(음성 바로→ 요약) (음성 → 텍스트 → 요약)
⇒ END-to-END 모델이 Cascade 모델 보다 좋다.

END-TO-END SPEECH SUMMARIZATION USING RESTRICTED SELF-ATTENTION

Roshan Sharma, Shruti Palaskar, Alan W Black and Florian Metze

Carnegie Mellon University
Pittsburgh PA,USA
roshansh@andrew.cmu.edu, spalaska@andrew.cmu.edu

arXiv:2110.06263v2 [cs.CL] 24 Jan 2022

ABSTRACT

Speech summarization is typically performed by using a cascade of speech recognition and text summarization models. End-to-end modeling of speech summarization models is challenging due to memory and compute constraints arising from long input audio sequences. Recent work in document summarization has inspired methods to reduce the complexity of self-attentions, which enables transformer models to handle long sequences. In this work, we introduce a single model optimized end-to-end for speech summarization. We apply the restricted self-atention technique from text-based models to speech models to address the memory and compute constraints. We demonstrate that the proposed model learns to directly summarize speech for the How-2 corpus of instructional videos. The proposed end-to-end model outperforms the previously proposed cascaded model by 3 points absolute on ROUGE. Further we consider the spoken language understanding task of predicting concepts from speech inputs, and show that the proposed end-to-end model outperforms the cascade model by 4 points absolute F-1.

***Index Terms*—** speech summarization, end-to-end , long sequence modeling, concept learning

1. INTRODUCTION

Summarization extracts and condenses desired information from the inputs, often text. Text can be summarized using abstraction or extraction [1]. Abstractive Text Summarization (ATS), generates a novel and concise summary of the input text. Abstractive summarization can be performed on multiple modalities [2, 3].

Speech Summarization is performed using a cascade of Automatic Speech Recognition (ASR) followed by Abstractive Text Summarization (ATS) [4, 5, 6]. [7] proposed an alternative cascade formulation- ASR followed by Concept Extraction and Summarization.They showed that specific and abstract concepts are useful as intermediate representations for multimodal summarization. However, cascade architectures makes the model structure complicated, and errors in the ASR degrade summarization performance. Therefore, we propose a single sequence model optimized end-to-end (E2E) for speech summarization.

Speech summarization involves very long input sequences. The prohibitive quadratic computational cost of self-attention makes standard transformer models unsuitable for longer sequences. To address this, [8] uses segment-wise recurrence within transformer self-attention to provide longer context, and [9] compresses the segment level contexts and provides them as additional input to enable a longer context. Other works have focused on making the self-attention sparse to reduce computational complexity. . Reformer [10] uses Locality Sensitive Hashing to compute localized self-attention in O(n.logn) and ETC [11] uses efficient global-local attention to scale to longer sequences. To reduce the complexity of self-attention to O(n), Linformer [12] uses a low-rank factorization of the self-attention matrix, and Big Bird [13] uses a combination of sliding window, global and random attention. Longformer [14] uses different attention patterns for each layer and restricted dilated self-attention with task-specific global attention. These long sequence techniques have been evaluated on text inputs, where the input sequence lengths are often several hundred times smaller than sequence lengths of video-level speech (see Table 1).

Abstractive speech summarization uses the complete long sequence context to generate a summary. Long context ASR has been explored by training on longer sequences [15, 16, 17] or by passing context across utterances [18]. [15] trains contextual models for joint ASR-diarization by concatenating turns from doctor-patient conversations. [16, 17] trains long context transformer ASR models by concatenating text and audio from previous utterances as input to decode the current utterance. Longer segments produce larger WER improvements and using both acoustic and lexical context is shown to be important. In this work, we

1. introduce a way to directly model speech summarization as an end-to-end task
2. demonstrate the effectiveness of restricted self-attention for speech inputs, critical for the success of end-to-end speech summarization, and
3. show that such an end-to-end model can also be applied

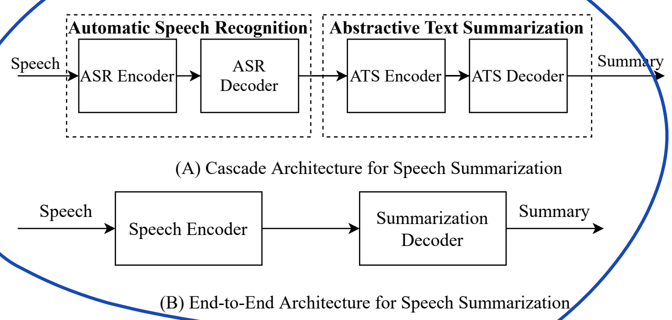

Fig. 1. Speech Summarization: Cascade and End-to-End Model Architectures

to learn concepts directly from speech inputs, a potential spoken language understanding task.

2. BACKGROUND

2.1. Cascade and E2E Modeling

Speech summarization can be modeled as a cascade of speech recognition and text summarization [4, 5]. Figure 1 shows the cascade and end-to-end approaches to speech summarization. The cascade approach benefits from strong ASR models pre-trained on large amounts of speech and summarization models like BART [19] trained on large amounts of text data. However, errors in ASR are compounded due to the cascade architecture, which serves as motivation for direct end-to-end modeling.

2.2. Concept Learning

Cascaded learning was recently shown as one of the ways to accomplish speech summarization [7]. In [7], authors propose multimodal speech summarization via semantic concept learning, where speech is first represented as a sequence of concepts, followed by a concept to summarization model that generates abstractive summaries. Concepts are abstractive representations of the input speech (and video) that can be used as anchoring points between speech and the summary. Domain-specific noun phrases are automatically extracted and used as abstract concepts (extracted from the human-annotated summaries for the training data). Their model is a pipeline model that does not attempt end-to-end generation of summaries given input speech. In this work, we attempt to model speech summarization as an end-to-end task and also evaluate the benefits of concept learning for such end-to-end speech summarization.

Table 1. Statistics of the How-2 2000h Dataset used for model training and evaluation. The mean and maximum statistics of N- the input length in frames, and L- the output length (in tokens) is shown.

Set	Max N	Mean N	Mean L	Max L
Train	145,082	9,806.58	60.54	173
Test	39,537	9,866.55	60.29	152

3. PROPOSED APPROACH

3.1. Restricted Self-Attentions

Different from other speech tasks like Automatic Speech Recognition or Speech Synthesis, the speech inputs for summarization are much longer. Table 1 shows the average and maximum frame lengths of input speech, and output token lengths for summarization. The input speech segments measure over 100s, whereas typical ASR utterance lengths are around 5s. The high computational complexity makes it intractable to train video-level speech models on a 32 GB GPU. Consider N is the length of the input speech sequence, and L is the length of the output token sequence($N >> L$, refer Table 1 for values) . Then, there are three attentions in the sequence model: encoder self-attention with a computational complexity of $O(N^2)$, decoder self-attention with a complexity of $O(L^2)$, and encoder-decoder source-target attention with a complexity of $O(NL)$. In order to make end-to-end training possible, the computational complexity of the encoder self-attention needs to be reduced.

Inspired by [14, 20], we break down the self-attention computation into fixed sized context windows of size W. For each sequence element, a surrounding context of width $W/2$ on each side is considered while computing the self-attention result. The number of such windows required will be $P = N/W$, and the cost of the encoder-self attention is now reduced to $O(PW^2)$, which is smaller than $O(N^2)$. To further reduce the computational complexity, we can drop one element for every D elements, i.e., dilation. Dilation further reduces the complexity to $O(P(W/D)^2)$.

3.2. End-to-End Speech Summarization

Given input speech frames for an *entire* video, we propose to directly summarize it into short, abstractive, textual summaries. The objective of mapping long speech frames (details in Table 1) onto significantly shorter textual tokens makes this an End-to-End Speech Summarization task. As training summarization models from scratch is challenging, we pretrain the sequence model using ASR. Then, the encoder-decoder model is fine-tuned for speech summarization.

3.3. End-to-End Concept Learning

Semantic concepts were shown to be a strong grounding aspect across modalities, especially to bridge the gaps in cascaded speech summarization [7]. Intermediate concept learning can be useful for controllability of generated summaries. Abstract concepts were extracted in [2] by transcribing the videos into text format, and then training a concept extractor. We contend that it would be useful to train a concept extractor from speech end-to-end. As we propose end-to-end speech summarization, we also evaluate the usability of our model to generate concepts directly from speech.

Given input speech at the video-level, we generate abstract semantic concepts as outputs. Sequence of abstract concepts is generated as natural language text. During inference, this model generates such semantic concepts directly from speech, which can be used further for downstream summarization in a cascaded setup.

요약성능이 window 크기에 따라 달라짐,

4. EXPERIMENTAL SETUP

4.1. Dataset and Evaluation

The How-2 Dataset [21] contains 2000h of instructional videos with corresponding text transcripts, video, speech, translations, and summaries.

Two tasks are evaluated: (a) Abstract Concept Generation from Speech, and (b) Abstractive Speech Summarization. Concept Generation is evaluated using Precision, Recall, and F-1 score. Summarization is evaluated using ROUGE [22], METEOR [23], and BERTScore [24]. BERTScore uses pre-trained contextual embeddings and cosine similarity to measure similarity between reference and hypothesis for text generation tasks.

4.2. Model Details

The ASR models are trained using ESPNet [25]. The speech encoder has 2x convolutional subsampling followed by 12 encoder layers, each with feed-forward dimension of 2048, and 8 attention heads. The transformer decoder has 6 layers, each with feed-forward dimension 512 and 4 attention heads. ASR models are trained with joint Connectionist Temporal Classification (CTC)-Attention [26] with the weight for CTC training set to 0.3. Adam optimizer is used with a peak learning rate of 0.002 in 25k training steps. The videos are trimmed to 100s for the video-level speech tasks owing to compute constraints. Batches of 20,000 frames/GPU are constructed for video-level training. Specaugment [27] is used during model training and fine-tuning. We use 40-dimension filterbank and 3-dimensional pitch features for training all models.

The Huggingface transformers library [28] is used to fine-tune text-only models. `BART-large` and `BART-base` [19] are used to finetune the ATS model in the cascade approach, and the decoder from the model is used for E2E training of the speech summarization model.

Table 2. Word Error Rate (WER) (%) for Test and Held Test sets of the 2000h How-to Corpus. Window Size of 20 is used for Restricted Self-Attention

Encoder	Decoder	Test WER (%)
Transformer	Transformer	10.2
Conformer	Transformer	**9.1**
+ Restricted Self-Attention	Transformer	9.3

Table 3. Effect of Window Size and Dilation in Self-Attention of the Speech Encoder on E2E Summarization Model Training. W is the Window Size, and D is the dilation factor (Section 3 for details).

W	D	ROUGE-L	METEOR	BERTScore
20	✗	52.0	26.5	90.5
40	✗	**53.1**	**27.3**	**90.6**
60	✗	52.5	27.1	90.5
100	5	51.9	26.3	90.5

참조문장과 생성문장을 각각 BERT에 넣어 문맥 벡터를 구한후 코사인 유사도 이용

LCS기법을 이용 최장길이로 매칭되는부분열

5. RESULTS AND DISCUSSION

5.1. Speech Recognition

The E2E Speech Summarization model is pre-trained for ASR. Table 2 shows the Word Error Rate (WER) for different encoder-decoder combinations. The use of a conformer [29] model improves ASR results by over 1 % absolute compared to the transformer. The use of restricted self-attention results in a slight decrease in performance.

5.2. Speech Summarization

Table 4 highlights summarization results on three types of models: ground-truth text-based models (considered the topline scores), ASR-based Cascade models, and direct E2E models. Cascade models use the best ASR model from Table 2, i.e., a conformer encoder and transformer decoder. `BART-large` and `BART-base` [19] are fine-tuned on ground-truth and ASR predicted text to establish the topline and cascade baselines. `BART-large` outperforms `BART-base` in ROUGE, METEOR, and BERT Scores among the topline and cascade models. Conformer ASR coupled with BART leads to strong cascade models that outperform previous works.

The best ASR model is finetuned on the summarization data to build the E2E model with conformer encoder and transformer decoder. The E2E model outperforms the best cascade model on all metrics with 4x fewer parameters, indicating that the end-to-end model is able to produce more fluent, semantically similar summaries. It is interesting to

Table 4. Summarization Performance of Topline, Cascade and E2E Models using automatic (ROUGE and METEOR) and semantic evaluation metrics (BERTScore).

	Model	Parameters	ROUGE-1	ROUGE-2	ROUGE-L	METEOR	BERTScore
Topline	Groundtruth Text						
	+ `BART-large` Summarization	400M	**61.8**	42.8	55.5	30.0	**91.0**
	+ `BART-base` Summarization	140M	60.6	40.4	53.7	27.7	90.7
Cascade	Conformer ASR	107M					
	+ `BART-large` Summarization	400M	59.2	38.8	52.3	27.8	90.6
	+ `BART-base` Summarization	140M	57.6	36.3	50.3	25.6	90.3
	S2S- PredText2Summary[2]	-	-	-	46.1	22.9	-
	Kaldi ASR + Concept2Summary [7]	-	-	-	51.4	**30.4**	-
E2E	Conformer Encoder						
	+ Transformer Decoder	**104M**	60.9	**43.0**	**55.9**	28.8	**91.0**

Table 5. Evaluation of Baseline and Proposed Concept Learning Models using Recall, Precision and F-1 Score

Model	Precision	Recall	F-1
Predicted Text2Concept [7]	52.5	57.3	54.8
Speech2Concept	**62.3**	55.8	**58.8**

note that the E2E model performs nearly as well the best topline model, indicating that the task of speech summarization can be performed just as well without text or transcribed speech.

5.3. Window Size and Dilation

To understand the impact of context window size on summarization performance, we train models with different window sizes using a subset of the training data. This subset consists of about 65 % of the full training data, and untrimmed videos (with video length $<=$ 100s). Then the model is evaluated on the full test set. Table 3 shows that summarization performance does depend on window size. A window size of $W = 40$ seems to yield the best ROUGE-L scores, while a smaller window of $W = 20$ yields a lower ROUGE-L score. From the first and last row, dilation reduces the computational complexity significantly while retaining similar performance.

5.4. Concept Learning

Table 5 evaluates the end-to-end concept learning model. Concepts being non-sequential text, we evaluate on Precision, Recall, and F1. The baseline is a cascade of two modules-ASR and *predicted* Text2Concept model, and achieves an F1 of 54.8 [7]. The proposed end-to-end Speech2Concept model achieves 57.2, a significant performance given the model sees no input text.

6. CONCLUSION

In this paper, we model speech summarization as a direct end-to-end task starting from input speech at a video-level and generating abstractive summaries as the output. We address the long speech input frames problem by applying restricted self-attention to help us achieve this task without running into severe memory and compute bottlenecks. Our approach at least outperforms a strong text-based summarization model, and at best, demonstrates strong performance compared to previous approaches to speech summarization (cascaded pipeline models). We also demonstrate the effects of various window size and dilations on summarization, concluding that larger window sizes are crucial for better models. Using restricted self-attention and a Conformer based speech recognizer, we achieve a competitive result on speech recognition on the commonly used How2 dataset. Finally, we demonstrate the potential of such end-to-end modeling on a Speech2Concept task that could be useful for downstream summarization as well as other speech-based tasks that earlier represented speech by predicted text from an automatic speech recognizer.

7. ACKNOWLEDGEMENTS

This work used the Bridges system, which is supported by NSF award number ACI-1445606, at the Pittsburgh Supercomputing Center (PSC).

8. REFERENCES

[1] U. Hahn and I. Mani, "The challenges of automatic summarization," *Computer*, vol. 33, no. 11, pp. 29–36, 2000.

[2] Shruti Palaskar, Jindřich Libovický, Spandana Gella, and Florian Metze, "Multimodal abstractive summarization for how2 videos," in *Proceedings of the 57th Annual Meeting of the As-*

sociation for Computational Linguistics, Florence, Italy, 2019, pp. 6587–6596, Association for Computational Linguistics.

[3] Tiezheng Yu, Wenliang Dai, Zihan Liu, and Pascale Fung, "Vision guided generative pre-trained language models for multimodal abstractive summarization," 2021.

[4] Chenguang Zhu, Ruochen Xu, Michael Zeng, and Xuedong Huang, "A hierarchical network for abstractive meeting summarization with cross-domain pretraining," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, Online, 2020, pp. 194–203, Association for Computational Linguistics.

[5] Dana Rezazadegan, Shlomo Berkovsky, Juan C Quiroz, A Baki Kocaballi, Ying Wang, Liliana Laranjo, and Enrico Coiera, "Automatic speech summarisation: A scoping review," *arXiv preprint arXiv:2008.11897*, 2020.

[6] Potsawee Manakul, Mark Gales, and Linlin Wang, "Abstractive spoken document summarization using hierarchical model with multi-stage attention diversity optimization," 2020.

[7] Shruti Palaskar, Ruslan Salakhutdinov, Alan W. Black, and Florian Metze, "Multimodal Speech Summarization Through Semantic Concept Learning," in *Proc. Interspeech 2021*, 2021, pp. 791–795.

[8] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov, "Transformer-XL: Attentive language models beyond a fixed-length context," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, 2019, pp. 2978–2988, Association for Computational Linguistics.

[9] Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap, "Compressive transformers for long-range sequence modelling," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. 2020, OpenReview.net.

[10] Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya, "Reformer: The efficient transformer," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. 2020, OpenReview.net.

[11] Joshua Ainslie, Santiago Ontanon, Chris Alberti, Vaclav Cvicek, Zachary Fisher, Philip Pham, Anirudh Ravula, Sumit Sanghai, Qifan Wang, and Li Yang, "ETC: Encoding long and structured inputs in transformers," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Online, 2020, pp. 268–284, Association for Computational Linguistics.

[12] Sinong Wang, Belinda Li, Madian Khabsa, Han Fang, and Hao Ma, "Linformer: Self-attention with linear complexity," *arXiv preprint arXiv:2006.04768*, 2020.

[13] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed, "Big bird: Transformers for longer sequences," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds. 2020, vol. 33, pp. 17283–17297, Curran Associates, Inc.

[14] Iz Beltagy, Matthew E. Peters, and Arman Cohan, "Longformer: The long-document transformer," *CoRR*, vol. abs/2004.05150, 2020.

[15] Laurent El Shafey, Hagen Soltau, and Izhak Shafran, "Joint Speech Recognition and Speaker Diarization via Sequence Transduction," in *Proc. Interspeech 2019*, 2019, pp. 396–400.

[16] Takaaki Hori, Niko Moritz, Chiori Hori, and Jonathan Le Roux, "Transformer-based long-context end-to-end speech recognition," in *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, Helen Meng, Bo Xu, and Thomas Fang Zheng, Eds. 2020, pp. 5011–5015, ISCA.

[17] Takaaki Hori, Niko Moritz, Chiori Hori, and Jonathan Le Roux, "Advanced long-context end-to-end speech recognition using context-expanded transformers," 2021.

[18] Ryo Masumura, Naoki Makishima, Mana Ihori, Akihiko Takashima, Tomohiro Tanaka, and Shota Orihashi, "Hierarchical transformer-based large-context end-to-end asr with large-context knowledge distillation," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 5879–5883.

[19] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, 2020, pp. 7871–7880, Association for Computational Linguistics.

[20] Niko Moritz, Takaaki Hori, and Jonathan Le Roux, "Capturing multi-resolution context by dilated self-attention," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 5869–5873.

[21] Ramon Sanabria, Ozan Caglayan, Shruti Palaskar, Desmond Elliott, Loïc Barrault, Lucia Specia, and Florian Metze, "How2: a large-scale dataset for multimodal language understanding," *arXiv preprint arXiv:1811.00347*, 2018.

[22] Chin-Yew Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*, Barcelona, Spain, 2004, pp. 74–81, Association for Computational Linguistics.

[23] Michael Denkowski and Alon Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proceedings of the Ninth Workshop on Statistical Machine Translation*, Baltimore, Maryland, USA, 2014, pp. 376–380, Association for Computational Linguistics.

[24] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi, "Bertscore: Evaluating text generation with BERT," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. 2020, OpenReview.net.

[25] Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya

Renduchintala, and Tsubasa Ochiai, "ESPnet: End-to-end speech processing toolkit," in *Proceedings of Interspeech*, 2018, pp. 2207–2211.

[26] Suyoun Kim, Takaaki Hori, and Shinji Watanabe, "Joint ctc-attention based end-to-end speech recognition using multi-task learning," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5-9, 2017*. 2017, pp. 4835–4839, IEEE.

[27] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin Dogus Cubuk, and Quoc V. Le, "Specaugment: A simple augmentation method for automatic speech recognition," in *INTERSPEECH*, 2019.

[28] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Online, 2020, pp. 38–45, Association for Computational Linguistics.

[29] Anmol Gulati, Chung-Cheng Chiu, James Qin, Jiahui Yu, Niki Parmar, Ruoming Pang, Shibo Wang, Wei Han, Yonghui Wu, Yu Zhang, and Zhengdong Zhang, Eds., *Conformer: Convolution-augmented Transformer for Speech Recognition*, 2020.