. Higher similarity values correspond to generated descriptions that are semantically close to the ground truth.

Table 3: Semantic evaluation by similarity between sentence embedding of generated captions and ground truth. Shows results of S2VT with different features (part (b) of Table 1 and Table 2).

Features	Similarity
(A) Visual-only	0.443
(B) MFCC	0.452
(C) A2V-Speech	**0.456**
(D) A2V-Video	**0.456**
(E) SoundNet	0.450
(F) ASR-Con	0.454
(G) ASR-Dec	0.448
(H) A2V-Speech+SoundNet	0.470
(I) A2V-Speech+ASR-Con	0.471
(J) SoundNet+ASR-Con	0.472
(K) A2V-Speech+SoundNet+ASR-Con	**0.477**

Thus the former metrics can be seen as textual evaluation; the latter, semantic evaluation.

5. EXPERIMENTAL RESULTS

5.1. Evaluation Scores

Table 1 shows the results of textual evaluation. Parts (A), (B), and (C) are respectively the results for bidirectional-LSTM, S2VT, and two-path encoding S2VT (2-S2VT). Row (1) in each part is the baseline using visual information only. Because two-path encoding S2VT reduces to the original S2VT when auditory content is not taken into account, the results in (B-1) and (C-1) are the same. Rows (2) to (7) exploit auditory content in different ways. A2V-Speech and A2V-Video stand for Audio Word2Vec learned from the Librispeech ASR corpus and the MSR-VTT corpus respectively. ASR-Con and ASR-Dec represent the two uses of sentence embedding: as a concatenation with visual features, or as input during decoding. The results in rows (B-7) and (C-7) are the same because when only the ASR sentence embedding is used in the decoding state, S2VT and two-path encoding S2VT are exactly the same. Bold results are the best among all kinds of features for the same model architecture, and underlined results are the best across all models and features.

Auditory contents do not have an obvious influence on the bidirectional LSTM model. MFCC is not helpful for any measures save BLEU@1 (rows (A-2) v.s. (A-1)). A2V-Video and SoundNet increase scores slightly in terms of some but not all evaluation measures (rows (A-4), (A-5) s.v. (A-1)).

[3] Although this method makes visual and auditory contents asynchronous, it leads to slightly better performance than sampling SoundNet feature at equal intervals. Due to space limitations, we do not show related experiments of this comparison.

[4] In preliminary experiments, we found that 256- and 512-dimensional LSTMs achieved the best results for S2VT and BiLSTM respectively. Due to space limitations, we do not show these results.

[5] Implemented as `https://github.com/vsubhashini/caption-eval`

Table 4: Results given various features

`https://www.youtube.com/watch?v=1DQwhuFhcJk`
start time: 168.17, end time: 179.48

Ground Truth	person singing a song
Visual-Only	a man is **talking** about a UNK
A2V-Speech	a man is **singing** a song
SoundNet	a man is **singing**

`https://www.youtube.com/watch?v=KMydT2yve3k`
start time: 955.61, end time: 972.57

Ground Truth	people on tv show are talking to a caller
Visual-Only	a **woman** is talking about the lady
A2V-Speech	a **man** is talking to a **woman**
SoundNet	a **man** is talking about a **woman** s UNK

`https://youtu.be/XTHdVBgEQrk`
start time: 0.0, end time: 10.0

Ground Truth	a basketball players makes a jump shot
Visual-Only	a group of people are dancing
A2V-Speech	a man is **running**
SoundNet	a **basketball** player is **scoring** a goal

This may be because the bidirectional LSTM model represents the whole video using a single vector; it may be difficult to use a vector to thus represent both visual and audio information[6]. S2VT and 2-S2VT yielded obvious improvements. With the S2VT model (part (B)), all auditory content, especially SoundNet, enhanced scores. In the 2-S2VT model (part (C)), audio features enhanced performance, except for MFCC features and A2V-Speech (rows (C-2), (C-3) v.s. (C-1)). Comparing the S2VT and 2-S2VT results (parts (C) v.s. (B)), we find that with audio information, S2VT outperformed 2-S2VT in all cases, except when using ASR-Con (rows (C-6) v.s. (B-6)). The results show that the interaction between the vision and audio information in the upper LSTM of S2VT is helpful; 2-S2VT outperforms S2VT with ASR-Con probably because the ASR results and the VGG features are at different levels, and little interaction is needed between them. Comparing all models and features in Table 1, SoundNet features with S2VT performed the best (row (B-5)). Because the S2VT model achieved the best performance in Table 1, it was used for the following experiments.

We further integrated the results from the S2VT models (part (B) of Table 1); these ensemble results are listed in Table 2. The likelihood of the output of each model[7] was first computed and normalized by its length. We take the normalized likelihoods as the confidence of the generated sentences. Among the results to be integrated, the generated sentence with the highest normalized likelihood is selected for evaluation. Because S2VT plus SoundNet (row (B-5) in Table 1) achieved the best results, we integrated it with the results of other models. Many different combinations were tested[8], and we found that ASR-Con and A2V-Speech are most complementary with SoundNet – see rows (B), (C), and (D) of Table 2. Because SoundNet attempts to detect audio events instead of the content of human speech[9], it is reasonable that ASR-Con and A2V-Speech, which contain information on the content of speech, improved performance after integration.

We then evaluated the semantic correctness of the generated sentences. As each video has several ground truth descriptions, we computed the cosine similarities between the generated sentence and each ground truth description, and took the maximum similarity as the evaluation score. We averaged the maximum similarity of each video clip over the testing set. Table 3 shows the semantic evaluation results of S2VT with different features (the textual evaluations of this set of results are also shown in part (b) of Table 1 and Table 2). Row (A) shows the results using vision features only, and rows (B) to (G) show the results when auditory contents are exploited. In terms of semantic evaluation, auditory contents also increase the evaluation scores regardless of the representation approach. Rows (H) to (K) are the ensemble results. Integrating the results of A2V-Speech, SoundNet, and ASR-Con improved performance in terms of semantic accuracy.

5.2. Observation

Table 4 lists some example results from the MSR-VTT testing set. The YouTube link and the start and end times of each example are shown. In each example, we display one of the ground truth descriptions, the generated captions of vision only, as well as those with A2V-Speech or SoundNet. We observe that auditory features appear to be sensitive to the sound of gender and auditory-related action. In the first example video clip, there is a person singing. Without hearing, the machine outputs "a man is talking ... ". After considering the auditory content, the machine generates the description "a man is singing ...". In the second example, a male host is talking to a female caller. Comparing the results with and without audio features, we find that the machine is aware there is a man in the scene only when it hears. The last example is a basketball highlight. Our results with audio features are close to the video.

6. CONCLUSION

In this paper, we utilize different kinds of acoustic features in different video captioning models. We find that considering auditory contents improves task performance. S2VT plus SoundNet achieves the best performance; integration with models that exploit human speech yields further improvements to performance. In future work, we plan to use other different models to investigate how vision and audio information interact with each other in video caption generation.

[6] In future work, we plan to improve this using the attention mechanism.

[7] The sum of the logarithm probabilities of each generated word in the sentence.

[8] Detailed results omitted due to space limitations.

[9] In SoundNet, all human speech is likely considered the same event and is represented using similar features.

7. REFERENCES

[1] Vasili Ramanishka, Abir Das, Dong Huk Park, Subhashini Venugopalan, Lisa Anne Hendricks, Marcus Rohrbach, and Kate Saenko, "Multimodal video description," in *Proceedings of the 2016 ACM on Multimedia Conference*, New York, NY, USA, 2016, MM '16, pp. 1092–1096, ACM.

[2] Qin Jin, Junwei Liang, and Xiaozhu Lin, "Generating natural video descriptions via multimodal processing," in *INTERSPEECH*, 2016.

[3] Chiori Hori, Takaaki Hori, Teng-Yok Lee, Kazuhiro Sumi, John R Hershey, and Tim K Marks, "Attention-based multimodal fusion for video description," *arXiv preprint arXiv:1701.03126*, 2017.

[4] Yu-An Chung, Chao-Chung Wu, Chia-Hao Shen, Hung-Yi Lee, and Lin-Shan Lee, "Audio Word2Vec: Unsupervised learning of audio segment representations using sequence-to-sequence autoencoder," *arXiv preprint arXiv:1603.00982*, 2016.

[5] Yusuf Aytar, Carl Vondrick, and Antonio Torralba, "SoundNet: Learning sound representations from unlabeled video," in *Advances in Neural Information Processing Systems*, 2016, pp. 892–900.

[6] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

[7] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh, "CIDEr: Consensus-based image description evaluation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 4566–4575.

[8] Satanjeev Banerjee and Alon Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 2005, vol. 29, pp. 65–72.

[9] Quoc Le and Tomas Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 1188–1196.

[10] Marcus Rohrbach, Wei Qiu, Ivan Titov, Stefan Thater, Manfred Pinkal, and Bernt Schiele, "Translating video content to natural language descriptions," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 433–440.

[11] Anna Rohrbach, Marcus Rohrbach, Wei Qiu, Annemarie Friedrich, Manfred Pinkal, and Bernt Schiele, "Coherent multi-sentence video description with variable level of detail," in *German Conference on Pattern Recognition*. Springer, 2014, pp. 184–195.

[12] Ran Xu, Caiming Xiong, Wei Chen, and Jason J Corso, "Jointly modeling deep video and compositional text to bridge vision and language in a unified framework," in *AAAI*, 2015, vol. 5, p. 6.

[13] Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond Mooney, Trevor Darrell, and Kate Saenko, "Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 2712–2719.

[14] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko, "Translating videos to natural language using deep recurrent neural networks," *arXiv preprint arXiv:1412.4729*, 2014.

[15] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville, "Describing videos by exploiting temporal structure," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4507–4515.

[16] Jun Xu, Tao Mei, Ting Yao, and Yong Rui, "MSR-VTT: A large video description dataset for bridging video and language," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 5288–5296.

[17] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko, "Sequence to sequence-video to text," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4534–4542.

[18] Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu, "Video paragraph captioning using hierarchical recurrent neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4584–4593.

[19] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui, "Jointly modeling embedding and translation to bridge video and language," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4594–4602.

[20] Yi Bin, Yang Yang, Zi Huang, Fumin Shen, Xing Xu, and Heng Tao Shen, "Bidirectional long-short term memory for video description," *CoRR*, vol. abs/1606.04631, 2016.

[21] Pingbo Pan, Zhongwen Xu, Yi Yang, Fei Wu, and Yueting Zhuang, "Hierarchical recurrent neural encoder for video representation with application to captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1029–1038.

[22] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., pp. 1097–1105. Curran Associates, Inc., 2012.

[24] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4842, 2014.

[25] Herman Kamper, Weiran Wang, and Karen Livescu, "Deep convolutional acoustic word embeddings using word-pair side information," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4950–4954.

[26] Wanjia He, Weiran Wang, and Karen Livescu, "Multi-view recurrent neural acoustic word embeddings," *CoRR*, vol. abs/1611.04496, 2016.

[27] Merlijn Blaauw and Jordi Bonada, "Modeling and transforming speech using variational autoencoders," in *INTERSPEECH*, 2016, pp. 1770–1774.

[28] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.

[29] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva, "Learning deep features for scene recognition using places database," in *Advances in neural information processing systems*, 2014, pp. 487–495.

[30] Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi, "Unsupervised learning of sentence embeddings using compositional n-gram features," *arXiv*, 2017.

[31] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely, "The Kaldi Speech Recognition Toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. Dec. 2011, IEEE Signal Processing Society, IEEE Catalog No.: CFP11SRW-USB.

[32] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "LibriSpeech: An ASR corpus based on public domain audio books," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5206–5210.