Multistage Fusion with Forget Gate for Multimodal Summarization in Open-Domain Videos

Nayu Liu[1,2], **Xian Sun**[1,2,*], **Hongfeng Yu**[1], **Wenkai Zhang**[1], **Guangluan Xu**[1]
[1]Key Laboratory of Network Information System Technology, Aerospace Information Research Institute, Chinese Academy of Sciences
[2]School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences
liunayu18@mails.ucas.ac.cn, sunxian@mail.ie.ac.cn

Abstract

Multimodal summarization for open-domain videos is an emerging task, aiming to generate a summary from multisource information (video, audio, transcript). Despite the success of recent multiencoder-decoder frameworks on this task, existing methods lack fine-grained multimodality interactions of multisource inputs. Besides, unlike other multimodal tasks, this task has longer multimodal sequences with more redundancy and noise. To address these two issues, we propose a multistage fusion network with the fusion forget gate module, which builds upon this approach by modeling fine-grained interactions between the multisource modalities through a multistep fusion schema and controlling the flow of redundant information between multimodal long sequences via a forgetting module. Experimental results on the How2 dataset show that our proposed model achieves a new state-of-the-art performance. Comprehensive analysis empirically verifies the effectiveness of our fusion schema and forgetting module on multiple encoder-decoder architectures. Specially, when using high noise ASR transcripts (WER>30%), our model still achieves performance close to the ground-truth transcript model, which reduces manual annotation cost.

1 Introduction

With the popularity of video platforms, personal videos abound on the Internet. Multimodal summarization for open-domain videos, first organized as a track of the How2 Challenge at the ICML 2019 workshop, aims to integrate multisource information of videos (video, audio, transcript) into a fluent textual summary. An example can be seen in Figure 1. This study, which uses compressed text description to reflect the salient parts of videos, is of considerable significance for helping users better retrieve and recommend videos.

Existing approaches have obtained promising results. For example, Libovický et al. (2018) and Palaskar et al. (2019) utilize multiple encoders to encode videos and audio transcripts and a joint decoder to decode the multisource encodings, which acquire better performance than single modality structures. Despite the effectiveness of these approaches, they only perform multimodal fusion during the decoding stage to generate a target sequence, lacking fine-grained interactions between multisource inputs to complete the missing information of each modality. For example, as shown in Figure 1, text context representations containing birds should be associated with visual semantic information containing parrots to build thorough multimodal representations.

Besides, unlike other multimodal tasks such as visual question answering (Antol et al., 2015; Gao et al., 2015) and multimodal machine translation (Elliott et al., 2015; Specia et al., 2016), a major challenge is that this task has longer input sequences with more noise and redundancy. The flow of noise information during multimodal fusion, such as redundant frames in video and noisy words in transcription, interferes with the interaction and complementarity of the effective information between modalities, which leads to a significant negative effect on the model. Moreover, when using an automatic speech recognition (ASR) system to transform audio to transcription instead of ground-truth transcription, high noise ASR-output transcripts further reduce model performance.

To address these two issues, we propose a multistage fusion network with the fusion forget gate module for multimodal summarization in videos. The model involves multiple information fusion processes to capture the correlation between multisource modalities spontaneously, and a fusion for-

*Corresponding author.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 1834–1845,
November 16–20, 2020.

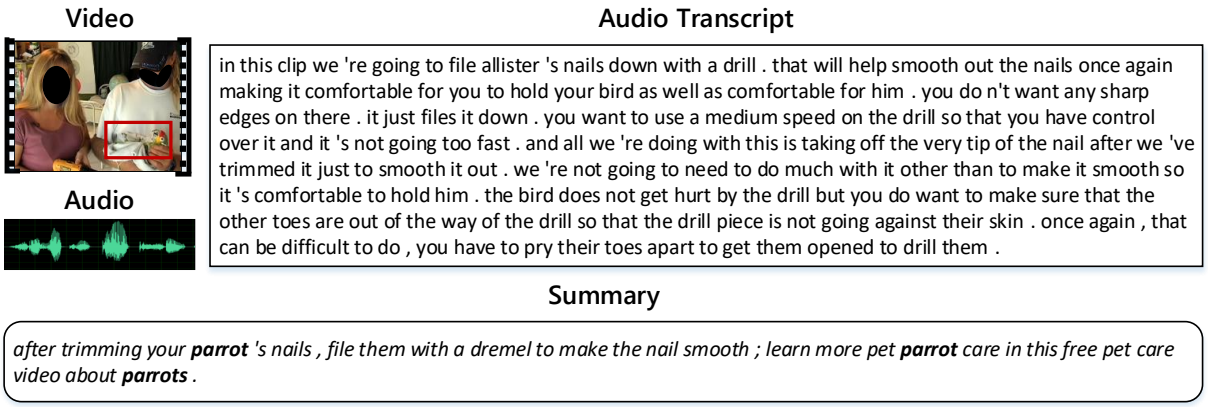

Figure 1: The audio transcript does not mention "**parrot**", only "**bird**" or "**allister**". The complete summary has to be derived from multi-source. This example is taken from the How2 dataset.

get gate is proposed to effectively suppress the flow of unnecessary multimodal noise. As illustrated in Figure 2, our proposed multistage fusion model mainly consists of four modules: 1) multisource encoders to build representations for video and audio (ground-truth or ASR-output transcript); 2) cross fusion block in which cross fusion generator (CFG) and a feature-level fusion layer are designed to generate and fuse latent adaptive streams from one modality to another at low levels of granularity; 3) hierarchical fusion decoder (HFD) in which hierchical attention networks are designed to progressively fuse multisource features carrying adaptive streams from other modalities to generate a target sequence; 4) fusion forget gate (FFG) (detailed in Figure 3) in which a memory vector and a forget vector are created for the information streams in the cross fusion block to alleviate interference from long-range redundant multimodal information.

We build our proposed model on both RNN-based (Sutskever et al., 2014) and transformer-based (Vaswani et al., 2017) encoder-decoder architectures and evaluate our approach on the large-scale public multimodal summarization dataset, How2 (Sanabria et al., 2018). Experiments show that our model achieves a new state-of-the-art performance. Comprehensive ablation experiments and visualization analysis demonstrate the effectiveness of our multistage fusion schema and forgetting module.

Specially, we also evaluate the model performances under the ASR-output transcript. We use an automatic speech recognition (ASR) system (Google-Speech-V2) to generate audio transcripts (word error rate>30%) to replace the ground-truth transcripts provided by the How2 dataset. Experiments show that our model still achieves performance close to the model trained with ground-truth transcripts, and significantly outperforms the state-of-the-art system, which indicates the advantage of our model in the absence of ground-truth transcript annotation.

The extracted ASR-output transcripts and code will be released on https://github.com/forkarinda/MFN.

2 Related Work

Unlike conventional summarization (Rush et al., 2015; See et al., 2017; Narayan et al., 2018), multimodal summarization compresses multimedia documents. According to different tasks, the input modalities are also different, such as text+image (Wang et al., 2012; Bian et al., 2013, 2014; Wang et al., 2016), and video+audio+text (Evangelopoulos et al., 2013; Li et al., 2017), which mainly focus on extractive approaches. With the popularity of sequence-to-sequence learning (Sutskever et al., 2014), the use of corpora with human-written summaries for multimodal abstractive summarization has attracted interest (Li et al., 2018; Zhu et al., 2018, 2020).

The above abstractive summarization research mainly focuses on text and image. Sanabria et al. (2018) first release the How2 dataset for multimodal abstractive summarization for open-domain videos. The dataset provides multisource information, including video, audio, text transcription and human-generated summary. This task is more challenging due to the diversity of multimodal information in the video and the complexity of the video feature space. The task was also added to the How2 Challenge in the 2019 ICML workshop,

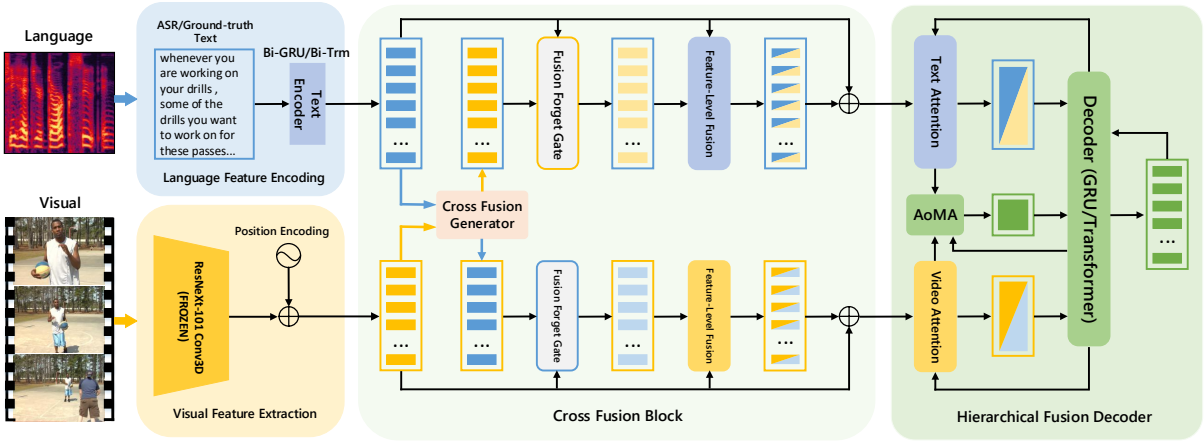

Figure 2: The structure of our full model. It is built on RNN-based and Transformer-based frameworks, respectively.

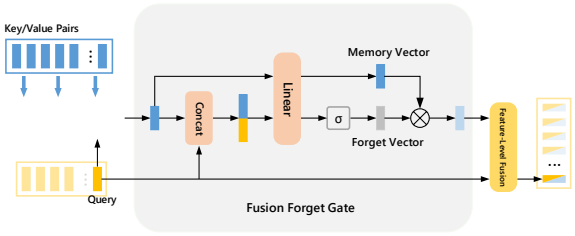

Figure 3: Detail of fusion forget gate. A memory vector and a forgetting vector are created for the information stream flowing through it, and then we get the product of two vectors as the final noise-filtered representation.

which we focus on in this paper. A similar task is video captioning (Venugopalan et al., 2015a,b), which mainly places emphasis on the use of visual information to generate descriptions, but this task focuses on how to make full use of multisource and multimodal long inputs to obtain a summary and additionally needs ground-truth transcripts. Recent methods use multiencoder-decoder RNNs to process multisource inputs but lack the interaction and complementarity between multisource modalities and the ability to resist the flow of multimodal noise. To handle above two challenges, our multistage fusion model is introduced.

3 Multistage Fusion with Forget Gate

In this section, we will explain our model in detail. The overall architecture of our proposed model is shown in Figure 2, and the fusion forget gate inside is illustrated in Figure 3. Specifically, multistage fusion consists of the cross fusion block and hierarchical fusion decoder, which aims to model the correlation and complementarity between modalities spontaneously. In addition, the fusion forget gate is applied in the cross fusion block to filter the flow of redundant information streams. We build our model based on the RNN and transformer encoder-decoder architectures, respectively.

3.1 Problem Definition

Our multimodal summarization system takes a video and a ground-truth or ASR-output audio transcription as input and generates a textual summary that describes the most salient part of the video. Formally, the transcript is a sequence of word tokens $T = (t_1, ..., t_n)$ and the video representation is denoted by $V = (v_1, ..., v_m)$, where v_m is the feature vector extracted by a pretrained model. The output summary is denoted as a sequence of word tokens $S = (s_1, ..., s_l)$ consisting of several sentences. The task aims to predict the best summary sequence S by finding:

$$\arg\max_{\theta} Prob(S|T, V; \theta) \quad (1)$$

where θ is the set of trainable parameters.

3.2 Multisource Encoders

Encoding Video. The video encoding features are obtained by a pretrained action recognition model: a ResNeXt-101 3D convolutional neural network (Hara et al., 2018) trained for recognizing 400 different human actions in the Kinetics dataset (Kay et al., 2017).

$$V = 3DCNN_{ResNeXt-101}(Frames) \quad (2)$$

The video representation features denoted by $V = (v_1, ..., v_m)$ are extracted every 16 nonoverlapping frames, where v_m is the 2048-dimensional vector.

We add learnable position embeddings for video features.

Encoding Transcript. For the RNN encoder, we use a bidirectional GRU (Cho et al., 2014) to encode the text to obtain a contextualized representation for each word:

$$T_{RNN} = BiGRU(t_1, t_2, ..., t_n) \tag{3}$$

For the transformer encoder, we employ an universal bidirectional transformer encoder (Vaswani et al., 2017) in which each layer is composed of a multihead self-attention layer followed by a feed-forward sublayer with residual connections (He et al., 2016) and layer normalizations (Ba et al., 2016), and denoted by the following equation:

$$T_{Trm} = BiTrm(t_1, t_2, ..., t_n) \tag{4}$$

We use learnable position embedding instead of sinusoidal position embedding.

3.3 Cross Fusion Generator

The cross fusion generator (CFG) is used to correlate meaningful elements across modalities. We apply the CFG to generate the adaptive fusion information from one modality encoding to another. The CFG learns two cross-modal attention maps, one is from text to video, and the other is from video to text. It is inspired by parallel co-attention (Lu et al., 2016), which computes an affinity matrix between two sequences, while we apply two unidirectional matrices instead of assigning shared parameters to both directions, and use scaled dot-product attention (Vaswani et al., 2017). At each of the cross-modal attention maps, the low-level signals from the source modality are transformed to key and value pairs to interact with the target modality as a query. Following the two maps, CFG is divided into the video-to-text fusion generator (V2TFG) and text-to-video fusion generator (T2VFG), which are detailed as follows:

Text-to-video Fusion Generator (T2VFG). The T2VFG generates the most relevant video information to low-level text features by a text-to-video cross-modal attention map. The cross-modal attention consists of text queries $Q_T = TW_{Q_T}$, video key and value pairs $K_T = VW_{K_T}, V_T = V$. The contextual video vector derived from the cross-modal attention map is calculated by

$$\begin{aligned} V_{Gen} &= CFG_{T \leftarrow V}(T, V) \\ &= softmax(\frac{Q_T(K_T)^T}{d})V_T \\ &= softmax(\frac{TW_{Q_T}(VW_{K_T})^T}{d})V \\ &= softmax(\frac{TW_{Q_T}(W_{K_T})^T V^T}{d})V \\ &= softmax(\frac{TW_\alpha V^T}{d})V \end{aligned} \tag{5}$$

where the common spatial parameter W_α is used to simplify the calculations.

Video-to-Text Fusion Generator (V2TFG). Similar to the T2VFG, the V2TFG aims to generate the latent adaptive text information stream for video modality. The difference between the V2TFG and T2VFG is that they flow in opposite directions. We transform the low-level video features to queries $Q_V = VW_{Q_V}$ and the text to key and value pairs $K_V = TW_{K_V}, V_V = T$, then calculate:

$$\begin{aligned} T_{Gen} &= CFG_{V \leftarrow T}(T, V) \\ &= softmax(\frac{VW_\beta T^T}{d})T \end{aligned} \tag{6}$$

where W_β is a mapping of text flowing to video.

3.4 Fusion Forget Gate

Although the CFG builds an unsupervised low-level signal alignment between original multi-source features, noise modality information generated by CFG is hard to be suppressed. In particular, when the whole modality cannot guide the task at all, the forced normalization of the softmax function in the attention structure makes the calculated fusion vector generated by the noise modality hard to be suppressed. For this reason, we propose a fusion forget gate (FFG) to filter low-level cross-modal adaptation information of each modality generated by the CFG.

The FFG reads the original modal signals as well as the adaptation information derived from other modalities, and determines whether the adaptation information is noise and matches the original modality. As shown in Figure 3, we assign a video FFG and a text FFG to receive bidirectional adaption information that originated from the CFG.

Specifically, it creates a memory vector and a forget gate to control the flow of noise and mismatched information. First, we project the connected source and target modality embeddings and

activate them with a sigmoid function to obtain a forget vector:

$$Forget_V(V_{Gen}, T) = \sigma([T; V_{Gen}]W_V + b_V) \quad (7)$$

$$Forget_T(T_{Gen}, V) = \sigma([V; T_{Gen}]W_T + b_T) \quad (8)$$

Then the adaptation information passes a linear mapping to obtain a memory vector, which prevents essential information from being weighted down due to the scaling limit of the sigmoid function ranging from 0 to 1. We apply the dot-product to the memory vector and the forget vector to represent the cross-modal adaptive stream after FFG filtering, which is finally calculated as follows:

$$\begin{aligned} &T'_{Gen} = FFG_T(T_{Gen}, V) = \\ &Memory_T(T_{Gen}) \odot Forget_T(T_{Gen}, V) \\ &= (T_{Gen}W_1 + b_1) \odot Forget_T(T_{Gen}, V) \end{aligned} \quad (9)$$

$$\begin{aligned} &V'_{Gen} = FFG_V(V_{Gen}, T) = \\ &Memory_V(V_{Gen}) \odot Forget_V(V_{Gen}, T) \\ &= (V_{Gen}W_2 + b_2) \odot Forget_V(V_{Gen}, T) \end{aligned} \quad (10)$$

where $\odot$ represents elementwise dot production and $W_V, W_T, W_1, W_2, b_V, b_T, b_1$ and b_2 are trainable parameters.

3.5 Feature-Level Fusion

This module combines the low-level signal T/V of the original modality with the matching adaptive stream V'/T' of other modalities. The fusion vector flowing through CFG and FFG has the same sequence length as the original modality so that we apply a concat&forward layer with a ReLU activation function. In addition, we specially add a residual connection inside the fusion layer to deepen the neural network's memory of the original modality. The calculation formulas are below:

$$T_F = Relu(T + [T; V'_{Gen}]W_1 + b_1) \quad (11)$$

$$V_F = Relu(V + [V; T'_{Gen}])W_2 + b_2) \quad (12)$$

where W_1, W_2, b_1, b_2 are trainable parameters.

3.6 Hierarchical Fusion Decoder

The HFD receives multimodal information of different granularity from multisource inputs and generates a target sequence. Inspired by hierarchical attention (Libovický and Helcl, 2017), HFD transforms the decoder hidden states and multisource encodings into a context vector by three attention maps: video attention, text attention, and attention over multimodal attention (AoMA). At each decoding time step t, the decoder hidden state h_t attends to video/text encodings V_F/T_F carrying aligned multimodal information separately via video/text attention to calculate the video/text context vector:

$$C_V = Attn(h_t, V_F) \quad (13)$$

$$C_T = Attn(h_t, T_F) \quad (14)$$

Then, a second attention mechanism is constructed over the two context vectors, and a higher-level context vector is computed. We concatenate the two contexts and apply a new MLP attention:

$$\begin{aligned} C_c =& AoMA(h_t, C_V, C_T) \\ =& softmax(W_1 \tanh(W_2 h_t + \\ & W_3[C_V; V_T])) \cdot [C_V; V_T] \end{aligned} \quad (15)$$

The context vector of hierarchical multimodal fusion is finally obtained and combined with the decoder hidden state vector to compute an output for attending the next decoder layer or caculating the vocabulary distribution.

$$y_{t+1} = Decoder_{RNN/Trm}(x_t, h_t, C_c) \quad (16)$$

Corresponding to the two encoders introduced in section 3.2, we design RNN-based and transformer-based decoding strategies. The formula expression and model diagram of the two structures are detailed in Appendix A.1.

4 Experimental Setup

4.1 How2 Dataset

We evaluate our method on the How2 dataset (Sanabria et al., 2018). The How2 dataset is a large-scale dataset of open-domain videos, spanning different topics, such as cooking, sports, indoor/outdoor activities, and music. It consists of 79,114 how-to instructional videos with an average length of 1.5 minutes and a total of 2,000 hours, accompanied by corresponding ground-truth English transcripts with an average length of 291 words, crowdsourced Portuguese translations of transcripts and user-generated summaries with an average length of 33 words. The statistics are shown in Figure 4 and Table 1.

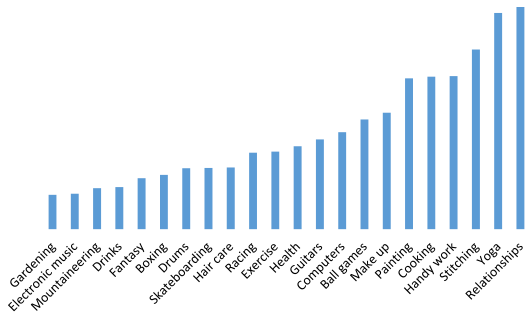

Figure 4: LDA topic distributions of the How2 dataset.

	train	val	test
Videos	73,993	2,965	2,156
Hours	1,766.6	71.3	51.7

Table 1: Statistics of How2 dataset.

4.2 Audio Recognition

We also extract audio transcripts by a speech recognition system (Google-Speech-V2). The word error rate (WER) of the speech-recognition output on the How2 test data is 32.9%.

4.3 Baseline Models

We compare our model with the following baseline models of single or multiple modalities:

S2S (Luong et al., 2015): a standard sequence-to-sequence architecture using an RNN encoder-decoder with a global attention mechanism.

PG (See et al., 2017): a commonly used encoder-decoder summarization model with attention (Bahdanau et al., 2015), which combines copying words from source documents and outputting words from a vocabulary.

FT: a strong baseline that applies a transformer-based encoder-decoder model to a flat sequence.

VideoRNN (Palaskar et al., 2019): a baseline of the video-only model implemented on the How2 dataset.

MT (Zhou et al., 2018): a transformer-based encoder-decoder architecture receiving sequence features of video for end-to-end dense video captions.

HA (RNN/Transformer) (Palaskar et al., 2019): a multisource sequence-to-sequence model with a hierarchical attention approach to combine textual and visual modalities, which is currently the state-of-the-art method for the multimodal summarization task on the How2 dataset.

4.4 Implement Details

For the RNN-based models, we uniformly use a 2-layer GRU with 128-dimensional word embeddings and 256-dimensional hidden states for each direction. We truncate the maximum text sequence length to 600.

For the transformer-based models, we uniformly use a 4-layer transformer of 512 dimensions with 8 heads. We truncate the maximum text sequence length to 800, and the maximum video sequence length to 1024.

For both the two architectures, we use the cross-entropy loss and Adam optimizer (Kingma and Ba, 2015). The initial learning rate is set to $1.5e^{-4}$. All trainable parameters are randomly initialized with the Kaiming initialization (He et al., 2015). The training of the proposed models are conducted on $\{1, 2\}$ GeForce RTX 2080 Ti GPUs for 50 epochs with a batch size of $\{4, 16\}$. During decoding for prediction, we use beam search with a beam size of 6 and a length penalty with $\alpha = 1$ (Wu et al., 2016).

For a fair comparison, following Palaskar et al. (2019), all the methods take the same 2048-dimensional video features extracted from a ResNeXt-101 3D convolutional neural network (Hara et al., 2018) as input; the vocabulary is built based on the How2 data, and do not use pre-trained word embeddings.

5 Results and Analysis

5.1 Model Performance

We adopt multiple automatic metrics to comprehensively evaluate model performance: BLEU (1,2,3,4) (Papineni et al., 2002), ROUGE (1,2,L) (Lin, 2004), METEOR (Banerjee and Lavie, 2005) and CIDEr (Vedantam et al., 2015). Table 2 shows the results for different models on the How2 dataset. Table 3 shows the model performances of using automatic transcripts obtained from a speech recognition system instead of ground-truth transcripts provided by the dataset. The results show that our proposed model achieves the state-of-the-art performance in each evaluation metric on both the RNN-based and transformer-based models. It can also be seen that the performances of the pure video modality models are modest because of the frozen video features extracted from a task-independent pretraining model.

In particular, Table 3 shows that when the performances of all the prior models trained with ASR-

Modality	Method	B-1	B-2	B-3	B-4	R-1	R-2	R-L	M	C
Ground-truth transcript	S2S	0.552	0.456	0.399	0.358	0.586	0.406	0.538	0.276	2.349
	PG	0.553	0.456	0.398	0.357	0.572	0.395	0.528	0.268	2.134
	FT	0.566	0.467	0.408	0.366	0.590	0.410	0.543	0.277	2.296
Video	VideoRNN	0.441	0.329	0.269	0.227	0.465	0.262	0.415	0.199	1.149
	MT	0.496	0.384	0.329	0.274	0.519	0.320	0.468	0.229	1.461
Ground-truth transcript+Video	HA (RNN)	0.572	0.477	0.418	0.375	0.603	0.425	0.557	0.288	2.476
	HA (Trm)	0.586	0.483	0.433	0.381	0.602	0.431	0.559	0.289	2.512
	Proposed (RNN)	0.591	0.504	0.451	0.411	**0.623**	**0.461**	**0.582**	**0.301**	**2.690**
	Proposed (Trm)	**0.600**	**0.509**	**0.453**	**0.413**	0.616	0.451	0.574	0.299	2.671

Table 2: Results on the How2 test set. The proposed approach achieves better performance in each evaluation metric with $p < 0.01$ under t-test. B: BLEU; R: ROUGE; M: METEOR; C: CIDEr.

Modality	Method	B-1	B-2	B-3	B-4	R-1	R-2	R-L	M	C
ASR-output transcript	S2S	0.467	0.351	0.287	0.242 (↓0.116)	0.481	0.282	0.434 (↓0.104)	0.214	1.319
	FT	0.498	0.384	0.320	0.276 (↓0.090)	0.511	0.310	0.458 (↓0.085)	0.228	1.551
ASR-output transcript+Video	HA (RNN)	0.517	0.408	0.345	0.301 (↓0.074)	0.539	0.342	0.487 (↓0.070)	0.246	1.729
	HA (Trm)	0.531	0.425	0.364	0.321 (↓0.060)	0.551	0.360	0.501 (↓0.058)	0.255	1.918
	Proposed (RNN)	0.570	0.482	0.425	0.384 (↓0.027)	**0.600**	**0.436**	**0.561 (↓0.021)**	**0.285**	**2.447**
	Proposed (Trm)	**0.578**	**0.482**	**0.428**	**0.390 (↓0.023)**	0.593	0.421	0.550 (↓0.024)	0.282	2.346

Table 3: Results on the How2 test set. The ASR-output transcripts is used to replace the provided ground-truth transcripts. The down arrow (↓) indicates the performance degradation when using ASR-output transcript to replace ground-truth transcript under the same model.

Archiecture	No.	Method	B-1	B-2	B-3	B-4	R-1	R-2	R-L	M	C
RNN	1a	T2VF	0.549	0.448	0.389	0.347	0.572	0.389	0.523	0.265	2.119
	2a	T2VF+FFG	0.573	0.484	0.428	0.388	0.610	0.439	0.564	0.288	2.442
	3a	V2TF	0.570	0.482	0.429	0.390	0.599	0.436	0.560	0.283	2.416
	4a	V2TF+FFG	0.573	0.485	0.432	0.393	0.603	0.442	0.563	0.285	2.458
	5a	T2VF+V2TF+HFD	0.571	0.481	0.427	0.387	0.601	0.435	0.560	0.282	2.426
	6a	T2VF+V2TF+HFD+FFG (full)	**0.591**	**0.504**	**0.451**	**0.411**	**0.623**	**0.461**	**0.582**	**0.301**	**2.690**
Transformer	1b	T2VF	0.587	0.492	0.436	0.395	0.606	0.436	0.563	0.291	2.538
	2b	T2VF+FFG	0.593	0.501	0.446	0.407	0.612	0.448	0.571	0.293	2.63
	3b	V2TF	0.577	0.477	0.418	0.379	0.596	0.418	0.552	0.284	2.439
	4b	V2TF+FFG	0.579	0.481	0.422	0.381	0.598	0.421	0.554	0.285	2.456
	5b	T2VF+V2TF+HFD	0.592	0.497	0.440	0.398	0.606	0.437	0.562	0.290	2.591
	6b	T2VF+V2TF+HFD+FFG (full)	**0.600**	**0.509**	**0.453**	**0.413**	**0.616**	**0.451**	**0.574**	**0.299**	**2.671**

Table 4: Ablation analysis on the How2 test set. T2VF: transcript-to-video fusion; V2TF: video-to-transcript fusion; HFD: hierarchical fusion decoder; FFG: fusion forget gate.

No.	Method (On RNN)	B-4	R-L
1	T2VF	0.301	0.483
2	T2VF+FFG	0.370	0.547
3	V2TF	0.353	0.528
4	V2TF+FFG	0.362	0.534
5	T2VF+V2TF+HFD	0.347	0.525
6	T2VF+V2TF+HFD+FFG (full)	**0.384**	**0.561**

Table 5: Ablation analysis on RNN-based models. The ASR-output transcripts is used to replace the provided ground-truth transcripts.

Full Model	setting	B-4	R-L
RNN	2-layers	0.411	0.582
	+ FFG on HFD (2-layers)	0.405	0.574
	3-layers	0.410	0.582
Trm	4-layers	0.413	0.574
	+ FFG on HFD (4-layers)	0.410	0.571
	6-layers	0.410	0.574

Table 6: Ablation analysis on the How2 test set.

Ground-truth Transcript: alvin dedeux : first thing you have to do is attach the thread to the hook , and what you want to do is secure it . i usually just lay it across in front of the hook and then wrap backwards that way just enough to catch that standing piece of the thread there . three or four wraps is usually good and then you can just , depending on what you 're doing , you can leave it hanging or you could clip it off close there . but now , my thread is not going to come loose so i 'm ready to start attaching my other materials . sometimes you can go ahead and wrap it all the way back , just make sure you got it on there secure , wrap it back the other way and then start your tying . or if you want to start tying back here , you 'd wrap it back here and keep it back here . but the trick is to just make those first couple of wraps , trap that thread and then go back this way a few times . and then either continue back to the back of the hook or up to the front . and that gives you a good solid foundation to start tying your fly .

ASR-output Transcript: first thing you have to do is attach the thread to the hook . what do you want to do as security . i suggest . lacrosse and fatherhood . and then wrap . backwards that way . just enough to catch . that . standing . piece of the . trader . therefore raps is usually good . and then . you can just depends on what you doing you can leave it hanging out you can clip it off close there . but you're now . my thread is not good . come loose . some radio star attachment other materials . sometimes you can go in wrap it all the way back . no just make sure you get it on there secure . rabbits back the other way . and then start retiring or if you want to start . start time back here grab it back and keep it back . but the trick is a just make those first couple of laps trap that . then go back this way few times . and then either continue back to the back of the head . turn up to the front . and um . the . gives a good song . foundation to start time

Summary: watch and learn how to tie thread to a hook to help with fly tying as explained by out expert in this free how-to video on fly tying tips and techniques .

Figure 5: A example taken from How2 test set. For the extracted ASR-output transcripts, we use the period "." as the separator of the automatically segmented audio clips.

output transcripts drop sharply due to the high error rate ($WER = 32.9\%$) of speech recognition, our model still has good performance close to the models trained with ground-truth transcripts. In using ASR-output transcripts, our framework outperforms the HA 8.3 BLEU-4 points, 7.4 ROUGE-L points, 3.9 METEOR points, and 71.8 CIDEr points on the RNN-based architecture, and 6.9 BLEU-4 points, 4.9 ROUGE-L points, 2.7 METEOR points, and 42.8 CIDEr points on the transformer-based architecture, which fully shows the effectiveness of our approach.

5.2 Ablations

The purpose of this study is to examine the role of the proposed multistage fusion and fusion forget gate (FFG). We divide the fusion process into transcript-to-video-fusion (T2VF) and video-to-transcript fusion (V2TF) in the cross fusion block, the following FFG, and the final HFD, and retrain our approach by ablating one or more of them.

- We retrain only T2VF and only V2TF and replace HFD with a standard decoder to handle single-source multimodal encodings.
- We add the FFG to the above T2VF and V2TF models separately.
- We retain T2VF, V2TF, HFD, and remove all the FFG of the full model.

Table 4 lists the results on the How2 dataset. We can observe that: 1) except that the V2TF's performance is weaker than the single-text modality on RNN {1a}, the performances of all the V2TF and T2VF models {3a, 1b, 3b} exceed the performances of the single-modality models. 2) Compared with using only V2TF or T2VF, using V2TF and T2VF together with HFD {5a, 5b} further improves the model effect. 3) When FFG is added, the performances of all the fusion structures improve, which is particularly evident in the RNN-based models. 4) Only one-way fusion structures with FFG {2a,4a,2b,4b} can achieve comparable and even better performance compared to the HA. These results demonstrate the effectiveness of the multistage fusion and inside FFG.

Table 5 lists the results of using the ASR-output transcript instead of the provided ground-truth transcript. The observation results are similar to those observed in Table 4. In particular, we can see a greater increase in the performance of the FFG when using high noise ASR-output trancript compared to using the ground-truth transcript. This further verifies the ability of FFG to the resist the flow of multimodal noise.

Additionally, we also evaluate 1) the effect of model depth and 2) the effect of FFG on HFD. We deepen the model depth, and apply FFG to the multimodal context representation generated by the AoMA in HFD. The results in Table 6 indicate that the two measures do not improve model performance.

Modality	Method	R-L	Output
-	Reference	-	watch and learn how to tie thread to a hook to help with fly tying as explained by out expert in this free how-to video on fly tying tips and techniques .
Ground-truth transcript	FT	0.543	learn about attaching the thread in fly tying and other fly fishing tips in this free how-to video on fly tying tips and techniques .
Video	MT	0.468	learn how to attach a backing tail to fly fishing backing in this free how-to video on fly tying and techniques .
Ground-truth transcript+Video	HA (RNN)	0.557	learn from our expert how to attach a hook to fly tying in this free how-to video on fly tying tips and techniques .
	HA (Trm)	0.559	learn about using a bobbin in fly tying from our expert in this free how-to video on techniques for and making fly tying nymphs .
	Proposed (RNN)	**0.582**	watch and learn from an expert how to attach the thread to fly tying in this free how-to video on fly tying tips and techniques .
	Proposed (Trm)	0.574	learn some great tips on attaching the thread to the fly fishing in this free how-to video on fly tying tips and techniques .
ASR-output transcript+Video	HA (RNN)	0.487	tying a knot for fly fishing is easy with these tips , get expert advice on woodworking in this free video .
	HA (Trm)	0.501	tying a knot onto a knot , make sure the snap is secure and connected to the hoop knot . attach a french braid to a knot with tips from an experienced handyman in this free video on fly tying .
	Proposed (RNN)	**0.561**	watch and learn from our expert on fly fishing tips in this free how-to video on fly tying tips and techniques .
	Proposed (Trm)	0.550	learn how to use a wrapped knot to wrap a fly fishing knot in this free how-to video on fly tying tips and techniques .

Table 7: Example outputs from different models.

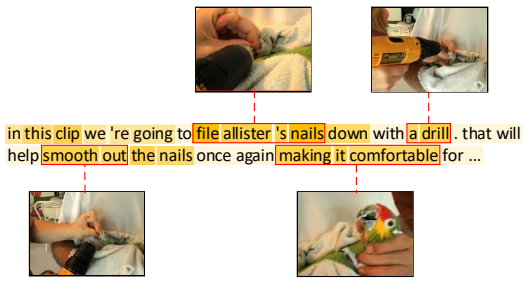

Figure 6: A visualization of FFG and attention in CFG.

5.3 Qualitative Analysis

We provide some example outputs from trained models. The example is taken from the How2 test set, and we show its ground-truth transcript and the extracted ASR-output transcript in Figure 5. Table 7 lists the generated results. We can observe that: 1) compared to single-modality models, the multi-modality models can generate more accurate and fluent contents. 2) In using ground-truth transcript, both HA and our proposed model generate accurate and fluent summaries. 3) In using ASR-output transcripts, our proposed model still generates a relatively accurate summary while the content generated by HA is not accurate enough, which intuitively illustrates the advantage of our model in the absence of ground-truth transcripts.

To better understand what our model has learned, we take the sample shown in Figure 1 to visualize the FFG and cross-attention in CFG. We sum the FFG weights and use the color depth of the word to represent the intensity of the FFG of controlling the flow of video to text, and demonstrate the interaction between video and text by displaying the video frame with the highest transcript-to-video attention when generating adaptive video streams. As shown in Figure 6, in the input segment, we can observe the following: 1) For some words related to the summary such as “file”, “nails”, the FFG retains video streams for it, in contrast, for words such as “once again”, the FFG forgets most of the video information. 2) For the words that FFG remembers deeply, the corresponding video frame has a certain correlation with it, for example, “file allister’s nails” point to a close-up of manicuring the parrot’s nails.

6 Conclusions

We introduce a multistage fusion network with fusion forget gate for generating text summaries for the open-domain videos. We propose a multistep fusion schema to model fine-grained interactions between multisource modalities and a fusion forget gate module to handle the flow of multimodal noise of multisource long sequences. Experiments on the How2 dataset show the effectiveness of the proposed models. Furthermore, when using high noise speech recognition transcription, our model still achieves the effect of being close to the ground-truth transcription model, which reduces the manual annotation cost of transcripts.

References

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision (CVPR)*, pages 2425–2433.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Satanjeev Banerjee and Alon Lavie. 2005. 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*, pages 65–72.

Jingwen Bian, Yang Yang, and Tat-Seng Chua. 2013. Multimedia summarization for trending topics in microblogs. In *Proceedings of the ACM international conference on Conference on information & knowledge management*, pages 1807–1812. ACM.

Jingwen Bian, Yang Yang, Hanwang Zhang, and Tat-Seng Chua. 2014. Multimedia summarization for social events in microblog stream. *IEEE Transactions on multimedia*, 17(2):216–228.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734.

D Elliott, S Frank, and E Hasler. 2015. Multi-language image description with neural sequence models. corr. *arXiv preprint arXiv:1510.04709*.

G. Evangelopoulos, A. Zlatintsi, A. Potamianos, P. Maragos, K. Rapantzikos, G. Skoumas, and Y. Avrithis. 2013. Multimodal saliency and fusion for movie summarization based on aural, visual, and textual attention. *IEEE Transactions on Multimedia*, 15(7):1553–1568.

Haoyuan Gao, Junhua Mao, Jie Zhou, Zhiheng Huang, Lei Wang, and Wei Xu. 2015. Are you talking to a machine? dataset and methods for multilingual image question answering. In *Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS)*, pages 2296–2304.

Google-Speech-V2. Google's speech to text api (v2).

Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. 2018. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6546–6555.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision (ICCV)*, pages 1026–1034.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 770–778.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. 2017. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Haoran Li, Junnan Zhu, Tianshang Liu, Jiajun Zhang, and Chengqing Zong. 2018. Multi-modal sentence summarization with modality attention and image filtering. pages 4152–4158.

Haoran Li, Junnan Zhu, Cong Ma, Jiajun Zhang, and Chengqing Zong. 2017. Multi-modal summarization for asynchronous collection of text, image, audio and video. pages 1092–1102.

Jindřich Libovickỳ and Jindřich Helcl. 2017. Attention strategies for multi-source sequence-to-sequence learning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 196–202.

Jindrich Libovickỳ, Shruti Palaskar, Spandana Gella, and Florian Metze. 2018. Multimodal abstractive summarization of open-domain videos. In *Proceedings of the Workshop on Visually Grounded Interaction and Language (ViGIL). NIPS*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. In *Proceedings of the International Conference on Neural Information Processing Systems (NIPS)*, pages 289–297.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1412–1421.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Ranking sentences for extractive summarization with reinforcement learning. pages 1747–1759.

Shruti Palaskar, Jindřich Libovickỳ, Spandana Gella, and Florian Metze. 2019. Multimodal abstractive summarization for how2 videos. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 6587–6596.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the annual meeting on association for computational linguistics (ACL)*, pages 311–318. Association for Computational Linguistics.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. pages 379–389.

Ramon Sanabria, Ozan Caglayan, Shruti Palaskar, Desmond Elliott, Loïc Barrault, Lucia Specia, and Florian Metze. 2018. How2: A large-scale dataset for multimodal language understanding.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1073–1083.

Lucia Specia, Stella Frank, Khalil Sima'An, and Desmond Elliott. 2016. A shared task on multimodal machine translation and crosslingual image description. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the International Conference on Neural Information Processing Systems (NIPS)*, pages 6000–6010. Curran Associates Inc.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 4566–4575.

Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2015a. Sequence to sequence-video to text. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 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 *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1494–1504.

Dingding Wang, Tao Li, and Mitsunori Ogihara. 2012. Generating pictorial storylines via minimum-weight connected dominating set approximation in multi-view graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 683–689.

William Yang Wang, Yashar Mehdad, Dragomir R Radev, and Amanda Stent. 2016. A low-rank approximation approach to learning joint embeddings of news stories and images for timeline summarization. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 58–68.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. 2018. End-to-end dense video captioning with masked transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8739–8748.

Junnan Zhu, Haoran Li, Tianshang Liu, Yu Zhou, Jiajun Zhang, and Chengqing Zong. 2018. Msmo: Multimodal summarization with multimodal output. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4154–4164.

Junnan Zhu, Yu Zhou, Jiajun Zhang, Haoran Li, Chengqing Zong, and Changliang Li. 2020. Multimodal summarization with guidance of multimodal reference. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.

A Appendices

A.1 Hierarchical Fusion Decoder

In this section, The formula expression and model diagram of RNN-based and Transformer-based decoder are illustrated. The structures are shown in Figure 7.

RNN-based HFD. At each decoding time step, an unidirectional GRU receives the target token embeddings x_t and previous hidden state h_{t-1} to

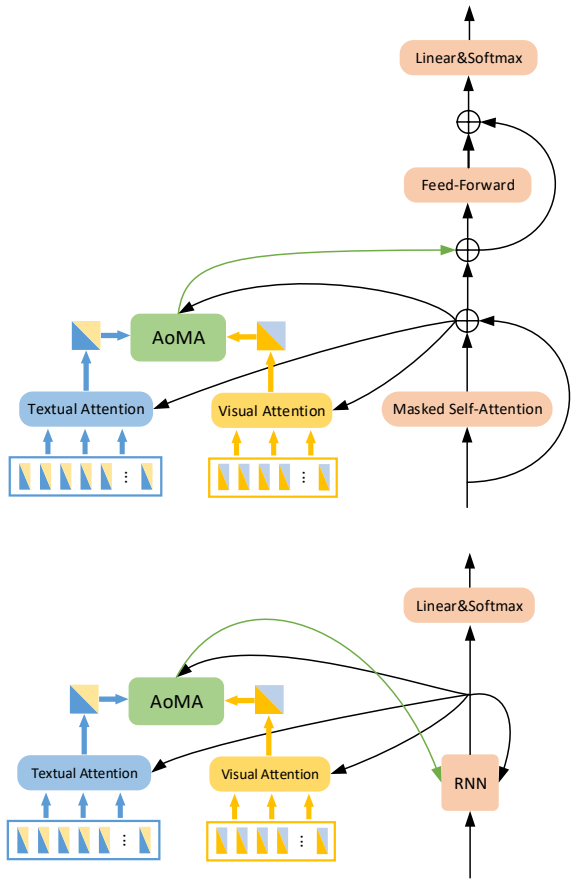

Figure 7: Transformer-based decoder is above, and RNN-based decoder is below.

compute a new hidden state h_t, which is defined as:

$$h_t = GRU(x_t, h_{t-1}) \tag{17}$$

The context vectors of each modality are firstly calculated by:

$$C_V = Attn_{MLP}(h_t, V_F) \tag{18}$$

$$C_T = Attn_{MLP}(h_t, T_F) \tag{19}$$

We adopt an MLP attention for RNN-based methods. Then the second attention AoMA over the video context vectors C_V and text context vectors C_T are implemented as:

$$\begin{aligned} C_c =& AoMA(h_t, C_V, C_T) \\ =& softmax(W_1 \tanh(W_2 h_t + \\ & W_3[C_V; V_T])) \cdot [C_V; V_T] \end{aligned} \tag{20}$$

The context vector C_C of multimodal fusion and the decoder state h_t are merged to get the output state y_{t+1}:

$$y_{t+1} = tanh(W[h_t; C_C] + b) \tag{21}$$

where W_1, W_2, W_3, W and b are trainable parameters.

Transformer-based HFD. Transformer-based HFD has a similar strategy as RNN-based. We mainly introduce how it absorbs multimodal information. It firstly receives target token embeddings x_t through the masked multi-head self-attention and residual connection to obtain the hidden state vector h_t, denoted as:

$$h_t = MHA_{masked}(x_t) \tag{22}$$

Then h_t is transformed into a query, separately attends to a set of key and value pairs mapped by previous encodings of each modality by the multi-head encoder-decoder attention, denoted as:

$$C_V = MHA(h_t, V_F) \tag{23}$$

$$C_T = MHA(h_t, T_F) \tag{24}$$

Similarly, the generated multimodal context vectors are fused by AoMA:

$$C_c = AoMA(h_t, C_V, C_T) \tag{25}$$

The final output state reaches through the feed-forward and add&norm layer like the general transformer, calculated as the following equation:

$$y_{t+1} = W_2 ReLu(W_1(C_c + h_t) + b_1) + b_2 + C_c + h_t \tag{26}$$

where W_1, W_2, b_1 and b_2 are trainable parameters.

A.2 Evalution Metrics

We use the nmtpytorch evaluation library https://github.com/lium-lst/nmtpytorch suggested by the How2 Challenge, which includes BLEU (1, 2, 3, 4), ROUGE-L, METEOR, and CIDEr evaluation metrics. As an alternative, nlg-eval https://github.com/Maluuba/nlg-eval can obtain the same evaluation score as nmtpytorch.

In addition, we also use a ROUGE evaluation library https://github.com/neural-dialogue-metrics/rouge, which supports the evaluation of ROUGE series metrics (ROUGE-N, ROUGE-L and ROUGE-W).

A.3 Data

The extracted ASR-output transcript data is available on https://github.com/forkarinda/MFN.