MAST: Multimodal Abstractive Summarization with Trimodal Hierarchical Attention

Aman Khullar*
IIIT Hyderabad
Hyderabad, India
aman.khullar@iiit.ac.in

Udit Arora*
New York University
New York, NY, USA
uditarora@nyu.edu

Abstract

This paper presents MAST, a new model for Multimodal Abstractive Text Summarization that utilizes information from all three modalities – text, audio and video – in a multimodal video. Prior work on multimodal abstractive text summarization only utilized information from the text and video modalities. We examine the usefulness and challenges of deriving information from the audio modality and present a sequence-to-sequence trimodal hierarchical attention-based model that overcomes these challenges by letting the model pay more attention to the text modality. MAST outperforms the current state of the art model (video-text) by 2.51 points in terms of Content F1 score and 1.00 points in terms of Rouge-L score on the How2 dataset for multimodal language understanding.

1 Introduction

In recent years, there has been a dramatic rise in information access through videos, facilitated by a proportional increase in the number of video-sharing platforms. This has led to an enormous amount of information accessible to help with our day-to-day activities. The accompanying transcripts or the automatic speech-to-text transcripts for these videos present the same information in the textual modality. However, all this information is often lengthy and sometimes incomprehensible because of verbosity. These limitations in user experience and information access are improved upon by the recent advancements in the field of multimodal text summarization.

Multimodal text summarization is the task of condensing this information from the interacting modalities into an output summary. This generated output summary may be unimodal or multimodal (Zhu et al., 2018). The textual summary may, in turn, be extractive or abstractive. The task of extractive multimodal text summarization involves selection and concatenation of the most important sentences in the input text without altering the sentences or their sequence in any way. Li et al. (2017) made the selection of these important sentences using visual and acoustic cues from the corresponding visual and auditory modalities. On the other hand, the task of abstractive multimodal text summarization involves identification of the theme of the input data and the generation of words based on the deeper understanding of the material. This is a tougher problem to solve which has been alleviated with the advancements in the abstractive text summarization techniques – Rush et al. (2015), See et al. (2017) and Liu and Lapata (2019). Sanabria et al. (2018) introduced the How2 dataset for large-scale multimodal language understanding, and Palaskar et al. (2019) were able to produce state of the art results for multimodal abstractive text summarization on the dataset. They utilized a sequence-to-sequence hierarchical attention based technique (Libovický and Helcl, 2017) for combining textual and image features to produce the textual summary from the multimodal input. Moreover, they used speech for generating the speech-to-text transcriptions using pre-trained speech recognizers, however it did not supplement the other modalities.

Though the previous work in abstractive multimodal text summarization has been promising, it has not yet been able to capture the effects of combining the audio features. Our work improves upon this shortcoming by examining the benefits and challenges of introducing the audio modality as part of our solution. We hypothesize that the audio modality can impart additional useful information for the text summarization task by letting the model pay more attention to words that are spoken with a certain tone or level of emphasis. Through our

* indicates equal contribution
Aman Khullar is presently at Gram Vaani

Proceedings of the First International Workshop on Natural Language Processing Beyond Text, pages 60–69
Online, November 20, 2020.
http://www.aclweb.org/anthology/W23-20%2d

Original text: let's talk now about how to bait a tip up hook with a maggot. typically, you're going to be using this for pan fish. not a real well known or common technique but on a given day it could be the difference between not catching fish and catching fish. all you do, you take your maggot, you can use meal worms, as well, which are much bigger, which are probably more well suited for this because this is a rather large hook. you would just, again, put that hook right through the maggot. with a big hook like this, i would probably put ten of these on it, just line the whole thing. this is going to be more of a technique for pan fish, such as, perch and sunfish, some of your smaller fish but if you had maggots, like this , or a meal worm, or two, on a hook like this, this would be a fantastic setup for trout, as well.
Text only: ice fishing is used for ice fishing. learn about ice fishing bait with tips from an experienced fisherman artist in this free fishing video.
Video-Text: learn about the ice fishing bait in this ice fishing lesson from an experienced fisherman.
MAST: maggots are good for catching perch. learn more about ice fishing bait in this ice fishing lesson from an experienced fisherman.

Table 1: Comparison of outputs by using different modality configurations for a test video example. Frequently occurring words are highlighted in red, which are easier for a simpler model to predict but do not contribute much in terms of useful content. The summary generated by the MAST model contains more content words as compared to the baselines.

experiments, we were able to prove that not all modalities contribute equally to the output. We found a higher contribution of text, followed by video and then by audio. This formed the motivation for our MAST model, which places higher importance on text input while generating the output summary. MAST is able to produce a more illustrative summary of the original text (see Table 1) and achieves state of the art results.

In summary, our primary contributions are:

- Introduction of audio modality for abstractive multimodal text summarization.
- Examining the challenges of utilizing audio information and understanding its contribution in the generated summary.
- Proposition of a novel state of the art model, MAST, for the task of multimodal abstractive text summarization.

2 Methodology

In this section we describe (1) the dataset used, (2) the modalities, and (3) our MAST model's architecture. The code for our model is available online[1].

[1] https://github.com/amankhullar/mast

2.1 Dataset

We use the 300h version of the How2 dataset (Sanabria et al., 2018) of open-domain videos. The dataset consists of about 300 hours of short instructional videos spanning different domains such as cooking, sports, indoor/outdoor activities, music, and more. A human-generated transcript accompanies each video, and a 2 to 3 sentence summary is available for every video, written to generate interest in a potential viewer. The 300h version is used instead of the 2000h version because the audio modality information is only available for the 300h subset.

The dataset is divided into the training, validation and test sets. The training set consists of 13,168 videos totaling 298.2 hours. The validation set consists of 150 videos totaling 3.2 hours, and the test set consists of 175 videos totaling 3.7 hours. A more detailed description of the dataset has been given by Sanabria et al. (2018). For our experiments, we took 12,798 videos for the training set, 520 videos for the validation set and 127 videos for the test set.

2.2 Modalities

We use the following three inputs corresponding to the three different modalities used:

- **Audio**: We use the concatenation of 40-dimensional Kaldi (Povey et al., 2011) filter bank features from 16kHz raw audio using a time window of 25ms with 10ms frame shift and the 3-dimensional pitch features extracted from the dataset to obtain the final sequence of 43-dimensional audio features.
- **Text**: We use the transcripts corresponding to each video. All texts are normalized and lower-cased.
- **Video**: We use a 2048-dimensional feature vector per group of 16 frames, which is extracted from the videos using a ResNeXt-101 3D CNN trained to recognize 400 different actions (Hara et al., 2018). This results in a sequence of feature vectors per video.

2.3 Multimodal Abstractive Summarization with Trimodal Hierarchical Attention

Figure 1 shows the architecture of our Multimodal Abstractive Summarization with Trimodal Hierarchical Attention (MAST) model. The model consists of three components - Modality Encoders, Tri-

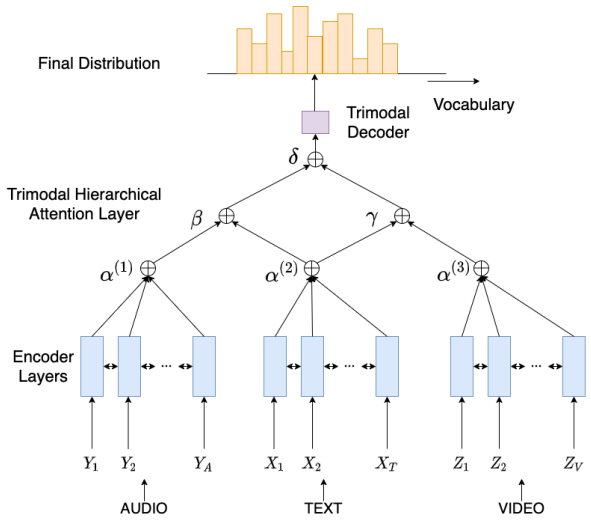

Figure 1: Multimodal Abstractive Summarization with Trimodal Hierarchical Attention (MAST) architecture: MAST is a sequence to sequence model that uses information from all three modalities – audio, text and video. The modality information is encoded using Modality Encoders, followed by a Trimodal Hierarchical Attention Layer, which combines this information using a three-level hierarchical attention approach. It attends to two pairs of modalities (δ) (Audio-Text and Video-Text) followed by the modality in each pair (β and γ), followed by the individual features within each modality (α). The decoder utilizes this combination of modalities to generate the output over the vocabulary.

modal Hierarchical Attention Layer and the Trimodal Decoder.

2.3.1 Modality Encoders

The text is embedded with an embedding layer and encoded using a bidirectional GRU encoder. The audio and video features are encoded using bidirectional LSTM encoders. This gives us the individual output encoding corresponding to all modalities at each encoder timestep. The tokens $t_i^{(k)}$ corresponding to modality k are encoded using the corresponding modality encoders and produce a sequence of hidden states $h_i^{(k)}$ for each encoder time step (i).

2.3.2 Trimodal Hierarchical Attention Layer

We build upon the hierarchical attention approach proposed by Libovický and Helcl (2017) to combine the modalities. On each decoder timestep i, the attention distribution (α) and the context vector for the k-th modality is first computed independently as in Bahdanau et al. (2014):

$$e_{ij}^{(k)} = v_a^{(k)\mathrm{T}} \tanh(W_a^{(k)} s_i + U_a^{(k)} h_j^{(k)} + b_{att}^{(k)}) \quad (1)$$

$$\alpha_{ij}^{(k)} = \mathrm{softmax}(e_{ij}^{(k)}) \quad (2)$$

$$c_i^{(k)} = \sum_{\substack{j=1, \\ k \in \{\text{audio, text, video}\}}}^{N_k} \alpha_{ij}^{(k)} h_j^{(k)} \quad (3)$$

Where s_i is the decoder hidden state at i-th decoder timestep, $h_j^{(k)}$ is the encoder hidden state at j-th encoder timestep, N_k is the number of encoder timesteps for the k-th modality and $e_{ij}^{(k)}$ is attention energy corresponding to them. W_a and U_a are trainable projection matrices, v_a is a weight vector and b_{att} is the bias term.

We now look at two different strategies of combining information from the modalities. The first is a simple extension of the hierarchical attention combination. The second is the strategy used in MAST, which combines modalities using three levels of hierarchical attention.

1. TrimodalH2: To obtain our first baseline model (*TrimodalH2*), with 2 level attention hierarchy, the context vectors for all three modalities are combined using a second layer of attention mechanism and its context vector is computed separately by using hierarchical attention combination as in Libovický and Helcl (2017):

$$e_i^{(k)} = v_b^T \tanh(W_b s_i + U_b^{(k)} c_i^{(k)}) \quad (4)$$

$$\eta_i^{(k)} = \mathrm{softmax}(e_i^{(k)}) \quad (5)$$

$$c_i = \sum_{k \in \{\text{audio, text, video}\}} \eta_i^{(k)} U_c^{(k)} c_i^{(k)} \quad (6)$$

where $\eta^{(k)}$ is the hierarchical attention distribution over the modalities, $c_i^{(k)}$ is the context vector of the k-th modality encoder, v_b and W_b are shared parameters across modalities, and $U_b^{(k)}$ and $U_c^{(k)}$ are modality-specific projection matrices.

2. MAST: To obtain our MAST model, the context vectors for audio-text and text-video are combined using a second layer of hierarchical attention mechanisms (β and γ) and their context vectors are computed separately. These context-vectors are then combined using the third hierarchical attention mechanism (δ).

1. Audio-Text:

$$e_i^{(k)} = v_d^T \tanh(W_d s_i + U_d^{(k)} c_i^{(k)}) \quad (7)$$

$$\beta_i^{(k)} = \text{softmax}(e_i^{(k)}) \quad (8)$$

$$d_i^{(1)} = \sum_{k \in \{\text{audio, text}\}} \beta_i^{(k)} U_e^{(k)} c_i^{(k)} \quad (9)$$

2. Video-Text:

$$e_i^{(k)} = v_f^T \tanh(W_f s_i + U_f^{(k)} c_i^{(k)}) \quad (10)$$

$$\gamma_i^{(k)} = \text{softmax}(e_i^{(k)}) \quad (11)$$

$$d_i^{(2)} = \sum_{k \in \{\text{video, text}\}} \gamma_i^{(k)} U_g^{(k)} c_i^{(k)} \quad (12)$$

where $d_i^{(l)}$, $l \in \{\text{audio-text, video-text}\}$ is the context vector obtained for the corresponding pair-wise modality combination.

Finally, these audio-text and video-text context vectors are combined using the third and final attention layer (δ). With this trimodal hierarchical attention architecture, we combine the textual modality twice with the other two modalities in a pair-wise manner, and this allows the model to pay more attention to the textual modality while incorporating the benefits of the other two modalities.

$$e_i^{(l)} = v_h^T \tanh(W_g s_i + U_h^{(l)} d_i^{(l)}) \quad (13)$$

$$\delta_i^{(l)} = \text{softmax}(e_i^{(l)}) \quad (14)$$

$$c_i^f = \sum_{l \in \{\text{audio-text, video-text}\}} \delta_i^{(l)} U_m^{(l)} d_i^{(l)} \quad (15)$$

where c_i^f is the final context vector at i-th decoder timestep.

2.3.3 Trimodal Decoder

We use a GRU-based conditional decoder (Firat and Cho, 2016) to generate the final vocabulary distribution at each timestep. At each timestep, the decoder has the aggregate information from all the modalities. The trimodal decoder focuses on the modality combination, followed by the individual modality, then focuses on the particular information inside that modality. Finally, it uses this information along with information from previous timesteps, which is passed on to two linear layers to generate the next word from the vocabulary.

3 Experiments

We train Trimodal Hierarchical Attention (MAST) and TrimodalH2 models on the 300h version of the How2 dataset, using all three modalities. We also train Hierarchical Attention models considering Audio-Text and Video-Text modalities, as well as simple Seq2Seq models with attention for each modality individually as baselines. As observed by Palaskar et al. (2019), the Pointer Generator model (See et al., 2017) does not perform as well as Seq2Seq models on this dataset, hence we do not use that as a baseline in our experiments. We consider another transformer-based baseline for the text modality, BertSumAbs (Liu and Lapata, 2019).

For all our experiments (except for the BerSumAbs baseline), we use the *nmtpytorch* toolkit (Caglayan et al., 2017). The source and the target vocabulary consists of 49,329 words on which we train our word embeddings. We use the NLL loss and the Adam optimizer (Kingma and Ba, 2014) with learning rate 0.0004 and trained the models for 50 epochs. We generate our summaries using beam search with a beam size of 5, and then evaluate them using the ROUGE metric (Lin, 2004) and the Content F1 metric (Palaskar et al., 2019).

In our experiments, the text is embedded with an embedding layer of size 256 and then encoded using a bidirectional GRU encoder (Cho et al., 2014) with a hidden layer of size 128, which gives us a 256-dimensional output encoding corresponding to the text at each timestep. The audio and video frames are encoded using bidirectional LSTM encoders (Hochreiter and Schmidhuber, 1997) with a hidden layer of size 128, which gives a 256-dimensional output encoding corresponding to the audio and video features at each timestep. Finally, the GRU-based conditional decoder uses a hidden layer of size 128 followed by two linear layers which transform the decoder output to generate the final output vocabulary distribution.

To improve generalization of our model, we use two dropout layers within the Text Encoder and one dropout layer on the output of the conditional decoder, all with a probability of 0.35. We also use implicit regularization by using early stopping mechanism on the validation loss with a patience of 40 epochs.

3.1 Challenges of using audio modality

The first challenge comes with obtaining a good representation of the audio modality that adds value beyond the text modality for the task of text summarization. As found by Mohamed (2014), DNN acoustic models prefer features that smoothly

change both in time and frequency, like the log mel-frequency spectral coefficients (MFSC), to the decorrelated mel-frequency cepstral coefficients (MFCC). MFSC features make it easier for DNNs to discover linear relations as well as higher order causes of the input data, leading to better overall system performance. Hence we do not consider MFCC features in our experiments and use the filter bank features instead.

The second challenge arises due to the larger number of parameters that a model needs when handling the audio information. The number of parameters in the Video-Text baseline is 16.95 million as compared to 32.08 million when we add audio. This is because of the high number of input timesteps in the audio modality encoder, which makes learning trickier and more time-consuming.

To demonstrate these challenges, as an experiment, we group the audio features across input timesteps into bins with an average of 30 consecutive timesteps and train our MAST model. This makes the number of audio timesteps comparable to the number of video and text timesteps. While we observe an improvement in computational efficiency, it achieves a lower performance than the baseline Video-Text model as described in Table 2 (MAST-Binned). We also train Audio only and Audio-Text models which fail to beat the Text only baseline. We observe that the generated summaries of the Audio only model are similar and repetitive, indicating that the model failed to learn useful information relevant to the task of text summarization.

4 Results and Discussion

Model Name	ROUGE			Content F1
	1	2	L	
Text Only	46.01	25.16	39.98	33.45
BertSumAbs	29.68	11.74	22.58	31.53
Video Only	39.23	19.82	34.17	27.06
Audio Only	29.16	12.36	28.86	26.65
Audio-Text	34.56	15.22	31.63	28.36
Video-Text	48.40	27.97	42.23	32.89
TrimodalH2	47.85	28.46	42.17	**35.65**
MAST-Binned	46.22	25.94	40.34	33.56
MAST	**48.85**	**29.51**	**43.23**	35.40

Table 2: Results for different configurations. MAST outperforms all baseline models in terms of ROUGE scores, and obtains a higher Content-F1 score than all baselines while obtaining a score close to the TrimodalH2 model.

4.1 Preliminaries

Our results are given in Table 2. To demonstrate the contribution of various modalities towards the output summary, we experiment with the three modalities taken individually as well as in combination. Text only, Video only and the Audio only are attention-based S2S models (Bahdanau et al., 2014) with their respective modality features taken as encoder inputs. To situate the efficacy of the encoder-decoder architecture for our task, we use the BertSumAbs (Liu and Lapata, 2019) as a BERT based baseline for abstractive text summarization. Audio-Text and the Video-Text are S2S models with hierarchical attention layer. The Video-Text model as presented by Palaskar et al. (2019) has been compared on the 300h version instead of the 2000h version of the dataset because the audio modality is only available in the former. TrimodalH2 model, adds the audio modality in the second-level of hierarchical attention. MAST-Binned model groups the features of the audio modality for computational efficiency. These models show alternative methods for utilizing audio modality information.

We evaluate our models with the ROUGE metric (Lin, 2004) and the Content F1 metric (Palaskar et al., 2019). The Content F1 metric is the F1 score of the content words in the summaries based on a monolingual alignment. It is calculated using the METEOR toolkit (Denkowski and Lavie, 2011) by setting zero weight to function words (δ), equal weights to Precision and Recall (α), and no cross-over penalty (γ) for generated words. Additionally, a set of catchphrases like the words - in, this, free, video, learn, how, tips, expert - which appear in most summaries and act like function words instead of content words are removed from the reference and hypothesis summaries as a post-processing step. It ignores the fluency of the output, but gives an estimate of the amount of useful content words the model is able to capture in the output.

4.2 Discussion

As observed from the scores for the Text Only model, the text modality contains the most amount of information relevant to the final summary, followed by the video and the audio modalities. The scores obtained by combining the audio-text and video-text modalities also indicate the same. The transformer-based model, BertSumAbs, fails to perform well because of the smaller amount of text

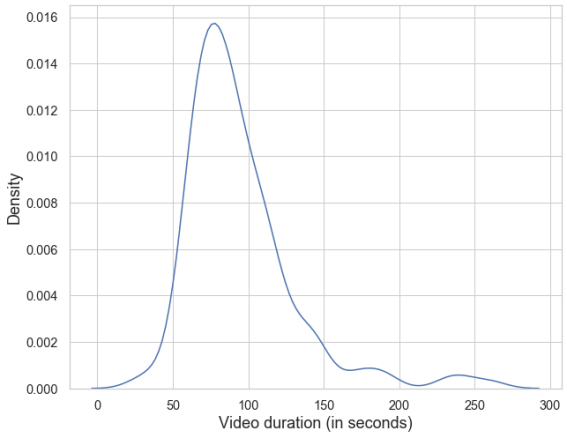

Figure 2: Distribution of the duration of videos (in seconds) in the test set.

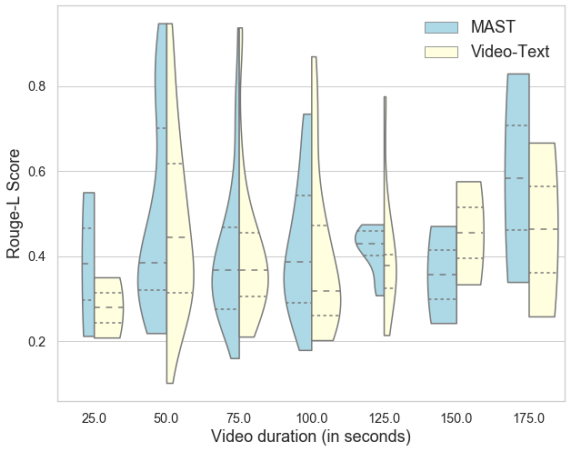

Figure 3: Distribution of Rouge-L scores of summaries produced for different video durations (in seconds) for MAST and Video-Text baseline. The videos are binned into groups of 25 seconds by duration and the distribution of Rouge-L scores within each group is shown using density plots. The dotted lines inside each group show the quartile distribution.

data available to fine-tune the model.

We also observe that combining the text and audio modalities leads to a lower ROUGE score than the Text Only model, which indicates that the plain hierarchical attention model fails to learn well over the audio modality by itself. This observation is in line with the result obtained by the TrimodalH2 model, where we simply extend the hierarchical attention approach to three modalities.

4.2.1 Usefulness of audio modality

The MAST and the TrimodalH2 models achieve a higher Content F1 score than the Video-Text baseline, indicating that the model learns to extract more useful content by utilizing information from the audio modality corresponding to the characteristics of speech, in line with our initial hypothesis as illustrated in Table 1

However, the TrimodalH2 model, which simply adds the audio modality in the second level of hierarchical attention, fails to outperform the Video-Text baseline in terms of ROUGE scores. Our architecture lets the MAST model choose between paying attention to a different combination of modalities with the text modality. This forces the model to pay more attention to the text modality, thereby overcoming the shortcoming of the TrimodalH2 model and achieving better ROUGE scores, while maintaining a similar Content F1 score when compared to TrimodalH2.

4.2.2 Attention distribution across modalities

To understand the importance of individual modalities and their combinations, we plot their attention distribution at different levels of attention hierarchy across the decoder timesteps. Figure 4a corresponds to attention weights as calculated in equation 14 while figures 4b and 4c correspond to the product of attention weights between equations 11, 8 and corresponding weight in equation 14 for each decoder timestep. The final attention within each individual modality at each decoder timestep is calculated by multiplying the corresponding cumulative attention weights obtained at level 2 of attention hierarchy with the attention weights obtained in equation 2 (figures 4d to 4f). The attention weights assigned to the audio modality have been added across input timesteps (group size of 30) in order to obtain a more interpretable visualization.

Through these visualizations, we observe that the text modality dominates the generation of the output summary while giving lesser attention to the audio and video modalities (the latter being more important). These findings support the extra importance being given to the text modality in the MAST model during its interaction with the other modalities. Figures 4b and 4d highlight the modest gains through the audio modality and the challenge in its appropriate usage.

4.2.3 Performance across video durations

We also look at how our model performs for different video durations in our test set. Figure 3 shows the variation in the Rouge-L scores across different videos for MAST and the Video-Text baseline. The figure shows videos binned into seven groups of 25 seconds by duration. We can observe from the quartile distribution that MAST outperforms the baseline in five out of the seven groups, gives simi-

lar performance for videos with a duration between 75-100 seconds, and underperforms for videos with a duration between 150-175 seconds. However, overall, by looking at the distribution of the duration of videos in our test set (Figure 2), we can observe that MAST outperforms the baseline for a vast majority of videos across durations.

5 Related Work

5.1 Abstractive text summarization

Abstractive summarization of documents was traditionally achieved by paraphrasing and fusing multiple sentences along with their grammatical rewriting (Woodsend and Lapata, 2012). This was later improved by taking inspiration from human comprehension capabilities when Fang and Teufel (2014) implemented the model of human comprehension and summarization proposed by Kintsch and Van Dijk (1978). They did this by identifying these concepts in text through the application of co-reference resolution, named entity recognition and semantic similarity detection, implemented as a two-step competition.

The real stimulus to the field of abstractive summarization was provided by the application of neural encoder-decoder architectures. Rush et al. (2015) were among the first to achieve state-of-the-art results on Gigaword (Graff et al., 2003) and the DUC-2004 (Over et al., 2007) datasets and established the importance of end-to-end deep learning models for abstractive summarization. Their work was later improved upon by See et al. (2017) where they used copying from the source text to remove the problem of incorrect generation of facts in the summary, as well as a coverage mechanism to curb the problem of repetition of words in the generated summary.

5.2 Pretrained language models

Another breakthrough for the field of natural language processing came with the use of pre-trained language models for carrying out various language downstream tasks. Pre-trained language models like BERT (Devlin et al., 2018) introduced masked language modelling, which allowed models to learn interactions between left and right context words. These models have significantly changed the way word embeddings are generated by training contextual embeddings rather than static embeddings. Liu and Lapata (2019) presented how BERT could be used for text summarization and proposed a new fine-tuning schedule for abstractive summarization which adopted different optimizers for the encoder and the decoder to alleviate the mismatch between the two. BERT models typically require large amounts of annotated data to produce state-of-the-art results. Recent works, like GAN-BERT by Croce et al. (2020) focus on solving this problem.

5.3 Advancements in speech recognition and computer vision

Parallel advancements in the field of speech recognition and computer vision have been able to give us successful methods to extract useful features of speech and images. Peddinti et al. (2015) built a robust acoustic model for speech recognition using a time-delay neural network. They were able to achieve state-of-the-art results in the IARPA ASpIRE Challenge. Similarly, with the advancements of convolutional neural networks, the field of computer vision has progressed significantly. He et al. (2016) demonstrated the strength of deep residual networks which learned residual functions with reference to the layers and were able to achieve state-of-the art results on the ImageNet dataset. Hara et al. (2018) showed that simple 3D Convolutional Neural Network (CNN) architectures outperform complex 2D architectures and trained a ResNeXt-101 3D CNN to recognize 400 different human actions on the Kinetics dataset (Kay et al., 2017).

5.4 Summarization beyond text

The advancements in these fields have in turn also facilitated text summarization. Rott and Červa (2016) used only the input audio to generate textual summaries while Sah et al. (2017) were among the first to show the possibility of summarizing long videos and then annotating the summarized video to obtain a textual summary. These models, however, were not able to capture the information of other modalities to obtain the output textual summary and hence their limitations led to the increasing use of multimodal data. A major hindrance in the field of multimodal text summarization was the lack of datasets. Li et al. (2017) created an asynchronous benchmark dataset with human-annotated summaries for 500 videos. Sanabria et al. (2018) then released a large-scale dataset for instructional videos. JN et al. (2020) and Zhu et al. (2018) presented multimodal text summarization models using textual and visual modalities as input and multimodal outputs of summarized text and video. Palaskar et al. (2019) used How2 dataset

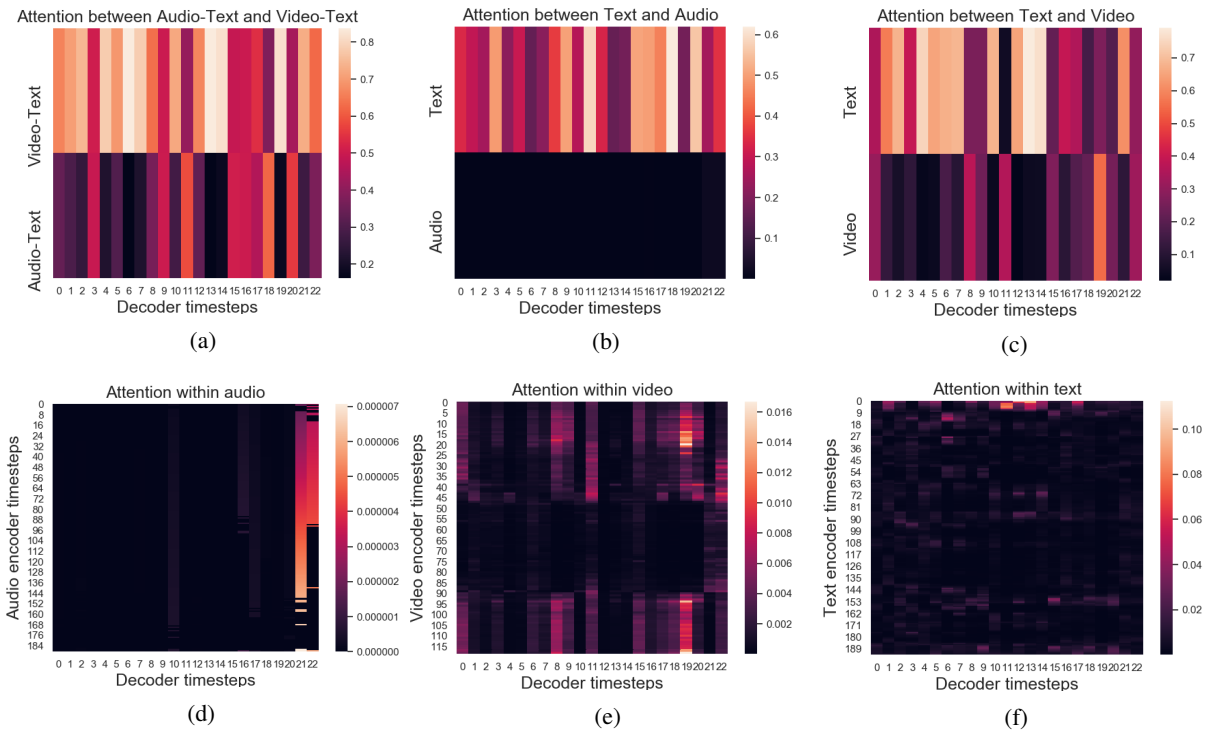

Figure 4: Visualization of attention weights in the Trimodal Hierarchical Attention layer for a sample video in the test set. Figures 4a to 4c show the varying attention distribution on different combinations of modalities across the decoder timesteps. Figures 4d to 4f show the attention distribution on the encoder timesteps for each modality across the decoder timesteps. This shows the usefulness of each modality for the generation of the summary.

to present an abstractive summary of open-domain videos. These models, however, are not completely multimodal since they do not utilise the audio information. A major focus of our work is to highlight the importance of using audio data as input and incorporate it in a truly multimodal manner.

6 Conclusion

In this work[2], we presented MAST, a state of the art sequence to sequence based model that uses information from all three modalities – audio, text and video – to generate abstractive multimodal text summaries. It uses a Trimodal Hierarchical Attention layer to utilize information from all modalities. We explored the role played by adding the audio modality and compared MAST with several baseline models, demonstrating the effectiveness of our approach.

In the future, we would like to extend this work by looking at alternate audio modality representations including using neural networks for audio feature extraction, and also explore the use of transformers for an end to end attention based learning. We also aim to explore the application of MAST to other multimodal tasks like translation.

[2] https://github.com/amankhullar/mast

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Ozan Caglayan, Mercedes García-Martínez, Adrien Bardet, Walid Aransa, Fethi Bougares, and Loïc Barrault. 2017. Nmtpy: A flexible toolkit for advanced neural machine translation systems. *The Prague Bulletin of Mathematical Linguistics*, 109(1):15–28.

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. *arXiv preprint arXiv:1406.1078*.

Danilo Croce, Giuseppe Castellucci, and Roberto Basili. 2020. Gan-bert: Generative adversarial learning for robust text classification with a bunch of labeled examples. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2114–2119.

Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In *Proceed-*

ings of the sixth workshop on statistical machine translation, pages 85–91.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Yimai Fang and Simone Teufel. 2014. A summariser based on human memory limitations and lexical competition. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 732–741.

Orhan Firat and Kyunghyn Cho. 2016. Conditional gated recurrent unit with attention mechanism. *https://github.com/nyu-dl/dl4mt-tutorial/blob/master/docs/cgru.pdf*.

David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2003. English gigaword. *Linguistic Data Consortium, Philadelphia*, 4(1):34.

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*, pages 6546–6555.

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*, pages 770–778.

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

Zhu JN, Zhang JJ, Li HR, Zong CQ, et al. 2020. Multimodal summarization with guidance of multimodal reference. Association for Computational Linguistics.

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. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Walter Kintsch and Teun A Van Dijk. 1978. Toward a model of text comprehension and production. *Psychological review*, 85(5):363.

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

Jindřich Libovickỳ and Jindřich Helcl. 2017. Attention strategies for multi-source sequence-to-sequence learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 196–202.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3721–3731.

Abdel-rahman Mohamed. 2014. Deep neural network acoustic models for asr.

Paul Over, Hoa Dang, and Donna Harman. 2007. Duc in context. *Information Processing & Management*, 43(6):1506–1520.

Shruti Palaskar, Jindrich Libovickỳ, Spandana Gella, and Florian Metze. 2019. Multimodal abstractive summarization for how2 videos. *arXiv preprint arXiv:1906.07901*.

Vijayaditya Peddinti, Guoguo Chen, Vimal Manohar, Tom Ko, Daniel Povey, and Sanjeev Khudanpur. 2015. Jhu aspire system: Robust lvcsr with tdnns, ivector adaptation and rnn-lms. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 539–546. IEEE.

Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al. 2011. The kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding*, CONF. IEEE Signal Processing Society.

Michal Rott and Petr Červa. 2016. Speech-to-text summarization using automatic phrase extraction from recognized text. In *International Conference on Text, Speech, and Dialogue*, pages 101–108. Springer.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389.

Shagan Sah, Sourabh Kulhare, Allison Gray, Subhashini Venugopalan, Emily Prud'Hommeaux, and Raymond Ptucha. 2017. Semantic text summarization of long videos. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 989–997. IEEE.

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. *arXiv preprint arXiv:1811.00347*.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

Kristian Woodsend and Mirella Lapata. 2012. Multiple aspect summarization using integer linear programming. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 233–243. Association for Computational Linguistics.

Junnan Zhu, Haoran Li, Tianshang Liu, Yu Zhou, Jiajun Zhang, Chengqing Zong, et al. 2018. Msmo: multimodal summarization with multimodal output.