CARNEGIE MELLON UNIVERSITY

Multimodal Learning from Videos

Exploring Models and Task Complexities

Thesis Proposal

by

Shruti Palaskar

Thesis proposal submitted in partial fulfillment
for the degree of Doctor of Philosophy

Thesis committee:
Florian Metze, CMU (co-chair)
Alan W Black, CMU (co-chair)
Yonatan Bisk, CMU
Cordelia Schmid, INRIA and Google AI

April 2021

Abstract

Human learning is inherently multimodal. We watch, listen, read, and communicate to learn from and understand our surroundings. There have been tremendous advancements in Machine Learning fields related to these human activities such as Speech Recognition or Computer Vision that make computationally modeling this human-like inherent multimodal learning a possibility. Multimodal Video Understanding as a machine learning task is close to this form of learning.

This thesis proposes to break down this complex task of video understanding into a series of relatively simpler tasks with increasing complexity. We start with the monotonic task of speech recognition and introduce an end-to-end audio-visual speech recognition model. A more complex task is speech translation that tackles re-ordered output sequences in addition to speech recognition, which is the second task in this thesis. For speech translation, we introduce a multimodal fusion model that learns to leverage the multiple views multimodal data provides in a semi-supervised way. Further, we progress to the tasks of multimodal video summarization and question answering that tackle abstract-level understanding tasks further involving information compression and restructuring. Finally, we propose to extend this work to multimodal commonsense rationale generation that not only performs abstract-level learning, but also provides an explanation of the achieved video understanding. For the four main tasks, we present a series of multimodal fusion models based on the nature and complexity of the task, the modalities involved in each and compare and contrast the models on commonly used video and language understanding datasets.

Contents

List of Figures

List of Tables

Chapter 1

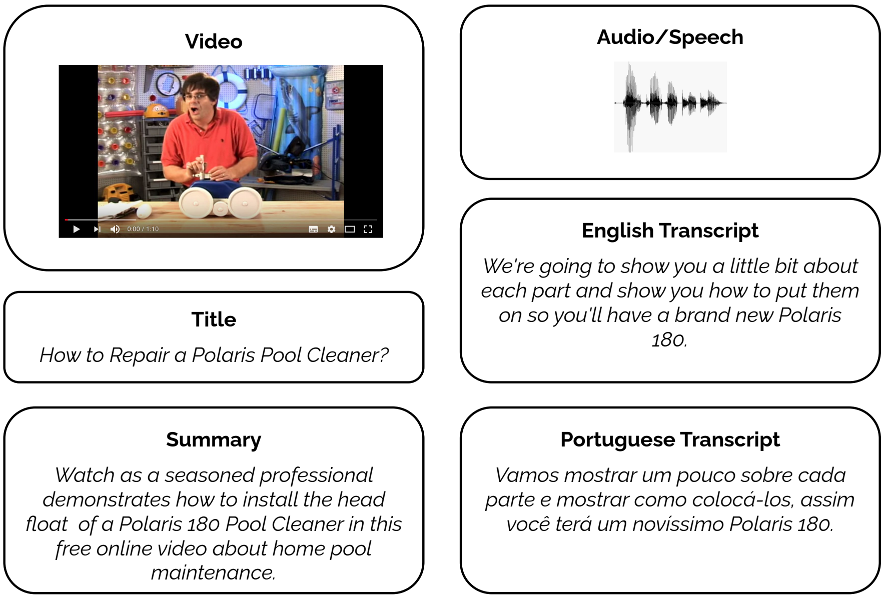
Introduction

Videos are an inherent part of our day-to-day lives in the 21st Century. One of the goals of Artificial Intelligence (AI) research is to teach machines to perform certain tasks to off-load human labor, by modeling and simulating human intelligence. Videos closely capture our interactions in the world and often contain multiple modalities of information such as the audio, transcription, and the visuals itself.

Self-recorded instructional videos are a domain of videos that particularly focus on coaching a viewer on certain tasks, colloquially known as the How-To videos. These videos are informative, explanatory, and with visual and spoken demonstration, for example, making a Cuban breakfast omelet, or fixing a Polaris swimming pool cleaner (e.g. Figure 1.1). Such videos provide a readily available, well crafted source of instructional information that can be used to teach machines certain How-To tasks.

With this long-term view in mind, we approach this problem through the task of Multimodal Video Understanding. This task accounts for the various modalities available in a video, as well as performs downstream tasks that demonstrate "understanding". Video Understanding as a research problem is still in its nascent phase with various tasks falling under the umbrella term understanding ranging from video classification to video commonsense reasoning.

Video Understanding was initially approached as a video classification problem analogous to image classification with time-series input (Karpathy et al., 2014; Yue-Hei Ng et al., 2015; Szegedy et al., 2016; Abu-El-Haija et al., 2016). Research soon progressed towards human action recognition (Simonyan and Zisserman, 2014a; Caba Heilbron et al., 2015; Feichtenhofer et al., 2016; Carreira and Zisserman, 2017; Kay et al., 2017; Gu et al., 2018), scene understanding (Feichtenhofer et al., 2014; Tran et al., 2015; Ros et al., 2016; Cordts et al., 2016), text-to-video retrieval (Miech et al., 2019; Gabeur et al., 2020; Albanie et al., 2020), or video captioning (Yao et al., 2015; Yu et al., 2016; Krishna et al., 2017; Zhou et al., 2018b) and description generation (Das et al., 2013; Regneri et al., 2013; Donahue et al., 2015). More complex language-based video understanding task formulations followed the success of easier tasks such as video captioning



How to Repair a Polaris Pool Cleaner?

We're going to show you a little bit about each part and show you how to put them on so you'll have a brand new Polaris 180.

Watch as a seasoned professional demonstrates how to install the head float of a Polaris 180 Pool Cleaner in this free online video about home pool maintenance.

Vamos mostrar um pouco sobre cada parte e mostrar como colocá-los, assim você terá um novíssimo Polaris 180.

FIGURE 1.1: An example from the How2 dataset showing the different modalities that exist in this data. There are 4 time-synchronous modalities: the video signal, the audio/speech signal, corresponding English transcripts and the Portuguese transcripts. There is a human annotated textual summary for each video. Additionally, we also have access to relevant metadata such as the video title, topics, categories, view count, etc.

or description generation. More recently, there is a huge push towards unsupervised, self-supervised, and semi-supervised approaches to representation learning that use existing annotations to train general purpose multimodal representations that can be later used in downstream tasks (Chen et al., 2019; Sun et al., 2019; Miech et al., 2020; Arnab et al., 2021).

Despite the tremendous progress in such tasks, the field of Video Understanding has not quite progressed towards *holistic* Video Understanding that studies the video signal in its natural form with audio, text, speech, metadata, and other available modalities. The Multimodal Video Understanding tasks involve at least one additional modality apart from the video signal used in tandem towards a downstream task. Often, there is a lack of annotated datasets that facilitate multimodal tasks which contain more than 3 parallel modalities. One of the first 5-way parallel multimodal videos dataset of instructional videos, the How2 dataset , was collected and released by Sanabria et al. (2018). Figure 1.1 shows a sample video along with its various modalities from the How2 dataset . Being the first-of-its-kind 5-way parallel dataset, it enabled the formulation of various video understanding tasks, with a special focus on language-based understanding such as summarization or translation. In this thesis, we present a series of tasks that can be performed unimodally and multimodally with this data, which together, constitute a step towards *holistic* Multimodal Video Understanding. Figure 1.2 gives an overview of the four major tasks detailed in this thesis.

The four main learning tasks for Multimodal Video Understanding covered in this thesis are Speech Recognition, Speech Translation, Summarization, and Rationalization. These tasks are ordered in terms of increasing task complexity and specific model architectures and constraints

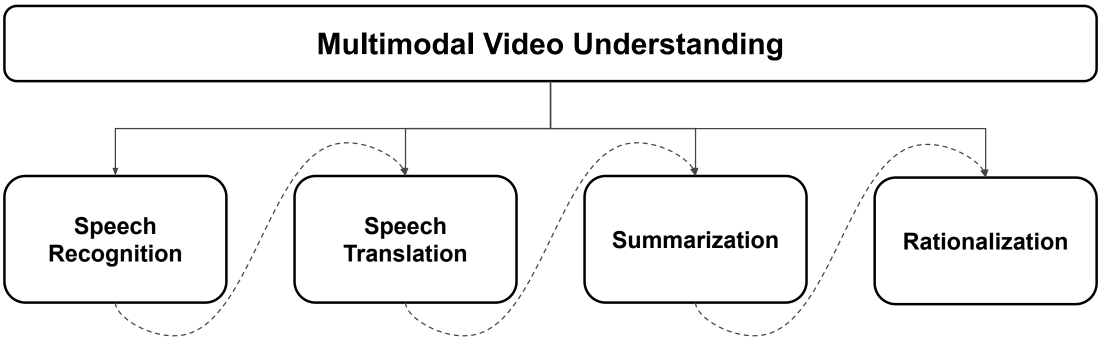


FIGURE 1.2: Overview of the Multimodal Video Understanding tasks in this thesis: Speech Recognition, Speech Translation, Summarization, and Rationalization. The tasks are ordered in increasing level of complexity from left to right.

are designed based on the task-specific complexities. Each of these tasks is performed unimodally as well as multimodally to investigate the influence of multimodal data for each task. Despite the wide array of understanding tasks, the tasks in this thesis are language generation-based, exploring video understanding from a speech-vision-language based generation perspective.

We start with a monotonically constrained audio-visual task, Video Speech Recognition (Palaskar and Metze, 2018; Palaskar et al., 2018; Caglayan et al., 2019). Input speech and corresponding transcription which is the output of speech recognition follow a strict monotonic constraint where the temporal order of the speech is maintained in the temporal order of the transcription. We refer to this task as the *Monotonic Learning* task.

Beyond *Monotonic Learning*, we can extend to a Speech Translation task, that maintains similar inputs as the Speech Recognition task, but handles a non-monotonic constraint in the output. The translation outputs can be re-ordered (not following the same temporal constraint as the input speech), and are language dependent. We refer to this type of learning as *Non-Monotonic*. The translation task in this thesis is on English and Portuguese (Palaskar et al., 2019b; Holzenberger et al., 2019).

Abstract Learning tasks are those that require abstraction of the inputs to generate the corresponding outputs, for example, generating natural language summaries for videos, or video question answering. Often, these tasks involve compression, restructuring, and rephrasing of input information to perform the necessary *abstraction.* These are comparatively complex tasks than Recognition or Translation and focus on video-level non-trivial understanding. Video Summarization (Palaskar et al., 2019a) and Video Question Answering (Sanabria et al., 2019; Palaskar et al., 2020) are the abstract learning tasks here.

Finally, to extend beyond these tasks, we address an *Explanatory* task that provides human interpretable rationales for human actions in videos beyond summarizing or answering questions about them. Visual Commonsense Reasoning (Zellers et al., 2019) and Video Question Answering with Explanation (Li et al., 2018) tasks have been proposed in recent years but are often modeled as a classification problem of choosing a correct option from four options. This formulation of the task has heavy annotation requirement and cannot be extended to open-vocabulary free-form generation. To extend this task beyond these constraints, we use existing datasets collected for classification in a free-form natural language rationale generation setting (Rationalization) as the final learning task in this thesis.

The collection and release of the `How2` dataset enabled the design and execution of many of these tasks. It also led to the collection of a much larger scale dataset containing similar types of videos (Miech et al., 2019). Using the `How2` dataset , a huge effort was led by Specia et al. (2020) in the Summer of 2018 at Johns Hopkins University, Baltimore, MD, to explore a *one-model-rules-all* type of model for this multimodal understanding task. A lot of interesting results emerged from this effort for each of the tasks above, one of it being that is it more effective to optimize towards each task individually than in a single large model (Specia et al., 2020) with current technology. Very recently, researchers are exploring large-scale pre-training using Transformer models (Vaswani et al., 2017) that try to build many-in-one latent representations by pre-training on auxiliary tasks such as classification (Chen et al., 2019; Lu et al., 2020; Nguyen and Okatani, 2019; Pramanik et al., 2019). Such models have not yet shown good performance on natural language generation based tasks covered in this thesis. With further developments in this direction, the *one-model-rules-all* approach might have success in the future.

Overall, with this breakdown of Multimodal Video Understanding into various smaller tasks, we aim to establish a structured pathway towards the bigger goal of *"Can machines learn multimodally from the world as humans naturally do?"*

1.1 Thesis Overview

Thesis Statement

Multimodal Video Understanding is a complex task, and various disconnected tasks have been proposed within this domain. This thesis ranks four such tasks according to the complexity and shows how increasingly expressive models are needed to perform well on them. Our ordering is intuitive to humans, and corresponds to the difficulty of human learning tasks.

Chapters & Contributions

I. Multimodal Learning Tasks. Multimodal Video Understanding is a complex task spanning numerous sub-tasks across the fields of Computer Vision, Natural Language, and Speech & Audio. To structure this learning problem in the context of this thesis, we identify four learning tasks, with gradually increasing complexities, and build multimodal fusion models based on the nature of the tasks and modalities involved. In this chapter, we define the four tasks, the models used for each, and the modalities involved that guide the model developed.

II. Speech Recognition. Automatic speech recognition simulates the human process of listening with a neural network. Similarly, audio-visual speech recognition emulates watching and listening synchronously. This is a strictly monotonic task. We propose a multimodal fusion model, Monotonic Input Fusion, based on the monotonicity and the time-scales of each three modalities: speech, video, and text.

Contributions

1. **[Finished Work]** Audio-Visual Speech Recognition
   We build a *Monotonic Input Fusion* model for end-to-end audio-visual speech recognition that learn to use relevant information from each modality to improve speech recognition performance. Using the multimodal fusion model, we demonstrate a relative improvement of 9% in the word error rate over unimodal models. The related publication is: Palaskar et al. 2018 which was later extended by Caglayan et al. 2019.

2. **[Finished Work]** Acoustic-to-Word Speech Recognition
   To fuse information in the semantic space, there is a difference of time-scale alignment of the speech signal (speech frames), video signal (latent representations for word-level labels) and the corresponding text sequence (characters). Towards bridging this gap, we build speech recognition models that operate directly at the word-level labels to match them with the visual representations. We build direct Acoustic-to-Word models that align speech frames with corresponding word labels and evaluate their performance against standard phoneme-based or character-based speech recognizers. The related publication is: Palaskar and Metze 2018.

III. Speech Translation. Speech Translation is a non-monotonic task that converts an ordered set of speech signals to a re-ordered set of Portuguese translations in words. The Acoustic-to-Word model from previous chapter is useful to bring the time-scales of speech and text to common word-level units here as well. Further, we also explore a semi-supervised learning model for this task called the *Latent Representation Fusion* model to utilize the inherent supervision provided by training across modalities. For translation, the Monotonic Input Fusion model can also be used if a supervised translation model is required.

Contributions

1. **[Finished Work]** Contextual Acoustic Word Embeddings
   We begin by building appropriate word-level representations from speech inputs. We propose a Contextual Acoustic Word Embeddings (CAWE) model that uses the location-aware attention (Chan et al., 2016) mechanism to localize and contextualize across all speech frames of a word to a single representation. the corresponding acoustic word embedding. Upon evaluation of these embeddings with their textual counterparts on 13 standard semantic similarity and classification tasks, we find the acoustic embeddings perform equally or better than the textual word embeddings, showing the semantic strength of the learned embeddings. The corresponding publication for this is Palaskar et al. 2019b.

2. **[Finished Work]** Multiview Learning for Multimodal Embeddings
   These acoustic word embeddings are used in a multi-view representation learning setup trained via Deep Canonical Correlation Analysis (Hotelling, 1992; Andrew et al., 2013) to perform semi-supervised speech recognition and speech translation. We evaluate these embeddings using a retrieval-based metric. Using the Latent Representation Fusion model, the semi-supervised model achieves within 3% WER of the fully supervised model for speech recognition, and within 7 BLEU points for speech translation. The corresponding publication for this is Holzenberger et al. 2019.

IV. Summarization & QA. Video Summarization is the task of generating a short informative textual summary highlighting the important contents of the video. For instructional videos, the video by itself is very detailed. A textual summary that attracts viewers based on its focus on the most differentiating factor is more relevant than summarizing the different instructions of the video – here the summary acting like a textual teaser of the video itself. In this chapter, we address this task of abstraction first. In the Video Question Answering task, we evaluate transfer learning capabilities of models trained on the `How2` dataset (all tasks so far are on `How2` dataset), on other open-domain datasets and tasks. One such dataset is the Audio-Visual Scene-aware Dialog dataset (Alamri et al., 2019) based on the Charades dataset (Sigurdsson et al., 2016) provides very similar types of modalities to the `How2` dataset : speech, video, summary, and question answer pairs, enabling transfer learning.

Contributions

1. **[Finished Work]** Video Summarization
   For the task of video summarization, our multimodal approach uses a *Hierarchical Latent Representation Fusion* model to fuse visual and textual data for summarization, leading to an absolute improvement 1.5 points in the content F1 score. We compare the proposed model and task with numerous strong multimodal and summarization models and find hierarchical model to perform best. Human evaluation on this novel task also measures the quality of generated summaries on coherency, fluency, informativeness, and relevance. Finally, we demonstrate the strong relevance of the visual signal for this abstractive generation task. The associated publication for this work is Palaskar et al. 2019a.

2. **[Finished Work]** Video Question Answering
   We explore various unimodal and multimodal transfer learning approaches for cross-dataset learning between the `How2` dataset and the Audio-Visual Scene-aware Dialog dataset (Alamri et al., 2019) here. We cast the task as a multi-modal video summarization problem, in which the input is video features along with concatenated with the textual question, and the summary is the desired "answer". We observed significant consistent gains by transfer learning for all unimodal tasks (Sanabria et al., 2019; Palaskar et al., 2020) through transfer learning but did not observe significant gain or drop in performance for the multimodal task. With our best multimodal transfer learning model for dialog question answering, we participated and ranked first in the 7^{th} Dialog State Tracking Challenge, Audio-Visual Scene-aware Dialog track (Yoshino et al., 2018) on both automatic and human evaluation metrics. The associated publications for this work are Sanabria et al. 2019 and Palaskar et al. 2020.

V. Rationalization. As part of proposed work, we extend to a natural language rationale generation task, also commonly known as commonsense reasoning. This is also a multimodal task with image or video for the visual modality. This is a relatively new area of work compared to other tasks covered in this thesis. The goal for this task is to extend question answering by providing supporting rationales for a given system-generated answer. This task differs from abstractive summarization as the information required for the rationale to be generated may not be present in the input at all. Rather, it is a commonsense deduction based on the provided context (for well-defined scope of the term "commonsense").

Contributions

1. **[Proposed Work]** Interpretable Visual Commonsense Reasoning
   For the Visual Commonsense Reasoning dataset collected by Zellers et al. 2019, we design a generation-based task for rationalization. In this work, we propose to study the efficacy of this generation task, and evaluate the dependency between the generated answer and generated rationale.

2. **[Proposed Work]** Improving Joint Answer and Rationale generation
   Further, we propose to specifically build joint models designed for computational control over the dependency between the generated answer and rationale for this task, called the *Hierarchical Interpretable Fusion* model.

1.2 Thesis Timeline

Thesis timeline for 2021-2022

April 2021	Thesis Proposal
Now-May 2021	Hierarchical Interpretable Learning for Rationale generation
May-Aug 2021	Summer Internship at AI2 on Rationalization
Aug-Dec 2021	Wrap up work on Rationalization
Jan-Feb 2022	Thesis Writing
Mar-Apr 2022	Thesis Defense

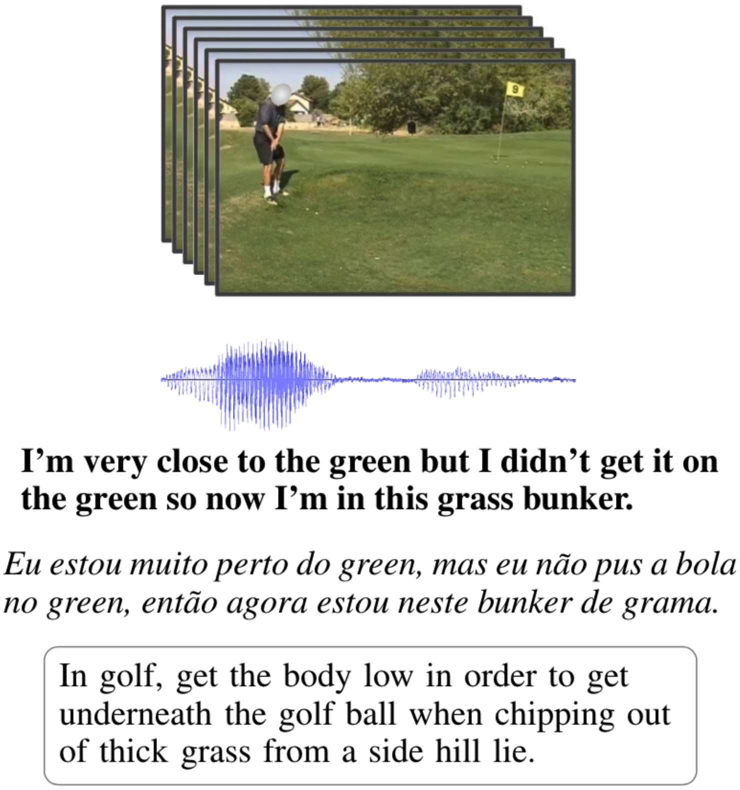
Chapter 2

Background and Review

Grounding or fusion in Multimodal learning research refers to anchoring one modality into others for joint inference. It is the computational means towards the human quality of using multiple senses at once, for example, watching, listening, and reading at once while watching a movie (Rentfrow et al., 2011).

Strongly coupled grounding exhibits in tasks where a question asked can only be answered through information contained in an image, e.g. Visual Question Answering (Antol et al., 2015). Various models have been proposed for such multimodal fusion and are often dataset and task based. Weakly coupled grounding refers to inherent inferences drawn via available data, for example, given an image of a person wearing a chef's hat and an apron, we can understand the potential genre of the video as cooking, baking, food, and scene as kitchen. Weak coupling is often achieved through semantic or representation learning while strong coupling is achieved through specifically designed modeling approaches such as bounding box grounding of an image with the corresponding referring expression.

Lack of One Dataset There has been considerable progress made across multimodal learning tasks, however, these tasks are often uncorrelated and modeled in isolation. Towards the human quality of using multiple senses at once, it might also help to model many of these tasks together, or in some form of order where the models build off of each other depending on the task at hand. There is a lack of a single dataset that has many of these modalities which could enable modeling various tasks together using a single dataset. The `How2` dataset was proposed by Sanabria et al. in 2018 which enabled, at the time, large-scale grounding for various language-oriented multimodal tasks with a 5-way parallel dataset. The following Section describes this dataset and contrasts it with some of the previously published datasets and multimodal tasks. More recently, Miech et al. collected a much larger version of a similar type of data that contain 100 Million videos.



I'm very close to the green but I didn't get it on the green so now I'm in this grass bunker.

Eu estou muito perto do green, mas eu não pus a bola no green, então agora estou neste bunker de grama.

In golf, get the body low in order to get underneath the golf ball when chipping out of thick grass from a side hill lie.

FIGURE 2.1: How2contains a large variety of instructional videos with English transcriptions, Portuguese translations, and English video summaries.

2.1 How2 Dataset

Multimodal sensory integration is an important aspect of human concept representation, language processing and reasoning (Barsalou et al., 2003). From a computational perspective, major breakthroughs in natural language processing (NLP), computer vision (CV), and automatic speech recognition (ASR) have resulted in improvements in a wide range of multimodal tasks, including visual question-answering (Antol et al., 2015), multimodal machine translation (Specia et al., 2016), visual dialogue (Das et al., 2017), and grounded ASR (Palaskar et al., 2018).

Despite these advances, state-of-the-art computational models are nowhere near integrating multiple modalities as effectively as humans. This can be partially attributed to a lack of resources that are *pervasively* multimodal: existing datasets are typically focused on a single task, *e.g.* images and text for image captioning (Chen et al., 2015), images and text for visual-question answering (Antol et al., 2015), or speech and text for ASR (Godfrey et al., 1992). These datasets play a crucial role in the development of their fields, but their single-task nature limits the collective ability to develop general purpose artificial intelligence.

Sanabria et al. introduce How2, a dataset of instructional videos paired with spoken utterances, English subtitles and their crowdsourced Portuguese translations, as well as English video summaries. The pervasive multimodality of How2 makes it an ideal resource for developing new

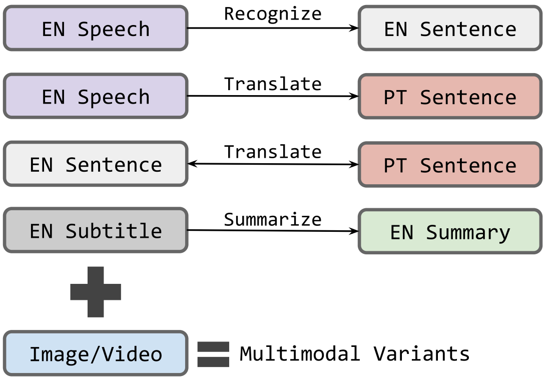


FIGURE 2.2: Modalities and possible tasks around How2: EN and PT respectively stands for English and Portuguese.

models for multimodal understanding (see Figure 2.2). In comparison to other multimodal resources, How2 is a naturally occurring dataset: neither the subtitles, nor the summaries have been crowdsourced. Furthermore, the visual content is inherently related to the spoken utterances – How2 is a dataset of people showing other people how to accomplish tasks.

Figure 2.1 shows an example from How2 in which the presenter is explaining how to play a golf shot. The English speech and subtitles are aligned with a Portuguese translation. If one only considers the text for this instance, it is unclear whether the "*green*" in the subtitles refers to the color green ("*verde*") or the golf playing surface ("*green*")[1], thus the textual context alone is not enough to disambiguate the meaning of the subtitles. However, given some additional visual context (green grass with a flag pole), or the audio context (outside with the sound of chipping a golf ball), or the sequential context of the video, multimodal models can integrate multiple inputs to understand this utterance.

The value of additional modalities can also be demonstrated in the context of ASR. Object and motion level visual cues can filter out systematic noise that co-occurs with activities. Scene information from an image can be used to learn a common auditory representational space through recordings with different environmental characteristics such as indoor vs outdoor settings (Miao and Metze, 2016a). It has also been shown that entities in an image can dynamically guide a speech recognition language model towards a more specific and relevant domain (Gupta et al., 2017a).

The How2 dataset consists of 79,114 English instructional videos from YouTube with English subtitles. The dataset consists of a total of 2,000 hours of video. Videos have an average length of 90 seconds (how, 2018) and manual Portuguese translations. This collection of videos and translations constitutes a large-scale resource for testing a substantial part of multimodal language processing methods in a real-world scenario.[2]

[1] At the time of writing, both Google Translate and Microsoft Translator incorrectly translate this sense of **green** as *verde*.

[2] The tools to download and construct the corpus are freely available at https://github.com/srvk/how2-dataset.

		Videos	Hours	Clips/Sentences
300h	train	13,168	298.2	184,949
	val	150	3.2	2,022
	test	175	3.7	2,305
	held	169	3.0	2,021
2000h	train	73,993	1,766.6	-
	val	2,965	71.3	-
	test	2,156	51.7	-

Table 2.1: Statistics of How2 dataset.

Statistic	**English**	**Portuguese**
Sentences	185K	
Words	3.50M	3.30M
Words per sentence	18.9	17.9
Unique words	57.5K	74.9K

Table 2.2: Training set statistics for English and Portuguese: the number of unique words is computed after tokenization.

An alignment process is needed to use the audio, the English subtitles, the Portuguese translations, and the video modality together. To this end, we first re-segment the English subtitles into sentences using NLTK (Loper and Bird, 2002). Then, we force-align the speech signal at the word level with an HMM-GMM pre-trained on the Wall Street Journal dataset. Finally, using the timings provided by the word alignment, we create video *clips* aligned to the initial segmented sentences. This process splits a video into a sequence of clips, aligned with the speech signal and the segmented sentences. Tables 2.1, 2.2 presents summary statistics of the *2000h* set and *300h* subset: the *val* and *test* sets can be used for early-stopping, model selection and evaluation; the *held* set is reserved for future evaluations or challenges. The total set (*i.e.* 2000h) contains around 22.5M words. The tokenized training set of *300h* subset contains around 3.8M (43K unique) and 3.6M (60K unique) words for English and Portuguese respectively. Videos are broken down into clips, as described above, with an average length of 5.8 seconds, or 20 words of spoken language.

Figures 2.3 show the LDA topic distribution and segment length analysis of the *300h* subset of the How2 dataset .

To estimate the topic diversity in How2 dataset, we ran a Latent Dirichlet Allocation (LDA) (Blei et al., 2003a) over the English subtitles. Then, we defined 22 clusters by analyzing empirical distances between videos and centroids. Finally, we applied a topic label to each cluster by analyzing the top words. Figure 2.3 shows the distribution of all videos according to each topic.

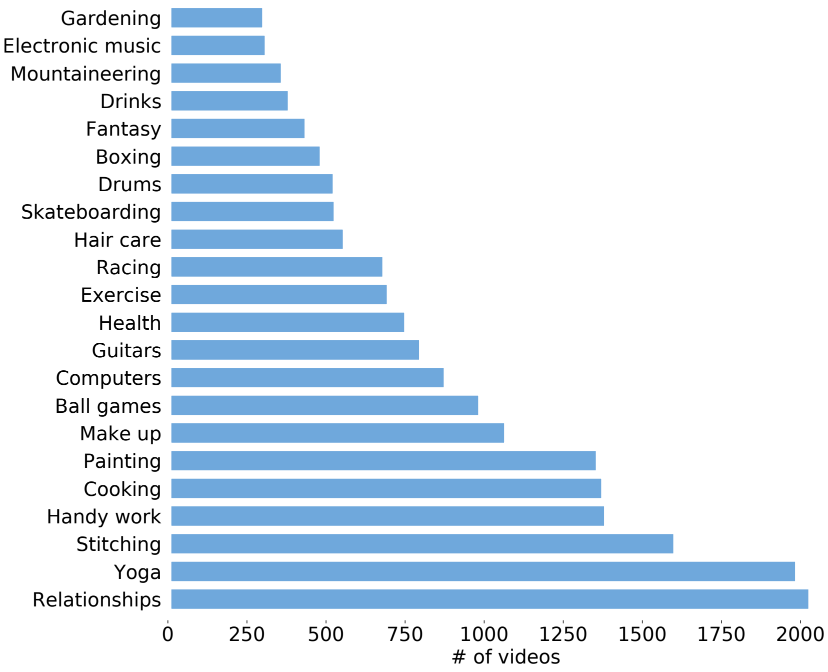


Figure 2.3: LDA topic distributions for the *300h* subset of the How2 data. The labels are manually annotated based on frequency of topic words. The overall *2000h* corpus exhibits very similar characteristics.

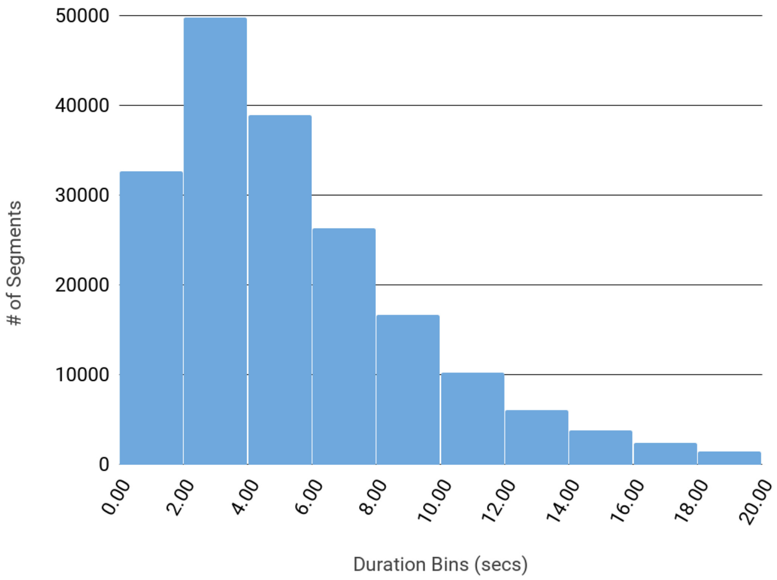


Figure 2.4: Segment durations for the *300h* subset. The overall *2000h* corpus exhibits very similar characteristics.

Features

The following features are extracted and used as representations of the various modalities of the How2 dataset .

Speech Features For speech, we extract 40-dimensional filter bank features from *16kHz* raw speech signal using a time window of *25ms* and an overlap of *10ms*. 3-dimensional pitch features are then concatenated to form the final 43-dimensional feature vectors. The speech features of a given video are further normalized using the mean and variance statistics from that specific video.

Action Features (video-level) We extract action-level video features from a 3D ResNeXt-101 (Hara et al., 2018) pre-trained on the Kinetics action recognition dataset (Kay et al., 2017) which comprises 400 different actions.

Object Features (frame-level) A ResNet-152 (He et al., 2016a) trained on ImageNet (Deng et al., 2009b) which consists of 1000 categories ranging from animals, flowers to devices and foods and so on.

Scene Features (frame-level) A ResNet-50 trained on Places365 (Zhou et al., 2017) for scene recognition with 365 categories including, but not limited to: garden, valley, studio, theater and office.

Topic distribution There are gold-labels for category of each video in the content-provider metadata that narrowly categorizes each video into 8 different topics. The most frequently occurring category according to the metadata is *Howto & Style* (39.5%) but as we are already dealing with instructional How-To videos, we wanted to look for more insight by obtaining a finer grained topic distribution. For this purpose, we clustered the entire English subtitles of a video using Latent Dirichlet Allocation (LDA) (Blei et al., 2003b). Upon analyzing the clusters with top words in each topic, inter-topic and intra-topic distances, we found that a good representation for the *300h* subset consists of 22 different topics. We hand-labeled these topics based on top words in each cluster. The relative frequency of each topic is shown in Figure 2.3.

2.2 Direction of Research

Video understanding has been approached via various tasks, either unimodal or multimodal. Most often, these tasks are uncorrelated and modeled in isolation. We propose to bring some order to this task by exploring four tasks chosen based on the complexity. Across these tasks, we demonstrate how increasingly expressive models are required to address the increase in complexity and multimodal views depending on the task. Additionally, we focus on open-vocabulary language generation tasks for possible extensions to domain-agnostic and large-scale modeling.

Free-form language generation, especially for multimodal data, is a sequential generation task conditioned on the various inputs as well as the text generated so far. Often, the vocabulary of these tasks are sub-words or word units that have an unrestricted vocabulary or a large word vocabulary, facilitating open-ended generation. For language-oriented multimodal video understanding tasks, free-form generation tasks model open-vocabulary generation beyond a closed-set classification tasks.

Instead of approaching video understanding as a set of various uncorrelated tasks, we order the tasks in increasing order of complexity and develop model architectures accordingly. Based on the task, the modality views get progressively complex. With the ordering presented in this thesis, other video understanding tasks can be arranged based on the input modalities, or model specifications, or task complexities[3].

This thesis explores grounding and multimodal learning with an emphasis on speech, audio, language, and vision inputs, for downstream tasks ranging from simple monotonic speech recognition to commonsense rationalization. In the following chapters, we first define the four tasks with the terminology, the models used, and the modality views, followed by a chapter on each of the main tasks.

[3] Tasks presented here are not comprehensive but cover a broad range of generation-based problems.

Chapter 3

Learning Tasks

Video Understanding is a broad field in itself spanning multiple tasks. In addition to this, there is no standard theory to contain the term *understanding* across multiple research areas in Machine Learning. Computer Vision research has approached this problem from the vision-focused lens – action recognition, object detection, localization, etc. Language research has approached this problem from the language generation angle – textual video summarization, question answering, sparse and dense captioning, etc. Audio and Speech modalities are often used as supplementary information for cross-modal learning as they are often tightly coupled with the temporal visual stream in a video.

To contain this broad field into smaller components in this thesis, we provide a comprehensive study with 4 main tasks that handle each of these three main modalities – audio, video, and language. We pick these learning tasks based on the theory of multimodal learning in humans (Rentfrow et al., 2011; Hattie, 2012; Brame, 2016), map them to existing machine learning tasks, and order them in increasing order of complexity. This order spans surface-level tasks such as Speech Recognition or Translation and expands to interpretable and explanatory rationale-generation tasks.

We also propose relevant model architectures based on the nature of the task (Monotonic, Non-monotonic, Abstract, and Explanatory), complexity of the task (utterance-level, video-level), the modalities involved in each (audio, video, text), and the expected outputs (transcription, translation, summarization, rationalization). In this chapter, we formally define the various terms used to construct this learning task structure, the four learning tasks, and the respective models and modalities for each.

3.1 Terminology

Monotonic Tasks A functional constraint where the input time-series sequence and output time-series sequence are dependent on each other. Between two ordered sets, a monotonic function preserves or reverses the order. For a given function, f such that $f : X \rightarrow Y$ is a set function from a collection of sets X to an ordered set Y, then f is said to be monotonic if whenever $A \subseteq B$ as elements of X, $f(A) \leqslant f(B)$.

Speech recognition is a speech-to-text task; given a window of speech frames (commonly used smallest divisible feature of speech representation) it is converted to its corresponding phonemes, characters, sub-words, words or word-pieces. These input speech frames are mapped to the corresponding textual units following a strictly monotonic constraint.

Non-monotonic Tasks For Translation tasks where the output sequence may not follow a strictly monotonic constraint. The output text can be ordered depending on the language being translated to. For speech recognition this is a strict condition across all languages, i.e. whatever is spoken is transcribed with preserved temporal order from left-to-right. For speech translation, this constraint is language-dependent. Many languages exhibit re-ordering in the translated text and the left-to-right order is not preserved.

Abstract Tasks To contrast with *Surface* tasks, we define *Abstract* tasks as those that operate at video-level time-scales rather than utterance-level. To perform *Abstract* tasks, access to video-level information across all modalities is necessary. This information is compressed, aggregated, restructured and rephrased as outputs in these tasks. Video summarization and question answering are the two *Abstract* tasks covered in this thesis.

Explanatory Tasks *Explanations or Rationales* can be useful method of evaluating *understanding* beyond abstraction tasks. They also provide further support for system-generated responses in video understanding tasks. *Explanatory* tasks refer to generation of interpretable rationales for *Abstract* tasks. An example of such tasks is the Visual Commonsense Reasoning (Zellers et al., 2019) or the Visual Question Answering with Explanations (Li et al., 2018). For *Abstract* and *Explanatory* tasks monotonicity is inherently not a constraint.

3.2 Learning Tasks

The four multimodal video understanding tasks covered in this thesis are:

I. **Speech Recognition**
Can we recognize what is being spoken? Does watching the video while listening help?

II. **Speech Translation**
Can we recognize and translate what is being spoken? Can we do it with lesser supervision?

III. **Summarization & QA**
If we have understood a video, can we summarize it? Can we answer questions?

IV. **Rationalization**
Can we provide reasoning for our answers?

These tasks are arranged in an increasing order of complexity. Speech Recognition is a monotonic task i.e. the input speech sequence and the output text sequence are bounded by a time-series monotonic constraint (left-to-right). Video speech recognition is a multimodal equivalent of this task where there is an added time-series video signal in the input. For Speech Translation, there is a monotonic constraint to some extent, but the outputs are re-ordered sequences of input speech where the extent of re-ordering depends on the output language being translated to. Speech Recognition and Translation can be thought of as *surface* tasks because of this monotonic constraint.

In contrast with Recognition and Translation, the next learning tasks covered are Summarization and Question Answering. In these tasks, video-level information needs to be compressed, aggregated, rephrased, and restructured to generate natural language summaries or answer questions based on the video. These tasks represent higher-level video understanding that abstracts information and outputs it in the form required without any monotonic constraint as in speech recognition or translation. Finally, we propose to cover the learning task of Rationalization that extends question answering to providing natural language reasoning for the generated answers. Rationalization can be considered an explanatory task that provides interpretable and observable rationales for a question answering task.

We design models for each of these tasks to satisfy these constraints and evolve the model as necessary based on the task complexities and modalities involved. These are discussed in the subsection below.

Task	Inputs	Outputs
Speech Recognition	Speech, Video	Transcripts
Speech Translation	Speech, Video, Transcripts	Portuguese Transcripts
Summarization & QA	Speech, Video, Transcripts, Questions	Summary, Answers
Rationalization	Video, Questions	Answers, Rationales

Table 3.1: Task-specific multimodal views available for each task.

3.3 Models and Modalities

Across the tasks, there are multiple views of the multimodal data available. In Table 3.1, we list the task-specific modalities (or views of a modality) available and note the ones used in the input vs. output in this work.

Based on the complexities of the four tasks and the modalities involved in each, we design specific multimodal fusion models. Across tasks, these models progressively get more complex to handle the idiosyncrasies of each. Figure 3.1 shows an overview of all four models. These models are explained in more detail in the respective Chapters.

I. **Input Fusion**
For the monotonic correlations between Speech and Text.

II. **Latent Representational Fusion**
For the non-monotonic, re-ordered sequences between Speech and Translations.

III. **Hierarchical Latent Representational Fusion**
For the video-level abstraction, compression, and re-phrasing of information as a Summary.

IV. **Hierarchical Interpretable Fusion**
For interpretable and explanatory Rationalization.

The Input Fusion model is used for video speech recognition type tasks that have a monotonic constraint. We fuse the modalities in the input (early fusion) to align video, speech, and text time-scales. But this is a restrictive model with limited modeling capacity for more complex tasks like translation where output is not strictly monotonic. For this reason, we extend the model to a Latent Representational Fusion model where the two input modalities are fused in the latent space instead of in the input. These fused multimodal representations are then used for translation.

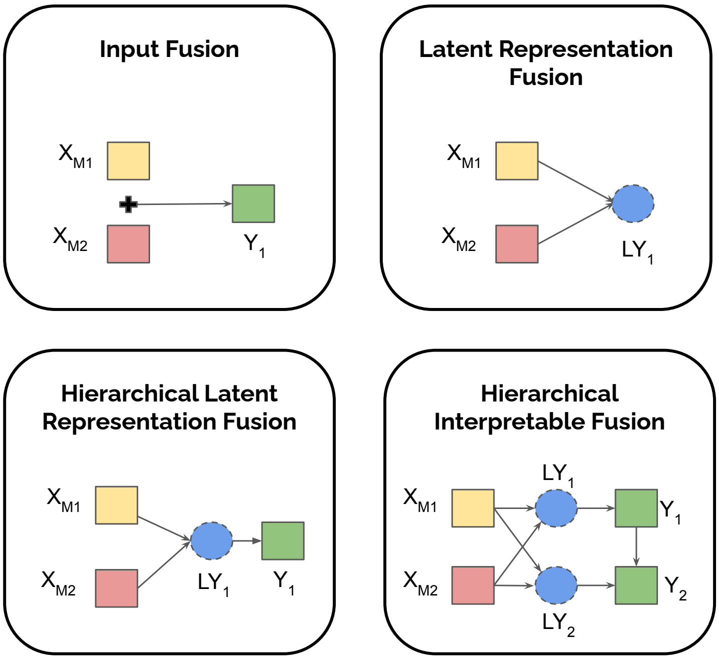


FIGURE 3.1: Overview of the model complexities across four learning tasks: Input Fusion, Latent Representation Fusion, Hierarchical Latent Representation Fusion, and Hierarchical Interpretable Fusion.

Both speech recognition and translation operate at a finer time-scale than summarization, question answering or rationalization which are the *abstract* learning tasks. These tasks need access to video-level information rather than utterance-level like *surface* tasks. We design a Hierarchical Latent Representation Fusion model for summarization and question answering that provides this video-level control for abstractive tasks. Finally, we propose a Hierarchical Interpretable Fusion model that extends the models so far to interpretable intermediate outputs for explanatory tasks. An example of utility of this model is for visual commonsense reasoning where for a given video and question, the model generates an answer as well as a rationale for why that answer is the correct answer. The Hierarchical Interpretable Fusion model is part of proposed work.

Chapter 4

Speech Recognition

4.1 Introduction

Humans are capable of processing speech by making use of multiple sensory modalities. Lip reading is a common example of audio-visual speech recognition by humans in noisy or distant scenarios, with seamless adaptation and balance between lip-reading and hearing. Although, beyond lip reading, the environment of source speech is often rich with semantic and/or acoustic context that helps us resolve ambiguities or to recall relevant words. In this Chapter, we present a novel task of end-to-end audio-visual speech recognition, and follow it with large-vocabular Acoustic2Word models that can directly map input speech features to word-units.

For audio-visual speech recognition, we propose to use multi-modal video information slightly differently than to replicate lip-reading (Chung et al., 2017) within a neural network. We propose a multimodally adapted end-to-end speech recognizer to the visual semantic concepts extracted from a *correlated visual scene* that accompanies some speech, for example in a "How-To" instructional video. If we see a person standing in a kitchen, holding sliced bread, it is likely that the person is explaining how to make a sandwich, and the acoustic conditions will be comparably clean. If a person is standing in front of an airplane, it is likely an informative video of that plane, and happening outdoors, and hence with noisier acoustics. With such semantic adaptation, we do not need constant access to the visual stream as is the case for lip-reading, and visual cues from the recording environment (indoor vs. outdoor) or the interaction between salient objects (people, instruments, vehicles, tools and equipment) can be exploited by the recognizer in various ways. We use the `How2` dataset of open-domain instructional videos for this task that are recorded in a varied acoustic environments, both indoors and outdoors (Sanabria et al., 2018).

Transcription or sub-titling of open-domain videos is a challenging domain for Automatic Speech Recognition (ASR) due to the data's challenging acoustics, variable signal processing and the essentially unrestricted domain of the data. Previous work has shown the visual channel – specifically object and scene features – can help to adapt the acoustic model (AM) and

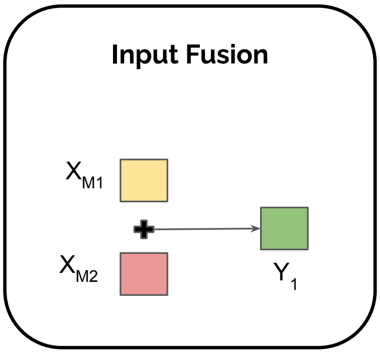


FIGURE 4.1: Illustration of the Monotonic Input Fusion technique for multimodal adaptation.

language model (LM) of a speech recognizer (Miao and Metze, 2016b; Gupta et al., 2017b). Basing off of the same hypothesis, we expanding this work to an end-to-end model that no longer requires separate AM and LM adaptation, but can jointly adapt both within a single model.

Aligning speech to word-like units is a challenge (Kamper et al., 2016; Bengio and Heigold, 2014). As a first step towards this problem, we begin by training direct acoustic-to-word speech recognition models with the aim to achieve useful speech embeddings from these trained encoders. In addition to providing a means to learn speech embeddings, Acoustic-to-Word recognition provides a straightforward solution to end-to-end speech recognition without needing external decoding, language model re-scoring or a lexicon. While character-based models offer a natural solution to the out-of-vocabulary problem, word models can be simpler to decode and may also be able to directly recognize semantically meaningful units. We analyze the encoder hidden states and the attention behavior, and show that the monotonically constrained location-aware attention naturally represents words as a single speech-word-vector, despite spanning multiple frames in the input. We finally show that the Acoustic-to-Word model also learns to automatically segment speech into words, a task that previously required careful human annotation. This property of Acoustic2Word speech recognizers was necessary for our work on semi-supervised Speech Translation in the following Chapter 5.

Monotonic Input Fusion Model

For given input modalities X_{M1} and X_{M2}, under the assumption of monotonicity, we propose an Input Fusion model that maintains input time-scale alignment between the modalities. In the case of speech and vision as the input modalities, we condition every speech frame on the corresponding visual representation to maintain monotonicity. Input Fusion is similar to standard Early Fusion in multimodal research with the main difference of fusion happening within the model, hence model-dependent, instead of model agnostic as traditional Early Fusion. Following Input Fusion, any end-to-end model can be applied to convert given inputs to output Y_1. This model is output-unit independent and can be applied to character-level or word-level outputs.

Chapter Structure

We begin this chapter by describing an end-to-end audio-visual speech recognition model applied to the How2 dataset. At the time (2017), end-to-end approaches for speech recognition were fairly new. To demonstrate the quality of the trained models, we also evaluate them on standard speech corpora and compare against prior work. We analyze the model success and failure cases and highlight the differences in standard speech recognition corpora used to train unimodal speech recognizers with multimodal speech corpora, and its corresponding effects on recognition performance. In the following Section, we focus on building direct Acoustic-to-Word Speech Recognizers that map acoustic features directly to semantically meaningful output units. Word-level output units are a common target for text-based and vision-and-language based downstream tasks. As the tasks described in this thesis often use all video modalities at once, having a common target unit space is useful. Specifically, Acoustic2Word modeling provides the necessary groundwork to perform Audio-to-Semantic Learning and Multiview Learning used for Speech Translation in the following Chapter. In this Chapter, we describe the methods necessary for Acoustic2Word modeling, and evaluate the models utility towards learning speech embeddings. The work presented in this chapter has been published previously as conference papers in Palaskar et al. 2018 and Palaskar and Metze 2018.

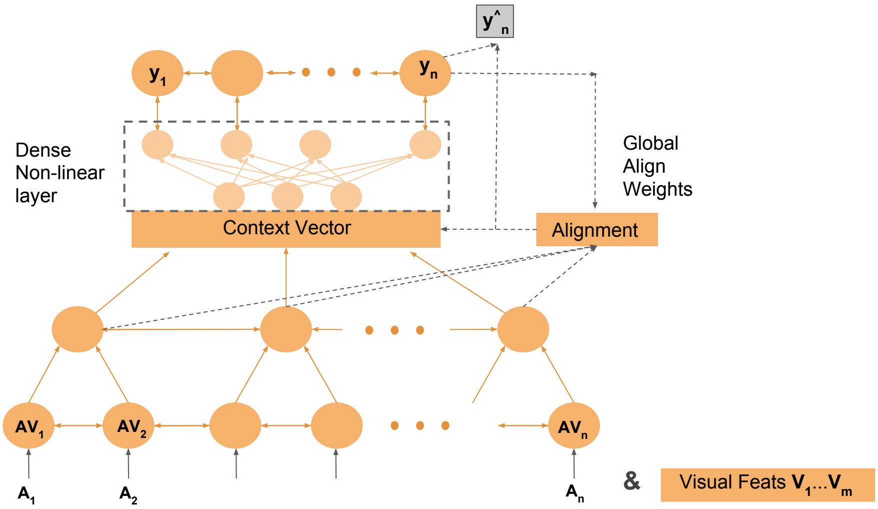


FIGURE 4.2: Audio-Visual Sequence-to-Sequence ASR.

4.2 Audio-Visual Speech Recognition

Following work builds on prior work on traditional models for audio-visual speech recognition (Miao and Metze, 2016b; Gupta et al., 2017b) and presents first results on multi-modal adaptation Sequence-to-Sequence models (Cho et al., 2014; Bahdanau et al., 2014a). We present a monotonic input fusion adaptation strategy first results with S2S model adaptation on audio-visual data. The ultimate goal of this work will be to view automatic speech recognition not primarily as the speech-to-text task, but as a process which sub-titles multi-media material removing repetitions, hesitations or corrections from spontaneous speech as required, much like "video captioning" (Vinyals et al., 2015b). We show that multi-modal adaptation helps by 2% absolute improvement in the token error rate.

While multi-modal adaptation improves recognition in such noisy datasets, we see that there is need for deeper insight into the S2S models for audio-visual speech recognition. These models behave very differently with clean, prepared datasets like WSJ than with spontaneous, noisy speech, How2. Ground truth references for the How2 data are less accurate than for WSJ; we see that this influences model training. We present insights into the differences in the output of these two approaches as these issues have not been addressed in prior work yet.

4.2.1 Model

We use the standard setup of an attention-based sequence-to-sequence (S2S) model (Chorowski et al., 2014; Bahdanau et al., 2014b) applied to audio-visual inputs. The encoder maps the input acoustic features vectors $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T)$ where $\mathbf{x}_i \in \mathcal{R}^d$ into a sequence of higher-level features $\mathbf{h} = (\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_{T'})$. The encoder is a multi-layer bi-directional Long Short Term Memory (BLSTM) RNN that is structured as a pyramid by skipping every other frame between certain encoder layers for efficient training. This reduces the length of the input from T to T'. This encoder network is analogous to the traditional acoustic model of an ASR. The decoder network is also an LSTM network that learns to model the output distribution over the next target conditioned on sequence of previous predictions i.e. $P(\mathbf{y}_l|y^*_{l-1}, y^*_{l-2}, ..., y^*_0, \mathbf{x})$ where $\mathbf{y}* = (y^*_0, y^*_1, ..., y^*_{L+1})$ is the ground-truth label sequence. This decoder network is similar to the language model in traditional ASR as it generates targets $\mathbf{y}$ from $\mathbf{h}$ using an attention mechanism. The attention model learns an alignment weight vector between the encoding $\mathbf{h}$ and the current output of decoder $\mathbf{y}_l$. At each time step, the attention module computes a context vector that is fed into the decoder together with the previous ground-truth label y^*_{l-1}.

We implement the Global Attention that learns a weighted context vector W_α calculated using the source hidden state h_s and the *current* target y_t (Luong et al., 2015). This context vector is global as it always attends to all source states s'. We compute a variable-length alignment vector α_t using: $exp(h_t^T.W_\alpha.h_s)$ and this is normalized over all input s' as $\sum_{s'} exp(h_t^T.W_\alpha.h_{s'})$. The model architecture with attention is shown in the Figure 4.2.

We use 3 layers of 512 bidirectional LSTM cells in the encoder. We use SGD with learning rate of 0.2 and decay of 0.9. We use curriculum learning (Bengio et al., 2009) for the first epoch to speed up convergence. We note that our training process is much simpler than (Chan et al., 2016; Chorowski and Jaitly, 2016; Bahdanau et al., 2016). The decoder is made of 2 layers of 512 bidirectional LSTM cells each. For decoding, we use a beam size of 5. We do not use any techniques for better decoding with WSJ as given in (Chorowski and Jaitly, 2016) but use a length normalization with How2 data (Sanabria et al., 2018), to address the length distributions variance shown in Figure 4.3. Training took 4 days with a TitanX GPU on the 90 h subset. Experiments were performed using the OpenNMT toolkit (Klein et al.).

Input Fusion The video adaptation technique we use with S2S is early fusion where 100 d visual features are concatenated with 120 d audio features giving 220 d vector for each frame. Our experiments with early fusion show that S2S benefits with this technique while CTC or DNN (Miao and Metze, 2015; Palaskar et al., 2018) does not. Caglayan et al. 2019 extend the experimentation of S2S adaptation to other fusion strategies and find that ensemble of various ASR models performs best.

4.2.2 Experimental Setup

We conduct our experiments on two data-sets, the Wall Street Journal (WSJ, SI-284, LDC93S6B and LDC94S13B), and the How2 audio-visual dataset (Sanabria et al., 2018). The How2 corpus consists of English language open-domain instructional videos that explain specific tasks like baking a cake, or nutrition habits, and have been recorded in various environments indoors and outdoors (like kitchen, or garden), usually with a portable video recorder. Ground truth transcriptions of these videos have been created by re-aligning provided sub-titles, which

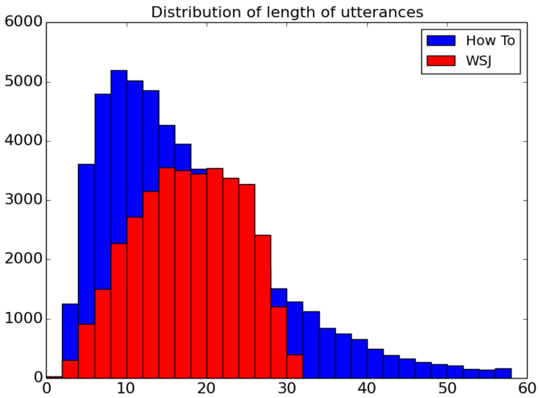


Figure 4.3: Length Distribution for the How2 and WSJ Train sets.

sometimes are mis-matched because of missed phrases, word repetitions, hesitations, and other noise and punctuation that hasn't been transcribed.

We use the 90 and 480 hour subsets of the How2 data and 87 hours of WSJ, and extract 40-dimensional MEL filter banks, with a step size of 30 ms, 3-fold oversampling of the data at 0, 10, and 20 ms offsets, and stacking 3 neighboring frames together, resulting in a 120-dimensional input vector. The 90 h subset of How2 has been selected randomly. We have a separate 4 h test set. 5% of the training data is used as dev set [1]. For WSJ, we use the eval92 test set. Both models are character-based with 43 labels/tokens: 26 alphabets, 10 digits, and special symbols for {'.', '"', '-', '/'}, space, start and end of sentence.

Figure 4.3 shows the length distributions for the How2 and WSJ train sets. This shows that for "open-domain" speech data, the distribution is less normalized when compared with prepared datasets.

Visual Features The visual features used here are the same as those in prior work with this dataset (Gupta et al., 2017b). We extract object and place/scene features from pre-trained CNNs and perform dimensionality reduction to obtain 100 dimension features. As described above, the data contain indoor and outdoor recordings of instructional videos where object and place features are most relevant. We use these features to infer acoustic and language information from the scene where the utterance has been recorded.

4.2.3 Results

In Table 4.1, we present the effect of visual adaptation on the Token Error Rate (TER) for speech models. Adaptation with visual features helps improve the absolute TERs by 1.6% using the S2S model. Using length norm described below, TER further improves to 2%. This is a significant improvement for speech recognition with S2S models. The test set named 'dev'

[1] These are early subsets of the How2 datasets created and used in prior work (Miao and Metze, 2016b; Miao et al., 2014). The official release of the How2 dataset (how, 2018) processed these initial splits before release.

is a tougher set than the 'test' set in the How2 dataset. This work was extended by Caglayan et al. to other monotonic fusion approaches for S2S models, namely the Visual Adaptive Fusion method leading to an absolute improvement of 1.4% word error rate, which is a similar degree of improvement as with the Input Fusion model.

	TER dev	**TER test**
Audio-only	18.4	16.3
Audio-Visual	**16.8**	**15.7**

TABLE 4.1: Results for Audio-only and Audio-Visual adaptation with the How2 data.

	S2S Model
WSJ	7.9
How2	15.3

TABLE 4.2: TER of the S2S model on WSJ (eval92) and How2 (test set).

In Table 4.2, TER with S2S on WSJ is a strong baseline compared to prior work (Bahdanau et al., 2016; Kim et al., 2017). We see a huge disparity in ASR for clean prepared data (WSJ) and real application data (How2) as discussed with Figure 4.4.

Spoken	now it does only say for do- or doesn't even say for dogs or cats it's neither
Reference	now **doesn't even** say dogs or cats **it says** neither
Audio-only S2S	now it does**n't we** say for **a** dog **or that use a** dogs or cats **so is night or**
Audio-Visual S2S	now it does**n't leave safer** dog **or it does use a** dogs or cat **so in night or**

TABLE 4.3: Typical transcription on How2 test set: S2S model keeps to the style of the reference which is an abstraction of the spoken content. Currently, there is little semantic difference between regular and adapted (AV) S2S output.

Figure 4.4 compares reference length to hypothesis length for 40 short and long WSJ and How2test utterances. On WSJ, the range of lengths of short and long utterances are similar, and reference and hypothesis follow each other closely. On How2, hypothesis prediction is very unstable and the model makes a lot of mistakes, even breaks completely at times. Length of the hypothesis is greater than the length of reference for short utterances, while it is lesser for longer utterances. As seen from the example in Table 4.3, the output of the S2S model is much closer to the reference transcript. The model learns a form of length normalization over the entire dataset hence performs badly on short and long utterances. We use the length normalization factor during decoding to stabilize the output of the S2S model and get absolute improvements of 2% (dev) and 1% (test) for the non-adapted case, which is slightly better than the adapted case and shows that adaptation stabilizes model performance.

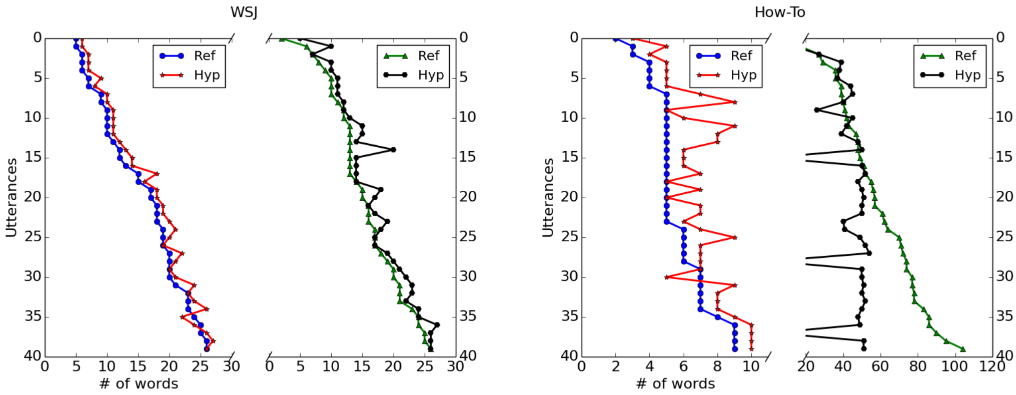


FIGURE 4.4: Length normalization by S2S for WSJ and How2

4.3 Acoustic-to-Word Speech Recognition

4.3.1 Model

Our S2S model is similar in structure to the Listen, Attend and Spell model (Chan et al., 2016) which consists of 3 components: the encoder network, a decoder network and an attention model. The difference between the S2S model used for audio-visual speech recognition described earlier is the type of attention mechanism used: location-aware attention. In this work, $y_i^* \in \mathcal{U}$ can be a token from a character, sub-word or word vocabulary.

Location-aware Attention We use a location-aware attention mechanism (Chorowski et al., 2015) that enforces monotonicity in the alignments, which may be beneficial for speech recognition. To do so, the location-aware attention applies a convolution across time to the attention of previous time step using trainable filters. This convolved attention feature is used for calculating the attention for the current time step. We apply a one-dimensional convolution $\mathcal{K}$ along the input feature axis t to get a set of T features $\{\mathbf{f}\}_{t=1}^{T}$ described as follows:

$$\begin{aligned}\{\mathbf{f}\}_{t=1}^{T} &= \mathcal{K} * \mathbf{a}_{l-1}\\ e_{lt} &= \mathbf{g}^{\mathrm{T}}\tanh\left(\mathrm{Lin}(\mathbf{y}_{l-1}) + \mathrm{Lin}(\mathbf{h}) + \mathrm{LinB}(\mathbf{f}_t)\right)\\ a_{lt} &= \mathrm{Softmax}(\{e_{lt}\}_{t=1}^{T})\end{aligned}$$

where $\mathbf{a}_{l-1} = [a_{l-1,1}, ..., a_{l-1,T}]^{\mathrm{T}}$, $\mathbf{g}$ is a learnable vector parameter, $\{e_{lt}\}_{t=1}^{T}$ is a T-dimensional vector, Lin() is a linear layer with learnable matrix parameters without bias vectors, LinB() is a linear layer with learnable matrix and bias parameters.

The S2S model is trained by optimizing the cross entropy loss function which maximizes the log-likelihood of the training data. We use beam search to perform inference. We also apply unigram label smoothing that distributes the probability of most-probable token to prevent the over-confidence of the model (Pereyra et al., 2017; Chorowski and Jaitly, 2016).

4.3.2 Experimental Setup

We use the standard 300-hour Switchboard corpus (SW, LDC97S62) (Godfrey et al., 1992) which consists of 2,430 two-sided telephonic conversations between 500 different speakers and contains 3 million words of text. We evaluate on the HUB5 eval2000 (LDC2002S09, LDC2002T43) containing Switchboard subset similar to training data and CallHome (CH) subset that is a tougher set. There are 196,656 total utterances out of which we use the first 4,000 utterances as a validation set. Our input features are 80-dimensional log-mel filter banks normalized with per-speaker mean and variance. We also use 3-dimensional pitch features.

We present three different types of target units for speech recognition: characters, Byte-Pair Encoding units (BPE) (Gage, 1994; Sennrich et al., 2016) and words. BPE units are generally longer than characters and shorter than words. The character vocabulary is made of 46 units containing 26 letters, 10 digits, and other frequently occurring special symbols. We use 12k BPE operations to get comparable vocabulary with word level models. We finally present a large-vocabulary model made of all 29,874 unique words in the Switchboard set. The vocabularies also contain non-language special symbols that denote noise, vocalized-noise and laughter. We train character and word level RNN language models on the Switchboard + Fisher (LDC2004T19) (Cieri et al., 2004) transcripts as is the common practice for this data.

Our encoder consists of 6 layers each with 320 bi-directional LSTM cells (Hochreiter and Schmidhuber, 1997). The second and third layer skip every other frame to get a reduction of $T/4$ in input frames. We use the AdaDelta (Zeiler, 2012) optimizer. The location-aware attention convolution uses 10 filters with width 100. We use a projection layer of 320 dimensions after each layer of the encoder. Our decoder is a single layer LSTM containing 300 cells. We initialize all parameters uniformly within $[-0.1, 0.1]$ unless otherwise specified. We use unigram label smoothing with weight 0.05. The beam size used for all experiments is 10. We use the ESPnet toolkit (Kim et al., 2017; Watanabe et al., 2017) for our experiments.

4.3.3 Results

We first compare a character-level S2S model with previously published Connectionist Temporal Classification (CTC) (Graves et al., 2006) model, both of which were considered the state-of-the-art model approaches at that time. Character-level evaluation and comparison will provide a better perspective on S2S and CTC comparison, as well as the corresponding difference in performance when the vocabulary is changed to word-level. Note that character-level modeling was the widely-used vocabulary at the time of this publication.

Our models obtained the best Word Error Rate (WER) of that time in both SW (5% absolute improvement without LM) and CH test sets among the previous S2S models (Lu et al., 2016; Zenkel et al., 2017; Toshniwal et al., 2017). We also perform better than previous CTC models (Hannun et al., 2014; Zweig et al., 2017; Audhkhasi et al., 2017) in the SW test set and the difference in the CH set is minor. Furthermore, we observe a 13% relative improvement in our results on the SW subset by using an RNNLM with shallow fusion (Gulcehre et al., 2015) which is trained at the character and word level. The best absolute WER at character-level modeling is 15.6% on the SW test set and 31% on the CH test set. Following the publication of this work, most recently, the state-of-the-art on this dataset has been reduced further to 5% and 8% respectively (Chan et al., 2021) by training on more data and by using large-scale models with a variety of training tricks (Park et al., 2019; Baevski et al., 2020) developed over the last few years. This latest result uses more than 5000 hours of speech for training as compared with 300 hours used in this work.

Next, we compare the A2W models with prior work using BPE and word units. We start by restricting the vocabulary to words occurring at least 5 times *(Word >= 5)* in the training set before moving on to complete vocabulary A2W modeling. *Word >= 5* leads to a vocabulary of 11069 words with an OOV rate of 2.3% for the CH test set. This model leads to an absolute WER of 23% on SW and 37.2% on CH. To address this high OOV rate, we try to match the word vocabulary by an equivalent BPE vocabulary of 12k merge operations. This model performs better than the word model as expected: 21.3% and 35.7% respectively. We also experiment with initializing the word>=5 model with a pretrained character model (similar to Audhkhasi et al. 2017) for better convergence and observe slight improvements: 22.4% and 36.1%.

We proceed to the large vocabulary speech recognition model made of all the words in the training set. This model performs better than our previous word models which may be due to absence of the frequently occurring OOV token. We get an absolute improvement of 4% in SW and 11% in CH subsets over the previously published number on such large vocabulary recognition (Lu et al., 2016) with or without a language model. Similarly, Chen et al. also perform large vocabulary recognition and our model shows 9% and 4% absolute improvement on the two test sets respectively.

Ideally, S2S A2W model does not need a separate language model as it directly predicts a sequence of words using the decoder LSTM. But as the LM is trained on a larger text corpus, we integrate it to check its effect and do not observe improvements as large as the character model.

Our work is not directly comparable with other S2S or CTC models that use BPE units or a smaller vocabulary (Audhkhasi et al., 2017; Zeyer et al., 2018) as the goal here is whole-word recognition to predict semantically meaningful units. This work is necessary for the Audio-to-Semantic learning and consequently Speech Translation work described in the following Chapter 5. With these results, we demonstrate our character-level and word-level models performed well as compared to the prior work published at that time. Irrespective of that, absolute WER performance is not the focus of this work. In the sections below, we describe the benefits of an A2W model for automatic speech segmentation and towards learning contextual acoustic word embedding.

4.3.4 Automatic Word Segmentation

In the following two sections we analyze the behavior of S2S models, specifically for the A2W recognition task. We analyze attention in the decoder and the hidden representations of the encoder.

Human Annotated Word Boundaries in SWBD. NXT Switchboard Annotations (LDC 2009T26) are a subset of the Switchboard corpus (LDC97S62) containing 1 million words that were annotated for syntactic structure and disfluencies as part of the Penn Treebank project. This subset contains human annotated word-level forced alignments that mark the beginning and end of each word in the utterance in time [2]. In the following sections, we analyze attention behavior of the A2W model and the speech-word-vectors obtained from it. To do this analysis, we need groundtruth word-level segmentations and this corpus is a good match.

From NXT Switchboard, we choose those utterances that are also present in the Treebank-3 (LDC99T42) corpus. The speech in this corpus is re-segmented to match the sentences in Treebank-3. We filter out utterances with less than 3 words resulting in 67,654 utterances in total. This is divided into 56,100 train, 5,829 validation and 5,725 test sets. We train a separate A2W model with this data using the same setup without using any explicit information about word-segments. We only train on this dataset to avoid introducing a more variability in our analysis, i.e. are the segmentations due to our model or due to training with a larger corpus (SW 300h)? In our setup, we split compound words into two words (eg. they‘re → they and ‘re).

Attention Behavior

In Figure 4.5 we plot the attention of a sample utterance from our validation set of the corpus. We notice that the attention is very peaky and focuses only on certain frames in the input although generally a word spans multiple input frames.

To understand this behavior of the model, let us revisit the location-aware attention. The location-aware attention is useful in speech to enforce a monotonic alignment between source and target. It does so by convolving the previous attention vector along input time-steps and feeding it as another input parameter while calculating attention of the current time step. This way, the model is informed where to pay attention “next” and would mostly look in the “future” to make a prediction.

As this model is trained towards word-units and the attention is focused only on certain frames, we speculate that the hidden states corresponding to those frames are the speech-word-vectors

[2] http://groups.inf.ed.ac.uk/switchboard/structure.html

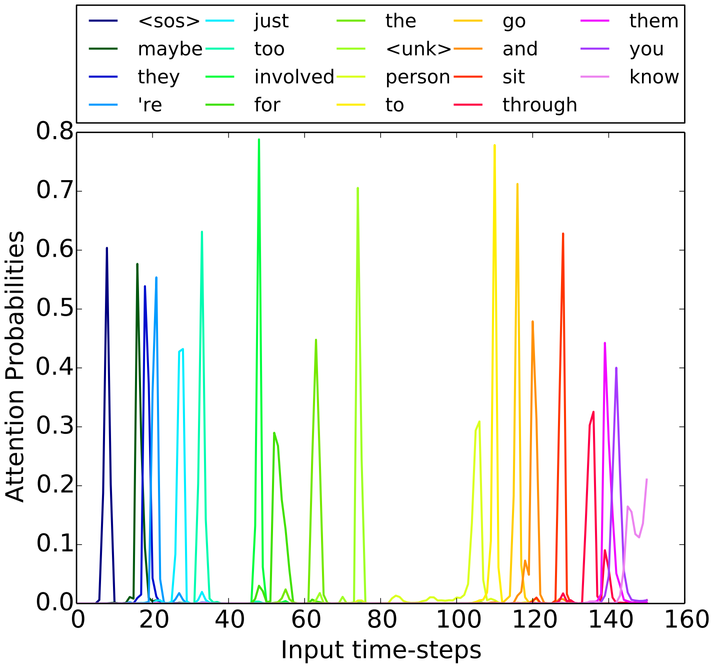


FIGURE 4.5: Attention visualization for a sample utterance from the validation set shows highly localized attention for a word-level S2S model

for those words. Here, we are able to extract speech-word-vectors from an end-to-end model trained for direct word recognition without the need of any predefined forced-alignments. The size of these embeddings is equal to the number of RNN cells in the last layer of the encoder.

Automatic Segmentation of Speech into Words

Given that the attention is highly localized, we attempt to quantify whether the attention weights corresponded to actual word boundaries. From the Switchboard NXT dataset, we chose all utterances (train, validation and test) for which we have 0% WER during testing. 39% of the total utterances have 0 WER. We perform decoding with beam size 1 here. We converted the human-annotated forced-alignments to their corresponding frame numbers using the 10ms frame rate of our model. The predicted frame number is calculated from the attention distribution shown in Figure 4.5 as follows. The input frame with the max attention probability is chosen as the predicted frame for the word. The frame error is calculated at each word level by taking an absolute difference between the predicted and grouthtruth frame number. A positive difference means the predicted frame was after the groundtruth alignment, and a negative difference means that it was before. We average this frame error for all words in all utterances (171073 words). An example of this computation is

	Avg. Frame Error		
	Train	**Val**	**Test**
W/o Last Word - Mean	0	-0.08	-0.01
W/o Last Word - Std Dev	3.7	3.3	2.0
All Words - Mean	0.4	0.3	0.3
All Words - Std Dev	10.1	9.8	10.5

Table 4.4: Average frame error mean and standard deviation (std dev.) between groundtruth forced-alignments and S2S word segment prediction

$$\begin{aligned}\text{Predicted} &= [988, 1008, 1012, 1044, 1092]\\ \text{Groundtruth} &= [988, 1005, 1013, 1042, 1100]\\ \text{and Frame Error} &= [0, +3, -1, +2, -8]\end{aligned}$$

The attention weights for the last word predicted in the sequence is often most erroneous. As an example, in Figure 4.5 we see that “know”, the last word, has a distributed attention weight, and has the least probability value (approximately 0.2) compared to other words. For better understanding, we also compute frame errors without considering the last word of every utterance.

We compute the mean and standard deviation of frame errors for all words. During training, we use a pyramidal encoder that reduces the input frame lengths by a factor of 4. Hence, while computing mean and standard deviation of frame errors, we scale them by 4 as well for fair comparison. The standard deviation of frame error without including last word is 3.6 frames after the groundtruth. For a word-based model, this is an encouraging result as usually a character unit spans 7 (or 1.75 frames after a pyramidal encoder) and a word would span many more.

Why does attention focus on the end of word? The optimization task in A2W recognition is to map a sequence of input frames (usually larger number of input frames than in character or BPE prediction models) to a sequence of target words. During training, the model learns where word boundaries occur by recognizing the attention distribution that leads to highest probability of generating the correct output. The bi-directional LSTM in the encoder has access to the past as well as future input. Therefore, the encoder learns to look into the future to recognize where a different word is beginning, and the BLSTM would hold richest embeddings in the unit corresponding to each of frame of the current word. We investigate the encoder embeddings in the next section in more detail. It is also important to note that the location-aware attention constrains the model to only look into the future, and not the past, which would push the boundaries towards word ends rather than beginnings. Hence, the attention mechanism learns to focus mostly on the word boundaries.

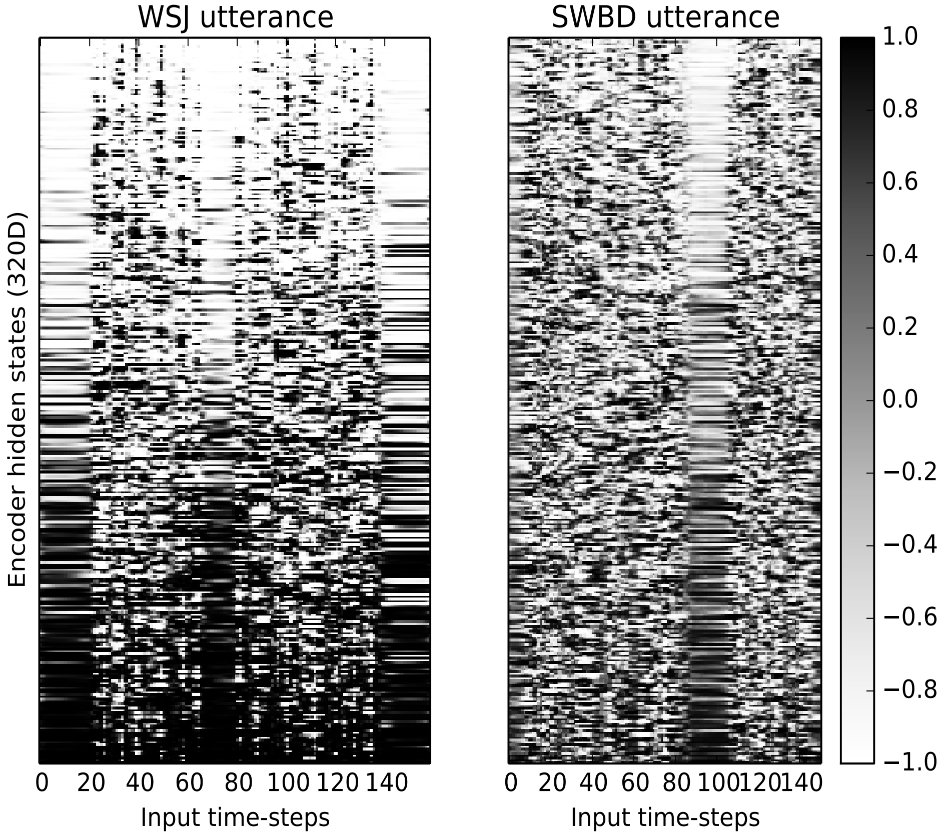


FIGURE 4.6: Encoder hidden state visualization for WSJ (acoustically clean data) and SWBD (acoustically noisier). Visualization shows encoder activations across input time frames.

We obtain a context vector from the attention mechanism that is a weighted sum of the encoder hidden states. Following this peaky nature of the attention mechanism, we expect to see certain patterns reflected in the encoder embeddings. This is explored in the following section.

Speech Embeddings

We train a similar A2W model on the Wall Street Journal corpus (WSJ, LDC93S6B and LDC 94S13B) which comprises about 90 hours of read speech in clean acoustic environments with a close-talk microphone. This dataset has about 300 different speakers in the train, validation (dev93) and test (eval92) sets. WSJ is sampled at 16kHz while SWBD is sampled at 8kHz and we upsample SWBD to 16kHz for implementation reasons. We bring the readers attention to these major differences in acoustic and speaker variability and domain of the data in WSJ and SWBD. In Figure 4.6 we visualize the encoder hidden states for sample utterances from the validation sets of WSJ (4k8c030h) and SWBD (same as in Figure 4.5). We train a WSJ model to compare hidden state activations of the noisier SWBD dataset with a clean WSJ dataset as we expect the activation patterns to be clearer and more interpretable in the cleaner dataset. The hidden state dimension here is same as the number of BLSTM cells in the last encoder layer (320D). For this visualization, we sort the hidden states of the encoder in an ascending order of total activation over time. We use a tanh non-linearity hence all values range from -1 to +1. We note that there are three types of patterns to observe in these activations: 1) stable horizontal lines, 2) disruptions, and 3) vertical dashed-line pattern across encoder hidden states (Y-axis) within the disruptions. Pattern 3 is easier to notice in the WSJ activations.

Upon listening to these utterances, we found that stable horizontal lines (pattern 1) corresponds to silence in the utterance, while disruptions (pattern 2) corresponds to the speech. We observe similar patterns identifying speech and non-speech in both WSJ and SWBD. From this, we understand that the model has learned to detect and segment pauses in speech. As WSJ is the acoustically cleaner corpus with less variability, "silence" acoustics are stable and repetitive throughout, which is what we observe in the beginning, middle and end of the WSJ utterance– while the SWBD "silence" activations look different. In WSJ, we can further identify multiple vertical dashed-line patterns across all encoder hidden states (i.e. Y-axis; pattern 3). This pattern is formed by encoder units turning on and off (+1, -1) when a word boundary is reached. This particular WSJ utterance has 15 words and we observe 15 vertical dashed-line patterns in the activations. This further reinstates that we are able to represent multiple frames of speech using single 320D speech-word-vectors. Pattern 3 is tougher to spot in SWBD comparatively but still noticeable; it might need more training data or better regularization with this data to obtain similar properties as the WSJ model.

4.4 Chapter Conclusion

We present a monotonically motivated Input Fusion model for audio-visual end-to-end speech recognition. We establish the efficacy of this model on a standard speech recognition corpus, WSJ, and extend it to a new open-domain noisy multimodal speech recognition dataset, How2. We find significant gains of over 1.6% absolute improvement in the token error rates with the audio-visual model. We delve deeper into the modeling capabilities and limitations of this model by analysis of the How2 datasetin comparison with standard, clean, WSJ corpus. We extend this analyses to compare the failure cases of model generated hypotheses for the noisy How2corpus compared with the WSJ corpus.

To extend this model to end-to-end word-level speech recognition we build direct Acoustic2Word (A2W) models. The motivation for this word-level speech recognition is to perform direct audio-to-semantic tasks addressed in further chapters. We build strong A2W models, comparing on standard widely-used speech corpora. We further investigate the power of these direct A2W models by analyzing monotonic location-aware attention mechanism and find that the model learns to automatically segment speech frames into individual words, leading to clearly identifying acoustic word embeddings. In the next chapter, we make use of these inherently learnt speech embeddings for semi-supervised learning.

Chapter 5

Speech Translation

5.1 Introduction

In this Chapter, we present semi-supervised multimodal Speech Translation with the limited supervision obtained through multimodal inputs. Speech Translation is a language-dependent re-ordering task where the input time-series speech sequence is converted to the corresponding re-ordered text sequence. Speech Translation has been approached by a cascaded pipeline approach with speech-to-text followed by text-to-text translation (Pham et al., 2019; Wu et al., 2019; Inaguma et al., 2020) or via end-to-end models (Inaguma et al., 2020; Guo et al., 2020), all in a fully supervised setting. Here, we utilize the inherent supervision available through multiple views of the same data to build semi-supervised speech translation systems that first learn multimodal representations, that are then used for a retrieval based speech translation task.

Semi-supervised learning is preferred over fully supervised learning in scenarios where data annotation is limited, expensive, or unavailable. In contrast, access to multiple views of the same data, or multimodal data, is more readily available. Specifically, data from instructional videos such as the How2 dataset is available in large quantities (Sanabria et al., 2018; Abu-El-Haija et al., 2016; Miech et al., 2019). As described earlier in Chapter 2, video data often comes with multiple views of the same data point: the speech, audio, video itself, English transcription, metadata, and possibly, auto-generated transcriptions in other languages. Using well-established representation learning method for such multiview data, Cannonical Correlation Analysis (CCA) (Hotelling, 1992), we apply it to such noisy large-scale data and demonstrate its utility in downstream tasks. This method is universal and can be applied to multiple downstream tasks such as speech recognition, text translation, and speech translation.

Apart from the noisiness of the data, application of this technique to such open-domain data involves solving multiple challenges before it can be applied to Speech Translation. CCA requires initial independently learned representations for all modalities. For speech/audio, no such method exists to learn contextual acoustic embeddings at scale. Prior work required clean annotated word boundaries or word-level speech (He et al., 2016b; Kamper et al., 2016). Secondly, the individual modality representations need to be aligned at the same time-scale, i.e. character embeddings, word embeddings, sentence embeddings, etc. Speech is often at phoneme or character-levels, text is at word-level, and vision can be at a word-level as well (annotated object/scene/action labels). To address both these issues, we develop a method to learn contextual acoustic word embeddings that bring speech/audio to the word-level embeddings.

Using the model described in Section 4.3, we build a Contextual Acoustic Word Embedding model that does not require manual alignment of word boundaries, but can instead recognize them on-the-fly (Palaskar and Metze, 2018). With this automatic segmentation, we can now extract the useful speech representations for each word (Palaskar et al., 2019b). With this technique, we bring the speech, text, and visual modality views to word-level latent alignment. These modality-specific representations are then used for further joint multi-view multimodal representation learning.

We explore an advanced, correlation-based representation learning method, Deep Generalized Canonical Correlation Analysis (DGCCA) (Arora and Livescu, 2013), on a 4-way parallel, multimodal dataset, and assess the quality of the learned representations on retrieval-based tasks. We show that the proposed approach produces rich representations that capture most of the information shared across views. Our best models for speech and textual modalities achieve retrieval rates from 70.7% to 96.9% on open-domain, user-generated instructional videos. This shows it is possible to learn reliable representations across disparate, unaligned and noisy modalities, and encourages using the proposed approach on larger datasets.

Latent Representation Fusion Model

The Input Fusion model provides an utterance-level monotonic adaptation with the input modalities X_{M1} and X_{M2}. We expand this model from strictly monotonic input fusion to a latent-space fusion, LY_1. The Latent Representation Fusion model relaxes the monotonic constraint allowing fusion for re-ordering tasks like speech translation. For speech translation, the monotonicity between input speech and output translations is language dependent, and enforcing strict monotonicity (for e.g. via input fusion) is not optimal. The Latent Representation Fusion model maintains the time-resolution of input speech (and video) at the utterance-level, and has the flexibility of output re-ordering in the latent space which was not possible in the Input Fusion model.

The Latent Representation Fusion model, is further expanded to perform semi-supervised modeling by leveraging information from the multiple multimodal views available. Using the DGCCA model for representation learning mentioned above, we train generalized multimodal representations that fuse the various input multimodal views into a single latent representation LY_1. We use retrieval-based evaluation for this model that also performs the downstream

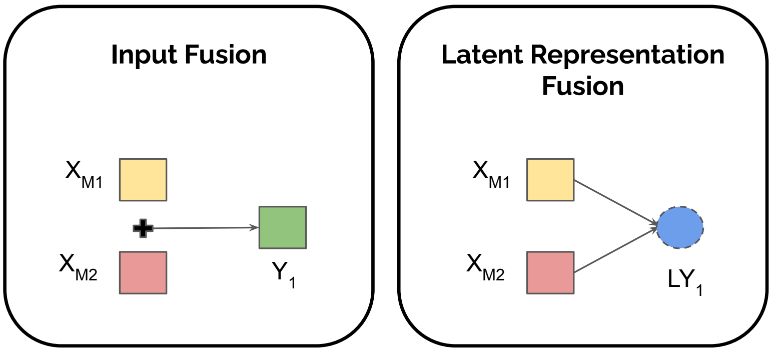


FIGURE 5.1: We build on top of the existing Input Fusion model for semi-supervised Speech Translation. We present the Latent Representation Fusion model that learns a multimodal latent representation LY_1, and uses that via retrieval for downstream tasks.

task; given the generalized multiview multimodal representation LY_1, we retrieve the corresponding translation representation $Y_1^{retrieval}$.

Chapter Structure

The work in this Chapter is structured in two main parts: (1) learning contextual acoustic word embeddings (a novel method to represent speech by word-level semantically meaningful embeddings), followed by (2) using these embeddings to perform semi-supervised multimodal Speech Translation. We begin by describing the approach for Audio-to-Semantic learning, the model used, and the evaluation of the quality of learned embeddings on 16 benchmark embedding evaluation tasks. In the following Section, we describe the multiview learning method for semi-supervised Speech Translation. This work presents one of the first applications of multiview learning to open-domain video data, the `How2` dataset . We first perform implicit evaluation of the effectiveness of the proposed approach on such data, followed by explicit evaluation on the Speech Translation downstream task. The work presented in this chapter was published in Palaskar et al. 2019b and Holzenberger et al. 2019.

5.2 Contextual Acoustic Word Embeddings

5.2.1 Audio-to-Semantic Learning

The task of learning fixed-size representations for variable length data like words or sentences, either text or speech-based, is an interesting problem and a focus of much current research. In the natural language processing community, methods like word2vec (Mikolov et al., 2013), GLoVE (Pennington et al., 2014), CoVe (McCann et al., 2017) and ELMo (Peters et al., 2018) have become increasingly popular, due to their utility in several natural language processing tasks. Similar research has progressed in the speech recognition community, where however the input is a sequence of short-term audio features, rather than words or characters. Therefore, the variability in speakers, acoustics or microphones for different occurrences of the same word or sentence adds to the challenge.

Prior work towards the problem of learning word representations from variable length acoustic frames involved either providing word boundaries to align speech and text (Chung and Glass, 2018), or chunking ("chopping" or "padding") input speech into fixed-length segments that usually span only one word (Kamper et al., 2016; Bengio and Heigold, 2014; Harwath and Glass, 2015; He et al., 2016b). Since these techniques learn acoustic word embeddings from audio fragment and word pairs obtained via a given segmentation of the audio data, they ignore the specific audio context associated with a particular word. So the resulting word embeddings do not capture the contextual dependencies in speech. In contrast, our work constructs individual acoustic word embeddings grounded in utterance-level acoustics.

We present different methods of obtaining acoustic word embeddings from an attention-based sequence-to-sequence model (Sutskever et al., 2014a; Chan et al., 2016; Chorowski et al., 2015) trained for direct Acoustic-to-Word (A2W) speech recognition (Palaskar and Metze, 2018). Using this model, we jointly learn to automatically segment and classify input speech into individual words, hence getting rid of the problem of chunking or requiring pre-defined word boundaries. As our A2W model is trained at the utterance level, we show that we can not only learn acoustic word embeddings, but also learn them in the proper context of their containing sentence. We also evaluate our contextual acoustic word embeddings on a spoken language understanding task, demonstrating that they can be useful in non-transcription downstream tasks.

Our main contributions are the following: (1) We demonstrate the usability of attention not only for aligning words to acoustic frames without any forced alignment but also for constructing Contextual Acoustic Word Embeddings (CAWE). (2) We demonstrate that our methods to construct word representations (CAWE) directly from a speech recognition model are highly competitive with the text-based word2vec embeddings (Mikolov et al., 2013), as evaluated on 16 standard sentence evaluation benchmarks. (3) We demonstrate the utility of CAWE on a speech-based downstream task of Spoken Language Understanding showing that pretrained

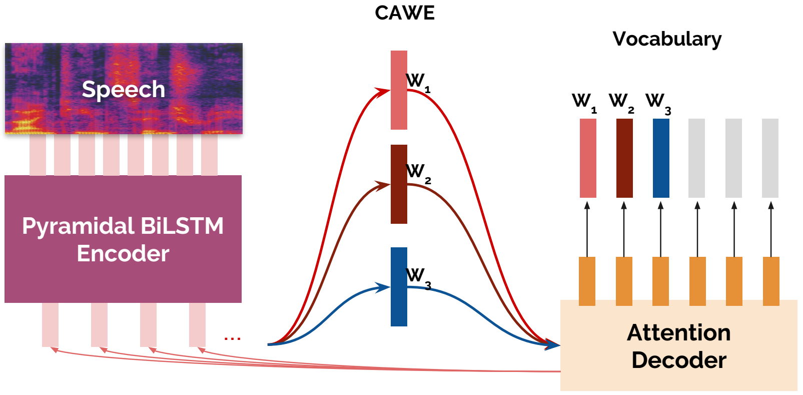


FIGURE 5.2: A2W model with the CAWE representations obtained by combining the encoders representations and attention weights.

speech models could be used for transfer learning similar to VGG in vision (Simonyan and Zisserman, 2014b) or CoVe in natural language understanding (McCann et al., 2017).

5.2.2 Model

Our S2S model is similar in structure to the Listen, Attend and Spell model (Chan et al., 2016) which consists of 3 components: the encoder network, a decoder network and an attention model. The encoder maps the input acoustic features vectors $\mathbf{a} = (\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_T)$ where $\mathbf{a}_i \in \mathcal{R}^d$, into a sequence of higher-level features $\mathbf{h} = (\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_{T'})$. The encoder is a pyramidal (sub-sampling) multi-layer bi-directional Long Short Term Memory (BLSTM) network. The decoder network is also an LSTM network that learns to model the output distribution over the next target conditioned on sequence of previous predictions i.e. $P(\mathbf{y}_l | y^*_{l-1}, y^*_{l-2}, ..., y^*_0, \mathbf{x})$ where $\mathbf{y}* = (y^*_0, y^*_1, ..., y^*_{L+1})$ is the ground-truth label sequence. In this work, $y^*_i \in \mathcal{U}$ is from a word vocabulary. This decoder generates targets $\mathbf{y}$ from $\mathbf{h}$ using an attention mechanism.

We use the location-aware attention mechanism (Chorowski et al., 2015) that enforces monotonicity in the alignments by applying a convolution across time to the attention of previous time step. This convolved attention feature is used for calculating the attention for the current time step which leads to a peaky distribution (Chorowski et al., 2015; Palaskar and Metze, 2018). Our model follows the same experimental setup and model hyper-parameters as the word-based models described in our previous work (Palaskar and Metze, 2018) with the difference of learning 300 dimensional acoustic feature vectors instead of 320 dimensional.

We now describe our method to obtain the acoustic word embeddings from the end-to-end trained speech recognition system. The model is as shown in Figure 5.2 where the embeddings are constructed using the hidden representations obtained from the encoder and the attention weights from the decoder. Our method of constructing "contextual" acoustic word embeddings is similar to a method proposed for text embeddings, CoVe (McCann et al., 2017).

The main challenge that separates our method from CoVe (McCann et al., 2017) in learning embeddings from a supervised task, is the problem of alignment between input speech and output words. We use the location-aware attention mechanism that has the property to assign higher probability to certain frames leading to a peaky attention distribution. We exploit this property of location-aware attention in an A2W model to automatically segment continuous speech into words as shown in our previous work (Palaskar and Metze, 2018), and then use this segmentation to obtain word embeddings. Below, we formalize this process of constructing contextual acoustic word embeddings.

Intuitively, attention weights on the acoustic frames hidden representations reflect their importance in classifying a particular word. They thereby provide a correspondence between the frame and the word within a given acoustic context. We can thus construct word representations by weighing the hidden representations of these acoustic frames in terms of their importance to the word i.e. the attention weight. We show this in the Figure 5.2 wherein the hidden representations and their attention weights are colored according to their correspondence with a particular word.

Given that a_j represents the acoustic frame j, let $encoder(a_j)$ represent the higher-level features obtained for the frame a_j (i.e. $encoder(a_j)$ = $\mathbf{h} = (\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_{T'})$). Then, for the i^{th} word w_i our model first obtains the mappings of w_i to acoustic frames a_K where K is the set such that $\forall k \in K$

$$k = \arg\max_j (attention(a_j))$$

over all utterances U containing the word w_i in the training set.

Below we describe three different ways of using attention to obtain acoustic word embeddings for a word w_i (here, $n(K)$ represents the cardinality of the set K):

$$w_i = \frac{\sum_{k \in K} encoder(a_k)}{n(K)} \tag{5.1}$$

$$w_i = \frac{\sum_{k \in K} attention(a_k) \cdot encoder(a_k)}{n(K)} \tag{5.2}$$

$$w_i = encoder(a_k) \;\; where \;\; k = \arg\max_{k \in K} attention(a_k) \tag{5.3}$$

Therefore, unweighted Average (U-AVG, Equation 5.1) is just the unweighted combination of all the hidden representations of acoustic frames mapped to a particular word. Attention weighted Average (CAWE-W, Equation 5.2) is the weighted average of the hidden representations of all acoustic frames using the attention weights for a given word. Finally, maximum attention (CAWE-M, Equation 5.3) is the hidden representation of the acoustic frame with the highest attention score for a given word across all utterances in the training data. We call the attention-weighted average and the maximum attention based techniques as Contextual Acoustic Word Embeddings (CAWE) since they are contextual owing to the use of attention scores (over all acoustic frames for a given word).

5.2.3 Experiments and Results

We use a commonly used speech recognition setup, the 300 hour Switchboard corpus (LDC97S62) (Godfrey et al., 1992) which consists of 2,430 two-sided telephonic conversations between 500 different speakers and contains 3 million words of text. Our second dataset is a 300 hour subset of the How2 dataset (Sanabria et al., 2018) of instructional videos, which contains planned, but free speech, often outdoor and recorded with distant microphones, as opposed to the indoor, telephony, conversational speech of Switchboard. There are 13,662 videos with a total of 3.5 million words in this corpus. The A2W obtains a word error rate of 22.2% on Switchboard and 36.6% on CallHome set from the Switchboard Eval2000 test set and 24.3% on dev5 test set of How2.

Comparing Methods for Constructing Embeddings

Datasets for Downstream Tasks We evaluate our embeddings by using them as features for 16 benchmark sentence evaluation tasks that cover Semantic Textual Similarity (STS 2012-2016 and STS B), classification: Movie Review (MR), product review (CJ), sentiment analysis (SST, SST-FG), question type (TREC), Subjectivity/Objectivity (SUBJ), and opinion polarity (MPQA), entailment and semantic relatedness using the SICK dataset for SICK-E (entailment) and SICK-R (relatedness) and paraphrase detection (MRPC). The STS and SICK-R tasks measure Spearman's coefficient of correlation between embedding based similarity and human scores, hence the scores range from $[-1, 1]$ where higher number denotes high correlation. All the remaining tasks are measured on test classification accuracies. We use the SentEval toolkit (Conneau and Kiela, 2018) to evaluate.

Training Details In all downstream evaluations involving classification tasks, we have used a simple logistic regression for classification since a better representation should lead to better scores without using complicated models (hence abstracting away model complexities from our evaluations). This also means that we can use the concatenation of CAWE and CBOW as features to the logistic regression model without adding tunable embedding parameters.

Discussion From the results in Table 5.1 we see that CAWE-M outperforms U-AVG by 34% and 13% and CAWE-W by 33.9% and 12% on Switchboard and How2 datasets respectively in terms of average performance on STS tasks and leads to better or slightly worse performance on the classification tasks. We observe that CAWE-W usually performs worse than CAWE-M which could be attributed to a noisy estimation of the word embeddings on the account of taking even the less confident attention scores while constructing the embedding. In contrast, CAWE-M is constructed using the most confident attention score obtained over all the occurrences of the acoustic frames corresponding to a particular word. We also observe that U-AVG performs worse than CAWE-W on STS and SICK-R tasks since it is constructed using an even

	Switchboard			How2		
Dataset	**U-AVG**	**CAWE-W**	**CAWE-M**	**U-AVG**	**CAWE-W**	**CAWE-M**
STS 2012	0.3230	0.3281	**0.3561**	0.3255	0.3271	**0.3648**
STS 2013	0.1252	0.1344	**0.1969**	0.2070	0.2071	**0.2716**
STS 2014	0.3358	0.3389	**0.3888**	0.3375	0.3426	**0.3940**
STS 2015	0.3854	0.3881	**0.4275**	0.3852	0.3843	**0.4173**
STS 2016	0.2998	0.2974	**0.3833**	0.3248	**0.3271**	0.3159
STS B	0.3667	0.3510	**0.4010**	0.3343	0.3440	**0.4000**
SICK-R	0.5640	0.5800	**0.6006**	0.5800	0.6060	**0.6440**
MR	63.86	63.75	**64.69**	63.46	63.19	**63.64**
MRPC	**70.67**	69.45	69.80	68.29	67.83	**70.61**
CR	71.42	72.13	**72.93**	**74.12**	73.99	73.03
SUBJ	**82.45**	82.22	81.19	81.48	**81.88**	81.01
MPQA	**73.76**	73.28	73.75	**74.21**	74.18	73.53
SST	66.45	**66.61**	65.02	63.43	63.43	**65.13**
SST-FG	32.81	32.04	**33.53**	31.95	**32.35**	32.03
TREC	63.80	62.40	**67.60**	**66.60**	66.00	60.60
SICK-E	**74.20**	73.41	74.06	75.14	75.34	**75.97**

Table 5.1: Comparing three methods to obtain acoustic word embeddings from an A2W model: unweighted average (U-AVG), weighted average (CAWE-W) and maximum attention (CAWE-M).

noisier process in which all encoder hidden representations are weighted equally irrespective of their attention scores.

Comparing with Text-based Embeddings

Datasets for Downstream Tasks The datasets are the same as described above in Section 5.2.3 (Comparing Methods for Constructing Embeddings).

Training Details In all the following comparisons, we compare embeddings obtained only from the training set of the speech recognition model, while the text-based word embeddings are obtained by training Continuous Bag-of-Words (CBOW) word2vec model on all the transcripts (train, validation and test). This was done to ensure a fair comparison between our supervised technique and the unsupervised word2vec method. This naturally leads to a smaller vocabulary for CAWE. Further, one of the drawbacks of A2W speech recognition model is that it fails to capture entire vocabulary, recognizing only 3044 words out of 29874 (out of which 18800 words occur less than 5 times) and 4287 out of 14242 total vocabulary for Switchboard and How2 respectively. Despite this fact, the performance of CAWE is very competitive with word2vec CBOW which does not suffer from reduced vocabulary problem.

	Switchboard			How2		
Dataset	**CAWE-M**	**CBOW**	**Concat**	**CAWE-M**	**CBOW**	**Concat**
STS 2012	0.3561	**0.3639**	0.3470	0.3648	0.3688	**0.3790**
STS 2013	0.1969	0.1960	**0.2010**	**0.2716**	0.2524	0.2675
STS 2014	**0.3888**	0.3745	0.3795	0.3940	**0.3973**	0.3971
STS 2015	0.4275	0.4459	**0.4481**	0.4173	**0.4781**	0.4710
STS 2016	**0.3833**	0.3471	0.3651	0.3159	**0.4023**	0.3388
STS B	0.401	**0.4100**	0.3995	0.4000	**0.4720**	0.4487
SICK-R	0.6006	0.6170	**0.6228**	0.6440	0.6550	**0.6945**
MR	64.69	66.24	**66.89**	63.64	66.03	**66.89**
MRPC	**69.80**	68.99	68.00	**70.61**	69.68	68.52
CR	72.93	74.49	**75.39**	73.03	**74.89**	74.84
SUBJ	81.19	**84.62**	84.59	81.01	84.75	**85.04**
MPQA	73.75	**76.44**	75.36	73.53	75.56	**75.60**
SST	65.02	68.37	**68.97**	65.13	67.66	**68.20**
SST-FG	33.53	34.71	**35.79**	32.08	33.62	**33.67**
TREC	67.60	69.80	**71.40**	60.60	**68.40**	67.40
SICK-E	74.06	75.02	**76.19**	75.97	76.29	**78.14**

Table 5.2: Sentence Evaluations on 16 benchmark datasets for Switchboard and How2 corpus. We compare the CAWE-M method with the word2vec embeddings trained with CBOW method and with CAWE-M + CBOW concatenated (Concat) embeddings.

Discussion In Table 5.2, we see that our embeddings perform as well as the text-embeddings. Evaluations using CAWE-M extracted from Switchboard based training show that the acoustic embeddings when concatenated with the text embeddings outperform the word2vec embeddings on 10 out of 16 tasks. This concatenated embedding shows that we add more information with CAWE-M that improves the CBOW embedding as well. The gains are more prominent in Switchboard as compared to the How2 dataset since How2 is planned instructional speech whereas Switchboard is spontaneous conversational speech (thereby making the How2 characteristics closer to text leading to a stronger CBOW model).

Evaluation on Spoken Language Understanding

Dataset In addition to generic sentence-level evaluations, we also evaluate CAWE on the widely used ATIS dataset (Price, 1990) for Spoken Language Understanding (SLU). ATIS dataset is comprised of spoken language queries for airline reservations that have intent and named entities. Hence, it is similar in domain to Switchboard, making it a useful test bed for evaluating CAWE on a speech-based downstream evaluation task.

Training Details For this task, our model is similar to the simple Recurrent Neural Network (RNN) based model architecture as investigated in (Mesnil et al., 2013). Our architecture is comprised of an embedding layer, a single layer RNN-variant (Simple RNN, Gated Recurrent

	CAWE-M	**CAWE-W**	**CBOW**
RNN	91.49	91.67	91.82
GRU	93.25	93.56	93.11

Table 5.3: Speech-based contextual word embeddings (CAWE-M and CAWE-W) match the performance of the text-based embeddings (CBOW) on the ATIS dataset with an RNN and GRU model

Unit (GRU)) along with a dense layer and softmax. In each instance, we train our model for 10 epochs with RMSProp (learning rate 0.001). We train each model 3 times with different seed values and report average performance.

Discussion (Mesnil et al., 2013) concluded that text-based word embeddings trained on large text corpora consistently lead to better performance on the ATIS dataset. We demonstrate that direct speech-based word embeddings could lead to matching performance when compared to text-based word embeddings in this speech-based downstream task, thus highlighting the utility of our speech based embeddings. Specifically, we compare the test scores obtained by initializing the model with CAWE-M, CAWE-W and CBOW embeddings and fine-tuning them based on the task.

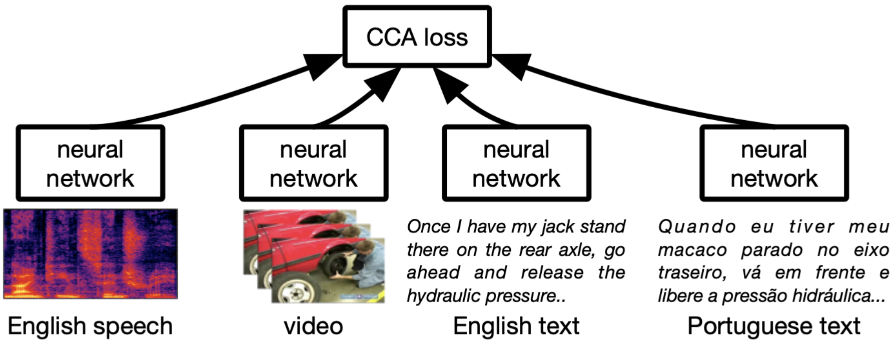

CCA loss
neural network
neural network
neural network
neural network
Once I have my jack stand there on the rear axle, go ahead and release the hydraulic pressure..
Quando eu tiver meu macaco parado no eixo traseiro, vá em frente e libere a pressão hidráulica...
English speech
video
English text
Portuguese text

FIGURE 5.3: Learning from multiple modalities using Canonical Correlation Analysis (CCA) loss.

5.3 Multiview Learning for Multimodal Embeddings

5.3.1 Multiview Learning

Many large datasets include multiple modalities, leading to an exploration of methods that exploit the multimodal structure of the data. Further, collecting large datasets with multiple views is relatively easier than collecting large datasets with high quality annotations. Multiple views help learn better representations for each view separately (Ngiam et al., 2011), or a shared representation across multiple views (Arora and Livescu, 2013), and multiview learning has also been shown to be useful in low-resource settings (Socher and Fei-Fei, 2010). However, the fusion of information from disparate modalities remains a challenging problem (Baltrušaitis et al., 2019).

We build multiview models on the How2 dataset (Sanabria et al., 2018), which at the time was the largest multi-view multimodal dataset. It contains various views of the *same* data point which is necessary for multi-view learning. While previous multiview models have exploited the natural alignment between views, such as speech and articulatory features (Wang et al., 2015), here we have to overcome challenges resulting from latently aligned views. For instance, there exists an alignment between words in an English sentence, and the words in its Portuguese translation. Figure 5.3 shows an overview of our learning algorithm with 4 different input views, described below.

Given these multiple parallel modalities, we address the following: how much information is shared across modalities? How can we learn a representation that captures information from all modalities? We measure this using intrinsic evaluations via retrieval for speech recognition and speech translation downstream tasks.

5.3.2 Model

In the following, we describe the sequence-to-sequence model-based encoders we used to extract features for speech, text and video data. We then describe the correlation-based methods

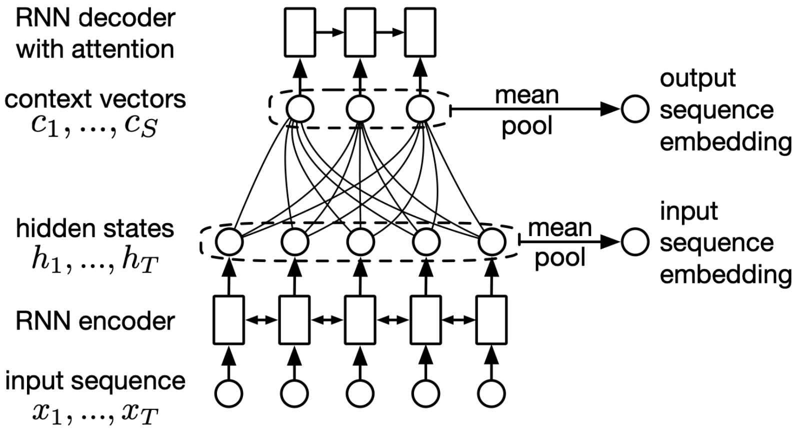

RNN decoder with attention
context vectors $c_1, \ldots, c_S$
mean pool
output sequence embedding
hidden states $h_1, \ldots, h_T$
mean pool
input sequence embedding
RNN encoder
input sequence $x_1, \ldots, x_T$

FIGURE 5.4: Extracting sequence embeddings from trained sequence to sequence models.

we used to learn representations.

Input Representations

We use word-level representations from Machine Translation (MT) and Automatic Speech Recognition (ASR) systems to build utterance-level representations. All the models we consider are attention-based, sequence-to-sequence models (Bahdanau et al., 2014b) that were trained in a supervised way on the How2 dataset . The encoder, a stacked bi-directional RNN, reads a sequence of feature vectors $x_1, \ldots, x_T$ and produces a sequence of hidden states $h_1, \ldots, h_{T'}$ ($T' <= T$ because of possible sub-sampling). The decoder, an RNN with attention mechanism, produces context vectors $c_1, \ldots, c_S$. We use the average of the h_i (resp. c_i) as the representation for the input sequence (resp. output sequence), as depicted in Figure 5.4. The RNNs for ASR are LSTMs (Hochreiter and Schmidhuber, 1997) and GRUs (Cho et al., 2014) for MT. For the acoustic representations to be at the same level of granularity as the word representations from MT, we use the Acoustic2Word model (Palaskar and Metze, 2018) as our ASR model, and obtain acoustic embeddings h_i at word-level.

For the video modality, we condense the information present in each utterance into a single vector as follows. We first use a ResNet (He et al., 2016a) to map each frame of the video to a multi-class posterior, based on the 1000 ImageNet classes. We then compute the average of those posteriors. As the video frames are sampled with a hard temporal threshold, it may contain noisy artifacts. We average over all the frames to capture the most persistent predictions and reduce the variability due to noise. We experimented with representations from action networks (Hara et al., 2018) trained on an action dataset (Kay et al., 2017), and obtained similar results as with ResNet features.

Linear CCA

To measure how much information is shared by pairs of representations, we use Canonical Correlation Analysis (CCA) (Hotelling, 1992). We assume that we are given two views of the same data point: for instance, for a given utterance, the audio recording and the transcription. These two views are represented by random variables X and Y (d_x - and d_y - dimensional respectively), and k is the dimensionality of the shared representation. Linear CCA seeks two linear transformations

$$U \in R^{d_x \times k} \quad \& \quad V \in R^{d_y \times k} \tag{5.4}$$

such that the components of $U^T X$ and $V^T Y$ are maximally correlated. Formally, we want to maximize

$$\mathbb{E}_{X,Y}[\mathrm{tr}(U^T X Y^T V)] \tag{5.5}$$

subject to

$$\mathbb{E}_X[U^T X X^T U] = E_Y[V^T Y Y^T V] = I_k. \tag{5.6}$$

For dataset $\{x_i, y_i\}_{i=1}^N$, we define C_{XY} the empirical cross-covariance matrix between X and Y , and C_{XX} and C_{YY} the empirical auto-covariance matrices of X and Y, respectively. U and V are given by the k left and right singular vectors of $C_{XX}^{-1/2}.C_{XY}.C_{YY}^{-1/2}$ with the largest singular values.

CCA is a better objective than predicting one view with the other when no single regression provides a fully adequate solution. For instance, it is very hard to generate speech from text. Instead, it is easier to predict the dependent variate which has the largest multiple correlation.

Deep CCA

Deep CCA (DCCA) (Andrew et al., 2013) is a natural extension of linear CCA, with the objective to maximally correlate $U^T f(X)$ and $V^T g(Y)$. f and g are non-linear feature extractors, which can be learned via gradient descent on the CCA objective. It is also natural to extend CCA to multiple views (Horst, 1961).

Instead of 2 views, we have J views $X_1, \ldots, X_J$ attached to each data point, stored in matrices $X_j \in R^{d_j \times N}$. Linear transformations $U_j \in R^{d_j \times k}{}_{j=1}^{J}$ are computed, that minimize the mutual reconstruction error under constraints, in a way equivalent to maximizing correlation. This framework can be extended to non-linear feature extractors (Arora and Livescu, 2013) with the objective:
minimize

$$\sum_{j=1}^{J} ||G - U_j^T f_j(X_j)||_2^2 \tag{5.7}$$

subject to

$$GG^T = I_k \tag{5.8}$$

with respect to parameters $\{G, \{f_j, U_j\}_j\}$. Here, $f_j(X_j) \in R^{h_j \times N}$ is the output of j-th feature extractor, and $G \in R^{k \times N}$ can be viewed as the learned representations for the given dataset. Effectively, each pair of feature extractor f_j and linear transformation U_j tries to reconstruct the learned representation G. The constraints on G prevent the feature extractors from collapsing the features. We refer to this method as Deep Generalized CCA (DGCCA). A linear variant of DGCCA where the f_j's are identity maps has been applied to acoustic feature learning (Arora and Livescu, 2013) and learning word embeddings (Rastogi et al., 2015).

5.3.3 Experiments

We applied the methods described above to the How2 dataset and evaluated the learned representations using a retrieval task.

Dataset Setup

We apply the described methods to the How2 dataset (Sanabria et al., 2018), which we use as a 4-way parallel corpus: video, speech, transcription in English, translation in Portuguese. The dataset contains 13,500 videos, or 300 hours of speech, and is split into 185,187 training, 2022 development (dev), and 2361 test utterances. It is yet unclear how much temporal coherence there is between the video modality and the language (text and speech) content, as objects mentioned early in the video may only appear much later on screen, or only for a very brief time.

Whenever we mention an MT task, it consists of translating English (en) text to Portuguese (pt) text; an ASR task consists of transcribing English speech to English text; a Speech Translation (ST) task consists of mapping English speech to Portuguese text.

Evaluations

To measure the richness of the learned representations, we use them in a retrieval task: given a source sequence in view 1, and a set of reference sequences in view 2, find the n sequences in the reference set that are closest to the source sequence. Since we have parallel corpora, we can check whether the correct sequence is present within those n sequences. We report this as Recall@10 (n = 10 throughout here), with scores ranging from 0 to 100. Finding the closest sequences consists of projecting the reference set as well as the source sequence into the shared space, then computing distances between source and references, and retrieving the closest points. The Recall@10 of picking at random from the reference set is 0.5% for the dev set and 0.4% for the test set.

	Linear CCA		Deep CCA		
	dev	**test**	**dev**	**test**	***k***
text (en) - text (pt)	82.5	81.4	95.1	94.6	400
speech - text (en)	98.3	96.9	92.1	90.1	160
video - text (en)	0.9	0.8	2.3	1.6	400
video - speech	0.8	0.6	1.9	1.8	160

TABLE 5.4: Recall@10 for retrieving reference modality given source modality (”source - reference”). Swapping source and reference change retrieval scores by less than 1% absolute.

5.3.4 Results

In the following, we will use k to indicate the dimensionality of the shared representation. Typically, k should be at most the smallest dimensionality of all views involved. We set k to half the smallest dimensionality as a balance between keeping as much information as possible while dropping uninformative components. Throughout all our experiments, we add the identity matrix scaled by 10^{-16} to the view-specific co-variance matrices.

In all experiments involving DCCA and DGCCA, we use 2-layer feedforward neural networks as feature extractors (f, g, f_j), the first layer with the same size as the input, and the second of size k. The training proceeds in epochs, which consist of a full pass over the training set with batch size 5500. After each epoch, we compute the retrieval scores between all possible pairs of different views on the dev set, and aggregate the scores by picking the highest of those scores. Our final model is the one with the highest aggregate score. For the experiments involving the video modality, we used a weight decay of 10^{-5}.

Bimodal Experiments

We start by applying linear CCA and deep CCA to pairs of views, at the utterance level. Text, speech and video sequences are represented with 800-, 320- and 1000-dimensional vectors respectively. As measured by the retrieval rates shown in Table 5.4, the representations learned for text (en and pt) and speech capture almost all of the information present in both views, in a space with half the dimensionality. The original text (en) and text (pt) representations having the same dimensionality, we scored the retrieval of a Portuguese sentence given an English sentence, which yielded a score of 0.38%. For other pairs of modalities, there is no obvious way of computing pairwise distances in the original space. The retrieval scores involving the video modality are very low (discussed below).

To tie the results in Table 5.4 to known metrics, we take the first retrieval result and score it as though it were the output of an ASR or MT system. Given a speech utterance from the test set that we want to transcribe, or a source sentence we wish to translate, we pick the closest sentence from a reference set using our learned DCCA model. We then score this pick using the relevant metric, WER for ASR and BLEU for MT. The score strongly depends on the contents of the reference set: if the reference set contains no appropriate sentence to transcribe (resp.

Reference Set	WER	BLEU (MT)	BLEU (ST)
train	134%	5.2	0.2
train + test	27.4%	80.7	19.8
Baseline S2S	24.3%	57.3	27.9

Table 5.5: Scoring top-1 retrieval result from DGCCA models with ASR, MT and ST metrics. Models used (from left to right) were trained using speech and text (en); text (en) and text (pt); speech, text (en), text (pt) and video. Source sentences for the retrieval are from the test set.

translate) the source, the WER (resp. BLEU) will be high (resp. low). We thus test on two reference sets: 1) the training set, 2) the union of the training and test set. In setting 1, the reference set does not contain the correct answers, whereas it does in setting 2. When using only the test set as a reference set, the score is almost perfect, and we only report the more challenging settings, in columns WER and BLEU (MT) of Table 5.5. The results on the train set are quite poor given that the train set may not contain appropriate sentences for the test set. We estimate this by finding, for each sentence in the test set, the closest sentence (in terms of edit distance) from the train set. This yields a BLEU of 10.6, and a WER of 63.0%. When using the union of test and train as a reference set, our model is still able to mostly pick out the correct sentences, achieving on par or better performance than a baseline sequence-to-sequence model. This is consistent with our retrieval scores, as the retrieval for text and text was slightly higher than speech and text.

n-modal experiments

In subsequent experiments, we used DGCCA to learn representations with more than 2 views. We learned representations with English text, speech and video, with $k = 160$, and report retrieval results in Table 5.6. As compared to Table 5.4, the retrieval scores between speech and text (en) decrease, as the model has to accommodate a third view. Keeping hyperparameters fixed and adding a fourth view, Portuguese text, we obtain the results in Table 5.7. Relative to Table 5.4, the text - text retrieval score increases, while the speech - text (en) score decreases, and the scores involving video decrease slightly. This could be explained by the fact that text - text retrieval is an easier task than those involving speech and video, so that the model trades off a higher loss in the video and speech domain for a lower loss in the text domain. To remedy this, one could add weights w_j to each reconstruction loss, or tune the architectures of the f_j. We again evaluate our speech - text (pt) retrieval with an ST task. The results are shown in column BLEU (ST) of Table 5.5, and are again consistent with the retrieval scores.

Discussion

As shown by our Recall@10 retrieval results, the CCA objective induces a shared space capturing most of the information shared across the original spaces. Scoring the top-1 retrieved data point with common MT, ASR and ST metrics is consistent with this finding. Moreover,

		Text (en)	**Speech**	**Video**
text (en)	dev	-	92.1	1.7
	test	-	89.8	1.8
speech	dev	92.1	-	1.9
	test	89.1	-	1.2
video	dev	1.4	1.9	-
	test	1.7	1.2	-

Table 5.6: Recall@10 for retrieving column modality given source row modality, for a DGCCA model trained on 3 views. Results from the bottom left triangle can be compared to those in Table 5.4.

		Text (pt)	**Text (en)**	**Speech**	**Video**
Text (pt)	dev	-	98.8	73.5	2.1
	test	-	98.3	71.0	1.1
Text (en)	dev	98.8	-	88.2	1.4
	test	98.4	-	85.4	0.9
Speech	dev	73.0	88.1	-	1.1
	test	70.7	85.4	-	1.0
Video	dev	2.1	1.1	1.0	-
	test	1.1	1.1	0.9	-

Table 5.7: Recall@10 for retrieving column modality given source row modality, for a DGCCA model trained on 4 views. Results from the bottom left triangle can be compared to those in Table 5.4.

this shared space is learned on top of high-level, unrelated representations: the training of the ASR and MT systems is entirely independent. Our results involving video are not in agreement with that hypothesis, and we see two possible explanations. First, there is a temporal mismatch between the video modality and the language content. Second, it is possible that the ResNet posteriors are either extremely noisy, or simply fail to identify certain relevant objects because of a domain mismatch. Previous work in the context of ASR shows that using the penultimate instead of the last layer of the ResNet makes little difference (Palaskar et al., 2018).

5.4 Chapter Conclusion

We propose a Contextual Acoustic Word Embedding (CAWE) model that is necessary to align the latent representations across the multiple modalities commonly available through video datasets for the semi-supervised learning approach we use (DGCCA). CAWE does not require word-boundary annotations within a spoken sentence as was necessary for previously developed acoustic word embedding methods. This method performs competitively on 16 standard word embedding evaluation benchmarks and are also useful for other acoustic downstream tasks such as spoken language understanding.

We use these word-level speech representations in an advanced, correlation-based representation learning method, Deep Generalized Canonical Correlation Analysis (DGCCA), which required latent space representation alignment across modalities. This method is a semi-supervised learning method that learns from the multiple multimodal views instead of fully supervised training. We use a real-world noisy dataset (the `How2` dataset) with this method for the first time; prior work had shown the utility of DGCCA either on synthetic datasets or on clean controlled-environment datasets. In addition to expanding this method to a real world dataset, we demonstrate its effectiveness on standard downstream tasks: speech translation, speech recognition, and text translation.

Chapter 6

Summarization & QA

6.1 Introduction

With Chapter 4 and Chapter 5, we focused on monotonic and re-ordering tasks that fuse multimodal information directly at the input or at a higher-level latent representation for Speech Recognition and Translation. Both of these tasks are conventionally modeled at a sentence-level with each sentence of a video considered an independent data point. While there could be necessary context across a video that improves these tasks with the entire video context instead of a single sentence, the memory and compute requirement make such modeling infeasible currently.

While sentence-level modeling is a solution to Speech Recognition and Translation, there are certain *abstraction* tasks such as Video Summarization or Question Answering that do not enjoy the same flexibility. For such abstraction tasks, entire video context is necessary to perform the information compression, re-phrasing, and re-structuring that is necessary to generate a natural language video summary or answer. In this Chapter, we extend the multimodal learning tasks to Video Summarization and Question Answering. We perform these tasks independently, as well as in conjunction using a transfer learning approach.

For video-level modeling and abstraction, there is a necessity for the multimodal fusion model to adapt to these task requirements as well. In this Chapter, we introduce the *Hierarchical Latent Representation Fusion* model that provides the necessary control for video-level abstraction. Figure 6.1 contrasts the Hierarchical Latent Representation Fusion model with the other fusion models used so far. Conceptually, the "hierarchical" design of this model allows for this video-level control to generate abstractive text accessing information from any of the input modalities: video or text transcript. Further discussion on this model is below.

Video summarization is a compression, rephrasing and restructuring task. Mulitmodal video summarization further combines information obtained from the visual modality. With these two input modalities, we address this by aiming to generate a short text summary of the video that describes the most salient content of the video. A video summary can aid the search and retrieval of relevant videos without watching. it can also help in representing content of a video where other searchable metadata is unavailable (Song et al., 2011; Wang et al., 2012; Otani et al., 2016; Torabi et al., 2016). Video summaries benefit users through better contextual information and user experience, and helps video sharing platforms with increased user engagement and retention by retrieving or suggesting relevant videos.

We study multimodal summarization with various methods to summarize the intent of *open-domain instructional videos* stating the exclusive and unique features of the video, irrespective of modality. We study this task in detail using the How2 dataset (Sanabria et al., 2018) which contains human annotated video summaries for a varied range of topics. Our models generate natural language descriptions for video content using the transcriptions (both user-generated and output of automatic speech recognition systems) as well as visual features extracted from the video. We also introduce a new evaluation metric (Content F1) that suits this task and present detailed results to understand the task better.

In addition to summarization, we explore the video question answering task that generates natural language answers based on audio, video, textual questions, and textual video-summaries. In this task, we extend beyond the summarization task that involves compression of *all* input information into a shortened form, to now generating text based on specific questions by selecting appropriate portions of the inputs. In addition to compression and rephrasing required for summarization, question answering further requires information selection. For question answering, we design a transfer learning model that leverages the summarization model properties and adapts them to answer generation. We participated in the 7^{th} Dialog State Tracking Challenge (2019) with this transfer learning model. Our submissions ranked first in the automatic as well as human evaluation metrics of this challenge (Yoshino et al., 2018).

Hierarchical Latent Representation Fusion Model

The Input Fusion model provides a technique for monotonic adaptation of the various input modalities, here X_{M1} and X_{M2}, and is specifically designed for modalities that have a high granularity of monotonic constraint, for example, speech and transcription. The Latent Representation Fusion model provides a higher-level adaptation in the latent space, LY_1. This higher-level representation relaxes the strict monotonicity constraint allowing re-ordering tasks like translation.

For video summarization and question answering, there is no such monotonic constraint in the task. Instead, the is an abstraction constraint wherein the information across the entire video (not utterances) needs to be compressed and restructured to form natural language summaries or answers. This is often an open-vocabulary text generation problem. For such an abstraction constraint, we propose to extend the Latent Representation Fusion model to a *Hierarchical Latent Representation Fusion* model that can fuse input modalities, perform abstraction, as well as generate observable outputs Y_1. In our formulation, Y_1 is the output of a text generator.

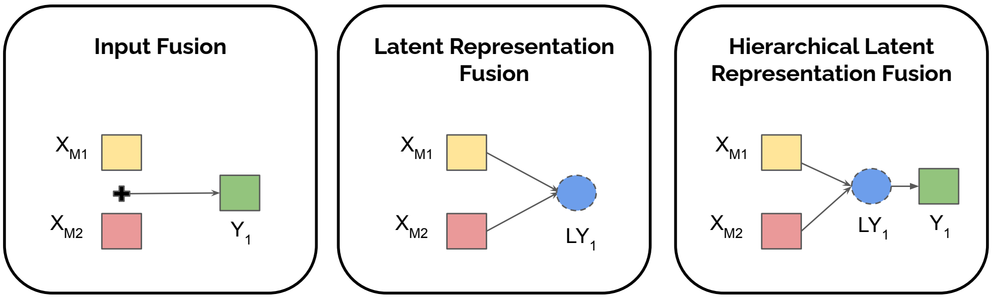


FIGURE 6.1: We build on top of the existing Input Fusion and Latent Representation Fusion models presented so far for Summarization and Question Answering. We present the Hierarchical Latent Representation Fusion model that not only learns a multimodal latent representation LY_1, but converts it into observable outputs Y_1 via hierarchical combination.

The output generator achieves explicit control over the input time-scales for both modalities X_{M1} and X_{M2} through LY_1. More specifically, to implement joint input fusion and abstraction, we use the Hierarchical Attention mechanism proposed by Libovický and Helcl within Latent Fusion (LY_1) step, and this modified LY_1 is used for text generation Y_1. The Hierarchical Attention mechanism was initially proposed for Machine Translation. In this work, we explore its applications for video-level abstraction tasks, Summarization and Question Answering. While the Hierarchical Latent Representation Fusion model can be used for Speech Recognition and Translation, the converse is not possible. For abstraction tasks, the hierarchical control over input modality time-scales provided by this model is essential. Through the experiments in this chapter, we show the need for such model evolution across learning tasks.

Chapter Structure

In this Chapter, we explore the two abstraction tasks in sequence, and in conjunction. We begin by describing the video summarization task formulation, followed by models used and the corresponding automatic and human evaluation results. The next Section describes the video question answering task, transfer learning from summarization to question answering to connect the two abstraction tasks, followed by the transfer learning results and the system description for our participation in the Dialog State Tracking Challenge with this transfer learning approach. The work presented in this chapter has been published in Palaskar et al. 2019a, Sanabria et al. 2019 and Palaskar et al. 2020.

6.2 Summarization

6.2.1 Task Formulation

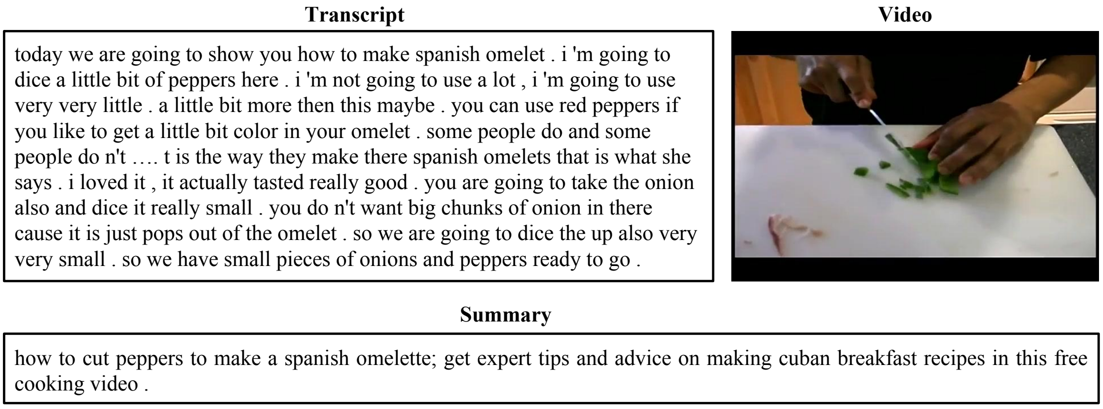
Summarization is a task of producing a shorter version of the content in the document while preserving its information and has been studied for both textual documents (automatic text summarization) and visual documents such as images and videos (video summarization). Automatic text summarization is a widely studied topic in natural language processing (Luhn, 1958; Kupiec et al., 1995; Mani, 1999); given a text document the task is to generate a textual summary for applications that can assist users to understand large documents. Most of the work on text summarization has focused on single-document summarization for domains such as news (Rush et al., 2015; Nallapati et al., 2016; See et al., 2017; Narayan et al., 2018) and some on multi-document summarization (Goldstein et al., 2000; Lin and Hovy, 2002; Woodsend and Lapata, 2012; Cao et al., 2015; Yasunaga et al., 2017).

Video summarization is the task of producing a compact version of the video (visual summary) by encapsulating the most informative parts (Money and Agius, 2008; Lu and Grauman, 2013; Gygli et al., 2014; Song et al., 2015; Sah et al., 2017). Multimodal summarization is the combination of textual and visual modalities by summarizing a video document with a text summary that summarizes the content of the video. Multimodal summarization is a more recent challenge with no benchmarking datasets yet. Li et al. (2017) collected a multimodal corpus of 500 English news videos and articles paired with manually annotated summaries. The dataset is small-scale and has news articles with audio, video, and text summaries, but there are no human annotated audio-transcripts.

Related tasks include image or video captioning and description generation, video story generation, procedure learning from instructional videos and title generation which focus on events or activities in the video and generating descriptions at various levels of granularity from single sentence to multiple sentences (Das et al., 2013; Regneri et al., 2013; Rohrbach et al., 2014; Zeng et al., 2016; Zhou et al., 2018a; Zhang et al., 2018; Gella et al., 2018). A closely related task to ours is video title generation where the task is to describe the most salient event in the video in a compact title that is aimed at capturing users attention (Zeng et al., 2016). Zhou et al. (2018a) present the YouCook II dataset containing instructional videos, specifically cooking recipes, with temporally localized annotations for the procedure which could be viewed as a summarization task as well although localized with time alignments between video segments and procedures.

6.2.2 Models

Text-based We study various summarization models. First, we use a Recurrent Neural Network (RNN) Sequence-to-Sequence (S2S) model (Sutskever et al., 2014b) consisting of an encoder RNN to encode (text or video features) with the attention mechanism (Bahdanau et al.,



Transcript

today we are going to show you how to make spanish omelet . i 'm going to dice a little bit of peppers here . i 'm not going to use a lot , i 'm going to use very very little . a little bit more then this maybe . you can use red peppers if you like to get a little bit color in your omelet . some people do and some people do n't …. t is the way they make there spanish omelets that is what she says . i loved it , it actually tasted really good . you are going to take the onion also and dice it really small . you do n't want big chunks of onion in there cause it is just pops out of the omelet . so we are going to dice the up also very very small . so we have small pieces of onions and peppers ready to go .

Video

Summary

how to cut peppers to make a spanish omelette; get expert tips and advice on making cuban breakfast recipes in this free cooking video .

Figure 6.2: How2 dataset example with different modalities. “Cuban breakfast” and “free cooking video” is not mentioned in the transcript, and has to be derived from other sources.

2014b) and a decoder RNN to generate summaries. Our second model is a Pointer-Generator (PG) model (Vinyals et al., 2015a; Gülçehre et al., 2016) that has shown strong performance for abstractive summarization (Nallapati et al., 2016; See et al., 2017).

Video-based We represent videos by features extracted from a pre-trained action recognition model: a ResNeXt-101 3D Convolutional Neural Network (Hara et al., 2018) trained to recognize 400 different human actions in the Kinetics dataset (Kay et al., 2017). These features are 2048 dimensional, extracted for every 16 non-overlapping frames in the video. This results in a sequence of feature vectors per video rather than a single/global one. A single-layer RNN encoder is used to represent these sequence-based video features.

Text-and-Video-based As our third model, we use hierarchical attention approach of Libovický and Helcl 2017 originally proposed for multimodal machine translation to combine textual and visual modalities to generate text. The model first computes the context vector independently for each of the input modalities (text and video). In the next step, the context vectors are treated as states of another encoder, and a new vector is computed. In Figure 6.3 we present the building block of our models.

More specifically, the computation of the hierarchical attention model is divided in two steps as described in Libovický and Helcl 2017. Firstly, independent context computation, using the standard attention mechanism proposed by Bahdanau et al. (Bahdanau et al., 2014b). Equations 1-3 describe the standard attention mechanism that we will use further to build the hierarchical attention mechanism. e_{ij} is the representation of the attention energies, α_{ij} is the attention distribution, and c_i is the context vector in the i^{th} decoder step.

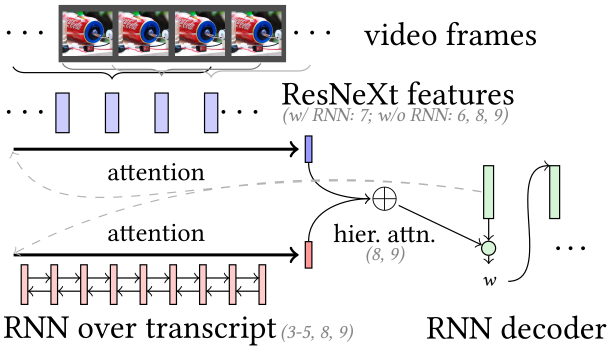


FIGURE 6.3: Building blocks of the sequence-to-sequence models, gray numbers in brackets indicate which components are utilized in which experiments.

$$e_{ij} = v_a^T tanh(W_a s_i + U_a h_j) \tag{6.1}$$

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{T_x} exp(e_{ik})} \tag{6.2}$$

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j \tag{6.3}$$

Equation 5.3 computes the context vector for each encoder state independently. The second step of hierarchical attention computation is as follows:

$$e_i^{(k)} = v_b^T tanh(W_b s_i + U_b^{(k)} c_i^{(k))} \tag{6.4}$$

$$\beta_i^{(k)} = \frac{exp(e_i^{(k)}}{\sum_{n=1}^{N} exp(e_i^{(n)})} \tag{6.5}$$

$$c_i = \sum_{k=1}^{N} \beta_i^{(k)} U_c^{(k)} c_i^{(k)} \tag{6.6}$$

The context vectors obtained via Equation 5.3 is projected onto a shared space (Equation 5.4) to compute another attention distribution over the projected context vectors (Equation 5.5), and their corresponding weighted average (Equation 5.6). $c_i^{(k)}$ is the context vector of the k-th encoder, additional trainable parameters v_b and W_b are shared for all encoders. $U_b^{(k)}$ and $U_c^{(k)}$ are encoder-specific projection matrices, that can be set equal and shared.

Set	Words
Transcript	the, to, and, you, a, it, that, of, is, i, going, we, in, your, this, 's, so, on
Summary	in, a, this, to, free, the, video, and, learn, from, on, with, how, tips, for, of, expert, an

Table 6.1: Most frequently occurring words in Transcript and Summaries.

6.2.3 Experimental Setup

Dataset The How2 dataset (Sanabria et al., 2018) contains about 2,000 hours of short instructional videos, spanning different domains such as cooking, sports, indoor/outdoor activities, music, etc. Each video is accompanied by a human-generated transcript and a 2 to 3 sentence summary is available for every video written to generate interest in a potential viewer. The example in Figure 6.2 shows the transcript describes instructions in detail, while the summary is a high-level overview of the entire video, mentioning that the peppers are being "cut", and that this is a "Cuban breakfast recipe", which is not mentioned in the transcript. We observe that text and vision modalities both contain complementary information, thereby when fused, helps in generating richer and more fluent summaries. Additionally, we can also leverage the speech modality by using the output of a speech recognizer as input to a summarization model instead of a human-annotated transcript. The How2 corpus contains 73,993 videos for training, 2,965 for validation and 2,156 for testing. The average length of transcripts is 291 words and of summaries is 33 words. A more general comparison of the How2 dataset for summarization as compared with certain common datasets is given in (Sanabria et al., 2018).

Table 6.1 shows the frequent words in transcripts (input) and summaries (output). The words in transcripts reflect conversational and spontaneous speech while words in the summary reflect their descriptive nature.

Evaluation We evaluate the summaries using the standard metric for abstractive summarization ROUGE-L (Lin and Och, 2004) that measures the longest common sequence between the reference and the generated summary. Additionally, we introduce the Content F1 metric that fits the template-like structure of the summaries. We analyze the most frequently occurring words in the transcription and summary. The words in transcript reflect the conversational and spontaneous speech while the words in the summaries reflect their descriptive nature. For examples, see Table 6.1.

The Content F1 metric is the F1 score of the content words in the summaries based over a monolingual alignment, similar to metrics used to evaluate quality of monolingual alignment (Sultan et al., 2014). We use the METEOR toolkit (Banerjee and Lavie, 2005; Denkowski and Lavie, 2014) to obtain the alignment. Then, we remove function words and task-specific stop words that appear in most of the summaries (see Table 6.1) from the reference and the hypothesis. The stop words are easy to predict and thus increase the ROUGE score. We treat remaining

No.	Description	ROUGE-L	Content F1
1	Random Baseline using Language Model	27.5	8.3
2a	Rule-based Extractive summary	16.4	18.8
2b	Next-neighbor Summary	31.8	17.9
3	Using Extracted Sentence from 2a only (Text-only)	46.4	36.0
4	First 200 tokens (Text-only)	40.3	27.5
5a	S2S Complete Transcript (Text-only, 650 tokens)	**53.9**	**47.4**
5b	PG Complete Transcript (Text-only)	50.2	42.0
5c	ASR output Complete Transcript (Text-only)	46.1	34.7
6	Action Features only (Video)	38.5	24.8
7	Action Features + RNN (Video)	**46.3**	**34.9**
8	Ground-truth transcript + Action with Hierarchical Attn	**54.9**	**48.9**

Table 6.2: ROUGE-L and Content F1 for different summarization models: random baseline (1), rule-based extracted summary (2a), nearest neighbor summary (2b), different text-only (3,4,5a), pointer-generator (5b), ASR output transcript (5c), video-only (6-7) and text-and-video model (8).

content words from the reference and the hypothesis as two bags of words and compute the F1 score over the alignment. Note that the score ignores the fluency of output.

In addition to automatic evaluation, we perform a human evaluation to understand the outputs of this task better. Following the abstractive summarization human annotation work of Grusky et al. (2018), we ask our annotators to label the generated output on a scale of $1-5$ on informativeness, relevance, coherence, and fluency. We perform this on randomly sampled 500 videos from the test set. We evaluate three models: two unimodal (text-only (5a), video-only (7)) and one multimodal (text-and-video (8)). Three workers annotated each video on Amazon Mechanical Turk. They compared outputs of unimodal and multimodal models with the ground-truth summary and assign a score between 1 (lowest) and 5 (highest). The annotators were shown the ground-truth summary and a candidate summary (without knowledge of the type of modality used to generate it). Annotation was restricted to English speaking countries. In total, 129 annotators participated in this task.

6.2.4 Results

As a baseline, we train an RNN language model (Sutskever et al., 2011) on all the summaries and randomly sample tokens from it. The output obtained is fluent in English leading to a high ROUGE score, but the content is unrelated which leads to a low Content F1 score in Table 6.2. As another baseline, we replace the target summary with a rule-based extracted summary from the transcription itself. We used the sentence containing words "how to" with predicates *learn*, *tell*, *show*, *discuss* or *explain*, usually the second sentence in the transcript. Our final baseline was a model trained with the summary of the nearest neighbor of each video in the Latent Dirichlet Allocation (LDA; Blei et al., 2003a) based topic space as a target. This model achieves

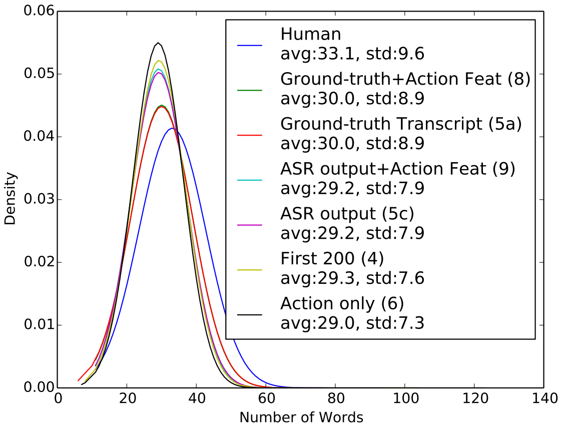


FIGURE 6.4: Word distribution in comparison with the human summaries for different unimodal and multimodal models. Density curves show the length distributions of human annotated and system produced summaries.

a similar Content F1 score as the rule-based model which shows the similarity of content and further demonstrates the utility of the Content F1 score.

We use the transcript (either ground-truth transcript or speech recognition output) and the video action features to train various models with different combinations of modalities. The text-only model performs best when using the complete transcript in the input (650 tokens). This is in contrast to prior work with news-domain summarization (Nallapati et al., 2016). We also observe that PG networks do not perform better than S2S models on this data which could be attributed to the abstractive nature of our summaries and also the lack of common n-gram overlap between input and output which is the important feature of PG networks. We also use the automatic transcriptions obtained from a pre-trained automatic speech recognizer as input to the summarization model. This model achieves competitive performance with the video-only models (described below) but degrades noticeably than ground-truth transcription summarization model. This is as expected due to the large margin of ASR errors in distant-microphone open-domain speech recognition.

We trained two video-only models: the first one uses a single mean-pooled feature vector representation for the entire video, while the second one applies a single layer RNN over the vectors in time. Note that using only the action features in input reaches almost competitive ROUGE and Content F1 scores compared to the text-only model showing the importance of both modalities in this task. Finally, the hierarchical attention model that combines both modalities obtains the highest score.

Human Evaluation In Table 6.3, we report human evaluation scores on our best text-only, video-only and multimodal models. In three evaluation measures, the multimodal models with the hierarchical attention reach the best scores.

In Figure 6.4, we analyze the word distributions of different system generated summaries with the human annotated reference. The density curves show that most model outputs are shorter

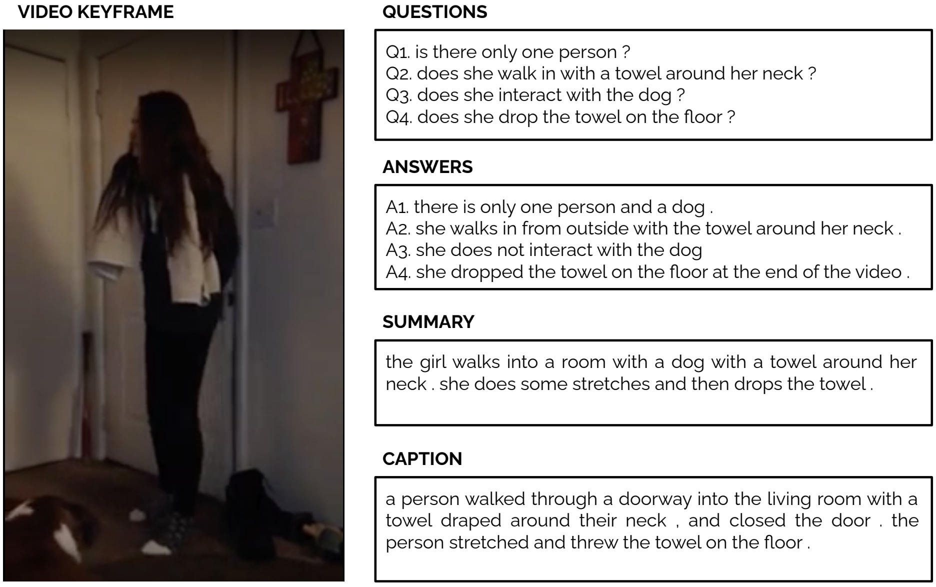
Model (No.)	INF	REL	COH	FLU
Text-only (5a)	3.86	**3.78**	3.78	3.92
Video-only (7)	3.58	3.30	3.71	3.80
Text-and-Video (8)	**3.89**	3.74	**3.85**	**3.94**

Table 6.3: Human evaluation scores on 4 different measures of Informativeness (INF), Relevance (REL), Coherence (COH), Fluency (FLU).

No.	Model	R-L	C-F1	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 .
8	Ground-truth text + Action Feat.	54.9	48.9	learn from our expert how to attach thread to fly fishing for fly fishing in this free how - to video on fly tying tips and techniques .
5a	Text-only (Ground-truth)	53.9	47.4	learn from our expert how to tie a thread for fly fishing in this free how - to video on fly tying tips and techniques .
5c	ASR output	46.1	34.7	learn tips and techniques for fly fishing in this free fishing video on techniques for and making fly fishing nymphs .
7	Action Features + RNN	46.3	34.9	learn about the equipment needed for fly tying , as well as other fly fishing tips from our expert in this free how - to video on fly tying tips and techniques .
6	Action Features only	38.5	24.8	learn from our expert how to do a double half hitch knot in this free video clip about how to use fly fishing .
2b	Next Neighbor	31.8	17.9	use a sheep shank knot to shorten a long piece of rope . learn how to tie sheep shank knots for shortening rope in this free knot tying video from an eagle scout .
1	Random Baseline	27.5	8.3	learn tips on how to play the bass drum beat variation on the guitar in this free video clip on music theory and guitar lesson .

Table 6.4: Example outputs of ground-truth text-and-video with hierarchical attention (8), text-only with ground-truth (5a), text-only with ASR output (5c), action features with RNN (7) and action features only (6) models compared with the reference, the topic-based next neighbor (2b) and random baseline (1). Arranged in the order of best to worst summary in this table.

than human annotations with the action-only model (6) being the shortest as expected. Interestingly, the two different uni-modal and multimodal systems with ground-truth text and ASR output text features are very similar in length showing that the improvements in Rouge-L and Content-F1 scores stem from the difference in content rather than length. Example presented in Table 6.4 shows how the outputs vary.



Q1. is there only one person ?
Q2. does she walk in with a towel around her neck ?
Q3. does she interact with the dog ?
Q4. does she drop the towel on the floor ?

A1. there is only one person and a dog .
A2. she walks in from outside with the towel around her neck .
A3. she does not interact with the dog
A4. she dropped the towel on the floor at the end of the video .

the girl walks into a room with a dog with a towel around her neck . she does some stretches and then drops the towel .

a person walked through a doorway into the living room with a towel draped around their neck , and closed the door . the person stretched and threw the towel on the floor .

Figure 6.5: An example from the Charades dataset. For every video, there exists a video-dialog of 10 questions and answers each. The dataset additionally has the audio, summary and caption for each video.

6.3 Question Answering

6.3.1 Task Description

Inspired by the popular American word guessing game, Charades, where one player acts out a phrase and the other players guess what phrase it is, Sigurdsson et al. 2016 collect the *Charades* dataset of common human household activities. They ask annotators on Amazon Mechanical Turk to record their activities at home using their personal equipment, and upload the recorded video to their platform. They follow a set procedure for these recordings called the *Hollywood in Homes* approach: (1) script generation, (2) video direction and acting based on the scripts, and (3) video verification. At the time, this was the largest dataset of common household activities recorded *at home* and in an uncontrolled environment. They collect 10,000 videos in total, averaging 30 seconds each. There are 157 unique actions across the dataset. An example video from this dataset can be viewed at `https://youtu.be/x9AhZLDkbyc`.

Alamri et al. in 2019 extend this dataset to a Video Question Answering task, the Audio-Visual Scene-Aware Dialog (AVSD) task. They take the videos collected in the Charades dataset and collect 10 question-answer dialog sets for each, in addition to a video summary and a caption. The summary is a compressed version of the actions/activities in the video, whereas the caption is a more descriptive text of video content. This was one of the first video dialog datasets that contained the audio, video, summary, question, answer, and dialog modalities. The goal of the AVSD task is to automatically answer questions about a visual stream (*i.e.*, videos or images) and maintain dialog context across the various questions.

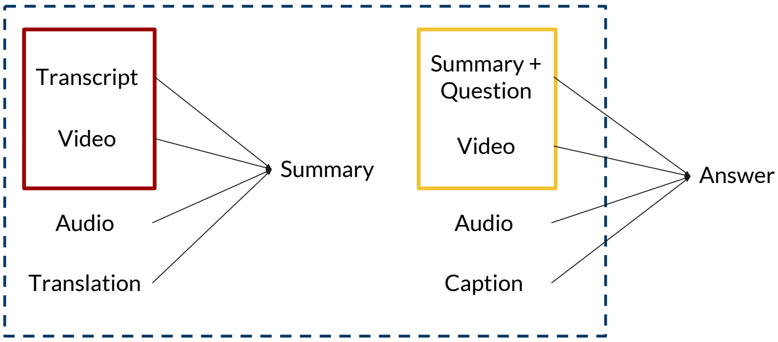


FIGURE 6.6: Our best performing model use the weights of a trained summarization model on the How2 dataset (left) to initialize the training of our DTSC7 challenge model (right).

6.3.2 Transfer Learning: Summarization to QA

There are many common modalities between the Charades dataset and the How2 dataset. To exploit this fact and increase the training data for this task, we first train models on the How2 data and then fine-tune (FT) them on the Charades dataset. The transfer learning setup and respective input modalities are shown in Figure 6.6. The models trained on How2 data use transcription of video (and/or video features) in the input and generate an abstractive textual summary of the video in the output. The methods used for training these are described in (Libovický et al., 2018). We initialize the training of a sequence-to-sequence model for the Charades data with the weights of this learned model, using summary+question (and/or video features) in the input and generating the answer in the output. While fine-tuning, we share the vocabulary for the two datasets and randomly initialize words that do not occur in both.

Although the two datasets have the same modalities, there are differences in the outputs. The main difference between the two datasets is that the summaries of the How2 dataset (usually 2-3 sentences) follow a particular pattern or template as described in (Sanabria et al., 2018), while the pattern of answers (usually single sentence) in the AVSD dataset do not follow any specific template, and are linguistically free-flowing[1]. Also, the videos in the How2 dataset are much longer and semantically-task specific (instructional) than the Charades dataset where they are about day-to-day activities, making the visual modality a semantically richer modality for How2.

We participated in the 7^{th} Dialog State Tracking Challenge (2019) which used this dataset and applied our video summarization models described in Section 6.2.2 via Transfer Learning to Dialog-based Video Question Answering. Our submissions ranked first in the automatic as well as human evaluation metrics of this challenge.

[1] The AVSD data collection procedure tried to avoid "Yes"-"No" answer biases and short answer biases by collecting more descriptive answers (Alamri et al., 2018)

6.3.3 Experimental Setup

Dataset The DSTC7 organizers crowdsourced human annotated questions, answers, captions, and summaries from videos belonging to the Charades dataset (Sigurdsson et al., 2016) to curate the Audio-Visual Scene Aware Dialog dataset (Alamri et al., 2017, 2018; Sigurdsson et al., 2016). The original videos of this dataset contain untrimmed and multi-action videos. In the DSTC7 dataset, each video has ten questions and answers pairs. The dataset statistics for training, validation, disclosed test and undisclosed evaluation test set are given in Table 6.5.

	Charades		**How2**
Split	**Sentences**	**Videos**	**Videos**
train	76590	7659	73993
val	17870	1787	2965
test	7330	733	2156
held_out	6745	1710	169*

Table 6.5: Dataset statistics for Charades and How2. The number of videos in the held_out test set of How2 is from the 300 hours subset of the data (*).

Multimodal Features To fully exploit the information provided in the videos we extract different representations from each modality. To do so, we use DNNs trained for a particular task to extract their internal representation. Based on empirical observations, we know that pre-trained DNNs capture specific characteristics to solve an specific task. Therefore we use DNNs trained for object recognition, place recognition, action recognition and audio even detection to extract meaningful representation of the data. We hypothesize that each of this features will capture information of the video that will be useful to answer each question.

Object Features: These feature are an intermediate representation of a CNN ResNet-50 trained with the ImageNet datset (Deng et al., 2009a)[2]. ImageNet is a dataset for object recognition with more than one milion of images annotated with 1000 classes.

Place Features: (Nallapati et al., 2016) extract scene feature representations from a static image. In this case, the network is trained to recognize scenes from an image. More specifically, (Nallapati et al., 2016) trained the network with Places dataset (Zhou et al., 2017) that contains 10 million images comprising with more than 400 classes.

I3D Flow: (Carreira and Zisserman, 2017) are video features extracted from an spatiotemporal CNN architecture trained for action recognition. The network is trained to recognize 400 different human actions. (Carreira and Zisserman, 2017) use a optical flow representation of the Kinetics Human Action Video dataset that contains 400 samples for class. We extract a 2048 dimensional representation from the `Mixed_5c` layer.

I3D RGB: are also video features from (Carreira and Zisserman, 2017) but instead of using optical flow, the network use video frames with three channels as input stream.

[2] https://github.com/KaimingHe/deep-residual-networks

3D ResNeXt: (Hara et al., 2018) is a 3 dimensional version of the traditional ResNet-101. The third dimensionality of the convolution allows us to extract features from the video instead from an image. The network, similar to VGGish and I3D Flow, is trained with the Kinetics Human Action Video dataset. From 3D ResNeXt, we extract a 2048 dimensional vector.

VGGish: (Hershey et al., 2017) are audio features that have been extracted from a CNN to perform audio even detection network. The network architecture is inspired by the traditional image classification network: VGG. This network works with log Mel spectrograms features extracted from 16 KHz audio recordings. The network was trained with 70M training videos (5.24 million hours) with a total of 30,871 target labels. We use a 128-dimensional embedding.

Evaluation We report the common natural language processing metrics like BLEU (Papineni et al., 2002), METEOR (Denkowski and Lavie, 2014), ROUGE-L (Lin and Och, 2004), and CIDEr (Vedantam et al., 2015). In addition to these objective evaluation metrics for this task, the organizers also evaluated some models on crowdsourced human scores. The human evaluators were asked to score model outputs based on how semantically, grammatically and factually correct the generated answers are.

Models We apply the same text-, video-, and text-and-video multimodal models as for summarization to this task. For the added audio modality via VGGish features, we apply the same hierarchical attention model. This is to evaluate the robustness of the hierarchical attention model on other abstractive tasks as well as to evaluate the fusion mechanism on a vast array of multimodal features. We additionally evaluate the benefits of transfer learning on datasets with shared modalities.

6.3.4 Results

Table 6.6 presents our different models trained using Charades and How2 data, and using various modalities one at a time (text-only, video-only) or together (text-and-video). First, we report the baseline results using the model architecture and code-base provided by the competition organizers (Alamri et al., 2017). We replicate their results using I3D RGB, I3D Flow and VGGish features. To compare the performance of different visual features, we use Objects, Places and 3D ResNeXt in the baseline and observe similar or slightly worse performance showing that all features are equally rich representations.

For text-only models (model #7 and #8 in Tables 6.6, 6.7), the input is a concatenation of the summary of the video followed by the question. The summary is repeated for every question following the assumption that it has relevant input information for each question. Further, we will see that improvements in the text-and-video models over text-only models show that only using the summary in the input may not be sufficient and visual features are useful in such scenarios. In the video-only models (model #9 and #10), we observe lower performance than

No.	Description	BL-1	BL-2	BL-3	BL-4	MET	R-L	C
	Input: Text and Video (different features), Model: Baseline (Alamri et al., 2017)							
1	Charades & I3D RGB & I3D Flow	0.273	0.173	0.118	0.084	0.117	0.291	0.766
2	Charades & I3D RGB & I3D Flow & VGGish	0.271	0.172	0.118	**0.085**	0.116	**0.292**	**0.791**
3	Charades & Objects	0.272	0.173	0.117	0.083	**0.118**	0.287	0.742
4	Charades & Places	0.269	0.171	0.116	0.082	0.116	0.286	0.727
5	Charades & 3D ResNeXt	0.264	0.166	0.112	0.079	0.116	0.284	0.711
6	Charades & 3D ResNeXt & Objects & Places	**0.276**	**0.176**	**0.120**	**0.085**	**0.118**	0.287	0.752
	Input: Text Only, Model: S2S							
7	Charades	0.297	0.200	0.142	0.105	0.138	0.330	1.079
8	How2 FT Charades	**0.311**	**0.212**	**0.152**	**0.114**	**0.146**	**0.337**	**1.169**
	Input: Video Only (3D ResNeXt features), Model: Video-RNN							
9	Charades	0.264	0.170	0.118	0.085	0.116	0.294	0.804
10	How2 FT Charades	**0.279**	**0.179**	**0.122**	**0.086**	**0.122**	**0.300**	**0.833**
	Input: Text and Video (different features), Model: Hierarchical Attention							
11	Charades & Objects	0.274	0.179	0.125	0.091	0.121	0.301	0.876
12	Charades & Places	0.287	0.191	0.136	0.101	0.133	0.320	1.036
13	Charades & VGGish	0.303	0.206	0.148	0.110	0.144	0.338	1.150
14	Charades & 3D ResNeXt	**0.306**	**0.209**	**0.150**	**0.112**	**0.144**	**0.338**	**1.161**
15	How2 FT Charades & 3D ResNeXt	**0.307**	**0.210**	**0.151**	**0.113**	**0.145**	**0.339**	**1.180**

Table 6.6: Automatic evaluation metrics on the test set provided by the organizers (groundtruth available). Models 1-6 are trained using the methods described in (Alamri et al., 2017) with different modalities. We treat them as our baselines. Models 7 and 8 are trained on text-only, models 9 an 10 on video-only and models 11-15 on text-and-video. Models 8, 10 and 15 are first trained on the How2 data and then fine-tuned FT on the Charades data.

the text-only model as expected. It is interesting to note that the video-only model is worse only by about 3-4 ROUGE-L points than the text-only model showing the richness of the visual features (3D ResNeXt). In text-and-video models (model #11-15), we use Hierarchical attention for multimodal adaptation with different visual features. We observe that 3D ResNeXt performs the best for adaptation models.

We fine-tune each of these models on the summarization models trained using the How2 data. In the text-only (model #8) and video-only (model #10), we see substantial gains using fine-tuning over models trained only on Charades data. For the text-and-video model, the gains are not too high, and further exploration of this behavior is necessary to understand why.

Table 6.7 shows the 4 best models from Table 6.6 which we submitted to the challenge. These were evaluated on the undisclosed evaluation test set by the challenge organizers. The baselines (model #1 and #2) are same as those in the previous table but evaluated on the undisclosed test

No.	Description	BL-1	BL-4	MET	R-L	C	Human
	Input: Text and Video (different features), Model: Baseline (Alamri et al., 2017)						
1	Charades & I3D RGB & I3D Flow	0.621	0.305	0.217	0.481	0.733	-
2	Charades & I3D RGB & I3D Flow & VGGish	0.626	0.309	0.215	0.487	0.746	2.848
	Input: Text Only, Model: S2S						
7	Charades *	0.692	0.364	0.254	0.543	1.006	-
8	How2 FT Charades *	**0.711**	**0.376**	**0.264**	**0.554**	**1.076**	3.394
	Input: Text and Video (3D ResNeXt features), Model:Hierarchical Attention						
9	Charades *	0.718	**0.394**	**0.267**	**0.563**	**1.094**	**3.491**
15	How2 FT Charades *	**0.723**	**0.387**	**0.266**	**0.564**	1.087	3.459
-	Groundtruth	-	-	-	-	-	3.938

Table 6.7: **Automatic and Human evaluation** scores on the undisclosed evaluation test set prepared by DTSC7 organizers (we do not have access to groundtruth). Models 1 and 2 are the same baselines as in Table 6.6. Models 3 and 4 are trained on text-only. Models 5 and 6 are trained on text-and-video using Hierarchical attention. Models 4 and 6 are first trained on the How2 data and then fine-tuned FT on the Charades data. Systems marks with an asterisk (*) were the ones submitted to the challenge. Model 6 i.e. 'How2 FT Charades' was the best performing model. Note that the first column has a reference number to the model in Table 6.6.

set. The trends we observe on the prototype test set are same as those observed on the undisclosed test set. Additionally, this table also contains the human evaluation scores. The human evaluators were asked to rate the system generated answers as well as the groundtruth references answers which scored a topline of 3.938, using the human evaluation strategies designed by the challenge organizers. Our best model scores 3.491 on this metric while the baseline scores only 2.848. This further shows that our models score well not only in quantitative scores but also in qualitative scores.

Error Analysis We perform certain qualitative analysis of the different models: text-only, video-only and text-and-video, each with fine-tuning, to better understand the quality of the results and the model behavior (refer Table 6.8). We compute the number of unique words in the answers generated by each of the three models. We observe that multimodal fusion and fine-tuning with How2 both help increase the number of unique words. Another metric we use is the average length of outputs (avg.). Fine-tuning leads to longer outputs in text-only and video-only models. These models also led to higher gains over Charades only models in Table 6.6. Our final metric is the percentage (%) of sentences changed in a given system when compared with the text-only model trained only on Charades data. We compute this metric by counting all tokens changed, as well as by counting only content-based tokens, *i.e.* not counting stop words or punctuation as changed. We see the maximum percentage of changed sentences are in the video-only models. The difference percentage change by considering only content words is approximately 10-15% absolute.

Modality	Model	# unique words	Avg. o/p len	% sent. changed	% sent changed in content word
Text only	Charades	384	8.98	-	-
	How2 FT Charades	726	9.23	79.46%	65.30%
Video only	Charades	269	9.22	83.60%	72.35%
	How2 FT Charades	331	9.37	87.00%	74.91%
Text & Video	Charades	488	8.95	76.37%	59.00%
	How2 FT Charades	740	8.98	77.72%	60.21%

Table 6.8: Qualitative evaluation of different systems. % sentences (sent) changed are with respect to text-only Charades model.

6.4 Chapter Conclusion

We extend the series of learning tasks for video understanding to video summarization and video question answering in this chapter. We present models for the abstractive nature of these natural language generation tasks and explore a transfer learning method between the two to leverage common modalities and common task characteristics.

We present the Hierarchical Latent Representation Fusion model to design video-level control over the temporal sequences of inputs modalities: audio, video and text. This model is an evolution over the Monotonic Input Fusion and Latent Representation Fusion models. We discuss the drawbacks of these two models for abstractive tasks and the need and benefits of the Hierarchical model. We demonstrate the application of this model to the two abstraction tasks and show consistent improvement with Hierarchical multimodal modeling over established baselines as well as uni-modal models.

Chapter 7

Proposed Work

In Chapters 4, 5, and 6, we discussed multimodal grounding for various video understanding tasks. Across chapters, task complexity and consequently multimodal modeling complexity and modalities involved increased. As proposed work, we extend this further to a *Rationalization* task. With the rationalization task, we extend beyond abstract-learning tasks to also generate explainable rationales for the candidate answers in a Question Answering task. These rationales can be a powerful interpretability tool as well as a user-targeted explanation technique that provides support for the model hypotheses.

Visual Commonsense Reasoning (VCR) is an example of multimodal rationalization that has been introduced in recent years (Li et al., 2018; Zellers et al., 2019). VCR is designed as a question answering task: given an image and a set of questions, a correct answer is to be chosen from amongst four choices, and a corresponding rationale has to be chosen, again from four choices, for that particular answer. The answer and rationale choices are human annotated and to simplify the reasoning task, this problem is designed as a classification problem.

As proposed work, we extend this problem to an open-ended language generation task that goes beyond classification by generating open-vocabulary rationales for Video Question Answering. This is a very hard problem for language generation as there are multiple potential correct rationales for a given question-answer pair. As with all open-ended generation problems, it also suffers from poor automatic evaluation metrics. Related work has proposed human-evaluation metrics to evaluate the faithfulness (Wu and Mooney, 2018; Jain et al., 2020; Jacovi and Goldberg, 2020) and plausibility of generated rationales (Marasović et al., 2020), but this evaluation is difficult to use during training and for reproducibility of results. Despite these challenges, Rationalization fits as the next relevant task for us to explore in this thesis in continuation to Recognition, Translation, and Summarization, as an extension towards interpretable and explainable multimodal video understanding.

The nature of rationalization is inherently interpretable i.e. it provides commonsense reasoning or explanations for a given question-answer pair. Additionally, there is a dependency between the answer and the corresponding rationale that has not been studied in related work yet,

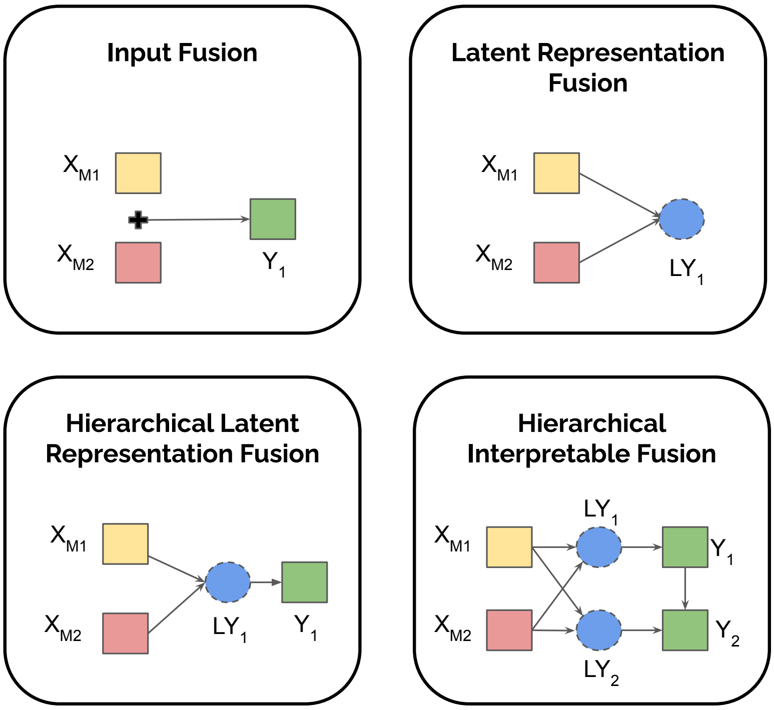


FIGURE 7.1: Expanding the fusion models to the final model: Hierarchical Interpretable Fusion model.

especially for joint, open-ended rationalization. To capture these two properties of (1) Interpretability, and (2) Dependency, we propose to model this task via a *Hierarchical Interpretable Fusion* model shown in Figure 7.1. Interpretability for this task can be achieved in two ways; first by generating an answer as an intermediate step before rationalization, and second by the generated rationale itself that explains the reasoning for a given answer to be the correct answer. Dependency in turn refers to the fact that the correctness of the answer and of the rationale go hand-in-hand. Specifically, the rationale generated needs to depend on the answer already generated. This constraint is implicitly controlled with the proposed Hierarchical Interpretable Fusion model (explained below) but explicit control for such modeling is a much more difficult problem (due to the open-ended nature of this language generation). To evaluate if this dependency is modeled within the proposed approach, we will design the necessary human evaluation experiments.

With the three previous multimodal fusion models, we focused explicitly on methods of fusion and controllability, each gradually increasing in complexity to handle the task and modality complexities. With the *Hierarchical Interpretable Fusion* model, we combine the *Latent Representation Fusion* and *Hierarchical Latent Representation Fusion* together, while also aiming for interpretable generation. For the scope of this work, we define interpretable generation as the generation of some observable, not latent, intermediate output Y_1 followed by final observable output Y_2. Each of these outputs undergo latent modality fusion represented by LY_1 and LY_2. LY_1 and Y_1, or LY_2 and Y_2 could individually form the *Hierarchical Latent Representation Fusion* model from the previous chapter. The dependency between the two outputs Y_1 and Y_2 as shown in Figure 7.1 represents the combination of the previous fusion models into this one.

We will develop and apply this model to the rationalization task where X_{M1} and X_{M2} will be the input image and questions, Y_1 will be the answer, and Y_2 the corresponding rationale. Given the nature of the rationalization task, this particular model is well suited for the interpretability as well as dependency properties.

Plan

We have been working on developing the Hierarchical Interpretable Fusion model. Currently, we are testing it on a simpler image captioning task with the intermediate output Y_1 being entities, followed by the natural language caption. We will extend this model to the rationalization task. I will continue work on rationalization at AI2 this summer with a particular focus on answer-rationale co-generation.

Timeline

Thesis timeline for 2021-2022

April 2021	Thesis Proposal
Now-May 2021	Hierarchical Interpretable Learning for Rationale generation
May-Aug 2021	Summer Internship at AI2 on Rationalization
Aug-Dec 2021	Wrap up work on Rationalization
Jan-Feb 2022	Thesis Writing
Mar-Apr 2022	Thesis Defense

Bibliography

2018. How2. https://github.com/srvk/how2.

Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. 2016. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675.*

Huda Alamri, Vincent Cartillier, Abhishek Das, Jue Wang, Anoop Cherian, Irfan Essa, Dhruv Batra, Tim K Marks, Chiori Hori, Peter Anderson, et al. 2019. Audio visual scene-aware dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7558–7567.

Huda Alamri, Vincent Cartillier, Raphael Gontijo Lopes, Abhishek Das, Jue Wang, Irfan Essa, Dhruv Batra, Devi Parikh, Anoop Cherian, Tim K Marks, et al. 2018. Audio visual scene-aware dialog (avsd) challenge at dstc7. *arXiv preprint arXiv:1806.00525.*

Huda Alamri, Chiori Hori, Tim K Marks, Devi Parikh, and Dhruv Batra. 2017. Audio visual scene-aware dialog (avsd) track for natural language generation in dstc7. -.

Samuel Albanie, Yang Liu, Arsha Nagrani, Antoine Miech, Ernesto Coto, Ivan Laptev, Rahul Sukthankar, Bernard Ghanem, Andrew Zisserman, Valentin Gabeur, et al. 2020. The end-of-end-to-end: A video understanding pentathlon challenge (2020). *arXiv preprint arXiv:2008.00744.*

Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. 2013. Deep canonical correlation analysis. In *ICML*, pages 1247–1255.

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*, pages 2425–2433.

Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. 2021. Vivit: A video vision transformer. *arXiv preprint arXiv:2103.15691.*

Raman Arora and Karen Livescu. 2013. Multi-view cca-based acoustic features for phonetic recognition across speakers and domains. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 7135–7139. IEEE.

Kartik Audhkhasi, Brian Kingsbury, Bhuvana Ramabhadran, George Saon, and Michael Picheny. 2017. Building competitive direct acoustics-to-word models for english conversational speech recognition. *arXiv preprint arXiv:1712.03133*.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *arXiv preprint arXiv:2006.11477*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014a. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014b. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio. 2016. End-to-end attention-based large vocabulary speech recognition. In *Proc. ICASSP*, pages 4945–4949. IEEE.

Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. 2019. Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):423–443.

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.

Lawrence W Barsalou, W Kyle Simmons, Aron K Barbey, and Christine D Wilson. 2003. Grounding conceptual knowledge in modality-specific systems. *Trends in cognitive sciences*, 7(2):84–91.

Samy Bengio and Georg Heigold. 2014. Word embeddings for speech recognition. In *Fifteenth Annual Conference of the International Speech Communication Association*.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003a. Latent dirichlet allocation. *JMLR*, 3(Jan):993–1022.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003b. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Cynthia J Brame. 2016. Effective educational videos: Principles and guidelines for maximizing student learning from video content. *CBE—Life Sciences Education*, 15(4):es6.

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. 2015. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 961–970.

Ozan Caglayan, Ramon Sanabria, Shruti Palaskar, Loïc Barraul, and Florian Metze. 2019. Multimodal grounding for sequence-to-sequence speech recognition. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8648–8652. IEEE.

Ziqiang Cao, Furu Wei, Li Dong, Sujian Li, and Ming Zhou. 2015. Ranking with recursive neural networks and its application to multi-document summarization. In *Twenty-ninth AAAI conference on artificial intelligence*.

Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308.

W. Chan, N. Jaitly, Q. Le, and O. Vinyals. 2016. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4960–4964.

William Chan, Daniel Park, Chris Lee, Yu Zhang, Quoc Le, and Mohammad Norouzi. 2021. Speechstew: Simply mix all available speech recognition data to train one large neural network. *arXiv preprint arXiv:2104.02133*.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. 2015. Microsoft COCO captions: Data collection and evaluation server. *Computing Research Repository (CoRR)*.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Uniter: Learning universal image-text representations.

Zhehuai Chen, Qi Liu, Hao Li, and Kai Yu. 2018. On modular training of neural acoustics-to-word model for lvcsr. *arXiv preprint arXiv:1803.01090*.

Kyunghyun Cho, Bart van Merrienboer, 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 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar.

Jan Chorowski, Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. End-to-end continuous speech recognition using attention-based recurrent NN: First results. *arXiv preprint arXiv:1412.1602*.

Jan Chorowski and Navdeep Jaitly. 2016. Towards better decoding and language model integration in sequence to sequence models. *arXiv preprint arXiv:1612.02695*.

Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. 2015. Attention-based models for speech recognition. In *Advances in neural information processing systems*, pages 577–585.

J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman. 2017. Lip reading sentences in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Yu-An Chung and James Glass. 2018. Speech2vec: A sequence-to-sequence framework for learning word embeddings from speech. *arXiv preprint arXiv:1803.08976*.

Christopher Cieri, David Miller, and Kevin Walker. 2004. The fisher corpus: a resource for the next generations of speech-to-text. In *LREC*, volume 4, pages 69–71.

Alexis Conneau and Douwe Kiela. 2018. Senteval: An evaluation toolkit for universal sentence representations. *arXiv preprint arXiv:1803.05449*.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. 2016. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José M.F. Moura, Devi Parikh, and Dhruv Batra. 2017. Visual Dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

P. Das, C. Xu, R. F. Doell, and J. J. Corso. 2013. A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009a. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009b. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition*, pages 248–255.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the ninth workshop on statistical machine translation*, pages 376–380. Association for Computational Linguistics.

Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634.

Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. 2014. Bags of spacetime energies for dynamic scene recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2681–2688.

Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. 2016. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941.

Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. 2020. Multi-modal transformer for video retrieval. In *European Conference on Computer Vision (ECCV)*, volume 5. Springer.

Philip Gage. 1994. A new algorithm for data compression. *The C Users Journal*, 12(2):23–38.

Spandana Gella, Mike Lewis, and Marcus Rohrbach. 2018. A dataset for telling the stories of social media videos. In *2018 Conference on Empirical Methods in Natural Language Processing*, pages 968–974.

John J Godfrey, Edward C Holliman, and Jane McDaniel. 1992. Switchboard: Telephone speech corpus for research and development. In *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 1, pages 517–520. IEEE.

Jade Goldstein, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. 2000. Multi-document summarization by sentence extraction. In *Proceedings of the 2000 NAACL-ANLP Workshop on Automatic summarization*, pages 40–48. Association for Computational Linguistics.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 NAACL:HLT*, pages 708–719.

Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. 2018. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6047–6056.

Çaglar Gülçehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, Volume 1: Long Papers*.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *arXiv preprint arXiv:1503.03535*.

Pengcheng Guo, Florian Boyer, Xuankai Chang, Tomoki Hayashi, Yosuke Higuchi, Hirofumi Inaguma, Naoyuki Kamo, Chenda Li, Daniel Garcia-Romero, Jiatong Shi, et al. 2020. Recent developments on espnet toolkit boosted by conformer. *arXiv preprint arXiv:2010.13956*.

Abhinav Gupta, Yajie Miao, Leonardo Neves, and Florian Metze. 2017a. Visual features for context-aware speech recognition. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5020–5024.

Abhinav Gupta, Yajie Miao, Leonardo Neves, and Florian Metze. 2017b. Visual features for context-aware speech recognition. In *IEEE ICASSP*, pages 5020–5024.

Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. 2014. Creating summaries from user videos. In *European conference on computer vision*, pages 505–520. Springer.

Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. 2014. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*.

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.

David Harwath and James Glass. 2015. Deep multimodal semantic embeddings for speech and images. *arXiv preprint arXiv:1511.03690*.

John Hattie. 2012. *Visible learning for teachers: Maximizing impact on learning*. Routledge.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016a. Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition*, pages 770–778.

Wanjia He, Weiran Wang, and Karen Livescu. 2016b. Multi-view recurrent neural acoustic word embeddings. *arXiv preprint arXiv:1611.04496*.

Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson. 2017. Cnn architectures for large-scale audio classification. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE.

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

Nils Holzenberger, Shruti Palaskar, Pranava Madhyastha, Florian Metze, and Raman Arora. 2019. Learning from multiview correlations in open-domain videos. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8628–8632. IEEE.

Paul Horst. 1961. Generalized canonical correlations and their applications to experimental data. Technical report, Washington Uni Seattle.

Harold Hotelling. 1992. Relations between two sets of variates. In *Breakthroughs in statistics*, pages 162–190. Springer.

Hirofumi Inaguma, Shun Kiyono, Kevin Duh, Shigeki Karita, Nelson Enrique Yalta Soplin, Tomoki Hayashi, and Shinji Watanabe. 2020. Espnet-st: All-in-one speech translation toolkit. *arXiv preprint arXiv:2004.10234*.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness? *arXiv preprint arXiv:2004.03685*.

Sarthak Jain, Sarah Wiegreffe, Yuval Pinter, and Byron C Wallace. 2020. Learning to faithfully rationalize by construction. *arXiv preprint arXiv:2005.00115*.

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

Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. 2014. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732.

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*.

Suyoun Kim, Takaaki Hori, and Shinji Watanabe. 2017. Joint ctc-attention based end-to-end speech recognition using multi-task learning. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pages 4835–4839. IEEE.

G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush. OpenNMT: Open-Source Toolkit for Neural Machine Translation. *ArXiv e-prints*.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision*, pages 706–715.

Julian Kupiec, Jan Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 68–73. ACM.

Haoran Li, Junnan Zhu, Cong Ma, Jiajun Zhang, and Chengqing Zong. 2017. Multi-modal summarization for asynchronous collection of text, image, audio and video. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1092–1102.

Qing Li, Qingyi Tao, Shafiq R. Joty, Jianfei Cai, and Jiebo Luo. 2018. VQA-E: Explaining, Elaborating, and Enhancing Your Answers for Visual Questions. In *ECCV*.

Jindřich Libovický 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, Vancouver, Canada.

Jindřich Libovický, Shruti Palaskar, Spandana Gella, and Florian Metze. 2018. Multimodal abstractive summarization of open-domain videos. In *NeurIPS Workshop on Visually Grounded Interaction and Language (ViGIL)*.

Chin-Yew Lin and Eduard Hovy. 2002. From single to multi-document summarization. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 457–464.

Chin-Yew Lin and Franz Josef Och. 2004. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics*, pages 605–612, Barcelona, Spain.

Edward Loper and Steven Bird. 2002. Nltk: the natural language toolkit. *Natural Language Processing with Python. O'Reilly Media Inc.*

Jiasen Lu, Vedanuj Goswami, Marcus Rohrbach, Devi Parikh, and Stefan Lee. 2020. 12-in-1: Multi-task vision and language representation learning. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Liang Lu, Xingxing Zhang, and Steve Renals. 2016. On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 5060–5064. IEEE.

Zheng Lu and Kristen Grauman. 2013. Story-driven summarization for egocentric video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2714–2721.

Hans Peter Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Inderjeet Mani. 1999. *Advances in automatic text summarization*. MIT press.

Ana Marasović, Chandra Bhagavatula, Jae sung Park, Ronan Le Bras, Noah A. Smith, and Yejin Choi. 2020. Natural language rationales with full-stack visual reasoning: From pixels to semantic frames to commonsense graphs. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2810–2829, Online. Association for Computational Linguistics.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems*, pages 6294–6305.

Grégoire Mesnil, Xiaodong He, Li Deng, and Yoshua Bengio. 2013. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding.

Yajie Miao, Lu Jiang, Hao Zhang, and Florian Metze. 2014. Improvements to speaker adaptive training of deep neural networks. In *2014 IEEE Spoken Language Technology Workshop*. IEEE.

Yajie Miao and Florian Metze. 2015. Distance-aware DNNs for robust speech recognition. In *Proc. INTERSPEECH*, Dresden, Germany. ISCA.

Yajie Miao and Florian Metze. 2016a. Open-domain audio-visual speech recognition: A deep learning approach. In *Interspeech*. ISCA.

Yajie Miao and Florian Metze. 2016b. Open-domain audio-visual speech recognition: A deep learning approach. In *Proc. INTERSPEECH*, pages 3414–3418.

Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 2020. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9879–9889.

Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. 2019. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE international conference on computer vision*, pages 2630–2640.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Arthur G Money and Harry Agius. 2008. Video summarisation: A conceptual framework and survey of the state of the art. *Journal of Visual Communication and Image Representation*, 19(2):121–143.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Ça glar Gulçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *CoNLL 2016*, page 280.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Ranking sentences for extractive summarization with reinforcement learning. *CoRR*.

Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. 2011. Multimodal deep learning. In *Proceedings of the 28th ICML*, pages 689–696.

Duy-Kien Nguyen and Takayuki Okatani. 2019. Multi-task learning of hierarchical vision-language representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10492–10501.

Mayu Otani, Yuta Nakashima, Esa Rahtu, Janne Heikkilä, and Naokazu Yokoya. 2016. Learning joint representations of videos and sentences with web image search. In *European Conference on Computer Vision*, pages 651–667. Springer.

Shruti Palaskar, Jindřich Libovický, Spandana Gella, and Florian Metze. 2019a. Multimodal abstractive summarization for how2 videos. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6587–6596, Florence, Italy. Association for Computational Linguistics.

Shruti Palaskar and Florian Metze. 2018. Acoustic-to-word recognition with sequence-to-sequence models. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 397–404. IEEE.

Shruti Palaskar, Vikas Raunak, and Florian Metze. 2019b. Learned in speech recognition: Contextual acoustic word embeddings. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6530–6534. IEEE.

Shruti Palaskar, Ramon Sanabria, and Florian Metze. 2018. End-to-end multimodal speech recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5774–5778. IEEE.

Shruti Palaskar, Ramon Sanabria, and Florian Metze. 2020. Transfer learning for multimodal dialog. *Computer Speech & Language*, page 101093.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL*.

Ngoc-Quan Pham, Thai-Son Nguyen, Thanh-Le Ha, Juan Hussain, Felix Schneider, Jan Niehues, Sebastian Stüker, and Alexander Waibel. 2019. The iwslt 2019 kit speech translation system. In *Proceedings of the 16th International Workshop on Spoken Language Translation (IWSLT 2019)*.

Subhojeet Pramanik, Priyanka Agrawal, and Aman Hussain. 2019. Omninet: A unified architecture for multi-modal multi-task learning. *arXiv preprint arXiv:1907.07804*.

Patti J Price. 1990. Evaluation of spoken language systems: The atis domain. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27, 1990.*

Pushpendre Rastogi, Benjamin Van Durme, and Raman Arora. 2015. Multiview lsa: Representation learning via generalized cca. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 556–566.

Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. 2013. Grounding action descriptions in videos. *TACL*, 1:25–36.

Peter J Rentfrow, Lewis R Goldberg, and Ran Zilca. 2011. Listening, watching, and reading: The structure and correlates of entertainment preferences. *Journal of personality*, 79(2):223–258.

Anna Rohrbach, Marcus Rohrbach, Wei Qiu, Annemarie Friedrich, Manfred Pinkal, and Bernt Schiele. 2014. Coherent multi-sentence video description with variable level of detail. In *Pattern Recognition - 36th German Conference, GCPR 2014*, pages 184–195.

German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. 2016. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3234–3243.

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. Association for Computational Linguistics.

Shagan Sah, Sourabh Kulhare, Allison Gray, Subhashini Venugopalan, Emily Prud'Hommeaux, and Raymond Ptucha. 2017. Semantic text summarization of long videos. In *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*, 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. In *Proc. Visually Grounded Interaction and Language (ViGIL) Workshop at NIPS*, Montreal, Canada. Https://arxiv.org/abs/1811.00347.

Ramon Sanabria, Shruti Palaskar, and Florian Metze. 2019. Cmu sinbad's submission for the dstc7 avsd challenge. In *Proc. 7th Dialog System Technology Challenges Workshop at AAAI*, Honolulu, Hawaii, USA.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *55th Annual Meeting of the Association for Computational Linguistics*, pages 1073–1083.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association* *for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. 2016. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision*, pages 510–526. Springer.

Karen Simonyan and Andrew Zisserman. 2014a. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576.

Karen Simonyan and Andrew Zisserman. 2014b. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Richard Socher and Li Fei-Fei. 2010. Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 966–973. IEEE.

Jingkuan Song, Yi Yang, Zi Huang, Heng Tao Shen, and Richang Hong. 2011. Multiple feature hashing for real-time large scale near-duplicate video retrieval. In *19th ACM international conference on Multimedia*, pages 423–432. ACM.

Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. 2015. Tvsum: Summarizing web videos using titles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5179–5187.

Lucia Specia, Raman Arora, Loic Barrault, Ozan Caglayan, Amanda Duarte, Desmond Elliott, Spandana Gella, Nils Holzenberger, Chiraag Lala, Sun Jae Lee, et al. 2020. Grounded sequence to sequence transduction. *IEEE Journal of Selected Topics in Signal Processing*.

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*, pages 543–553, Berlin, Germany.

Md Arafat Sultan, Steven Bethard, and Tamara Sumner. 2014. Back to basics for monolingual alignment: Exploiting word similarity and contextual evidence. *Transactions of the Association for Computational Linguistics*, 2:219–230.

Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7464–7473.

Ilya Sutskever, James Martens, and Geoffrey E Hinton. 2011. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pages 1017–1024. JMLR.org.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014a. Sequence to sequence learning with neural networks. In *Proc. NIPS*.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014b. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pages 3104–3112, Montreal, Canada.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Atousa Torabi, Niket Tandon, and Leonid Sigal. 2016. Learning language-visual embedding for movie understanding with natural-language. *CoRR*, abs/1609.08124.

Shubham Toshniwal, Hao Tang, Liang Lu, and Karen Livescu. 2017. Multitask learning with low-level auxiliary tasks for encoder-decoder based speech recognition. *arXiv preprint arXiv:1704.01631*.

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

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*, pages 4566–4575.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015a. Pointer networks. In *Advances in Neural Information Processing Systems*, pages 2692–2700. Curran Associates, Inc.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015b. Show and tell: A neural image caption generator. In *Proc. CVPR*, pages 3156–3164.

Meng Wang, Richang Hong, Guangda Li, Zheng-Jun Zha, Shuicheng Yan, and Tat-Seng Chua. 2012. Event driven web video summarization by tag localization and key-shot identification. *IEEE Transactions on Multimedia*, 14(4):975–985.

Weiran Wang, Raman Arora, Karen Livescu, and Jeff Bilmes. 2015. On deep multi-view representation learning. In *International Conference on Machine Learning (ICML)*, pages 1083–1092.

Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R Hershey, and Tomoki Hayashi. 2017. Hybrid ctc/attention architecture for end-to-end speech recognition. *IEEE Journal of Selected Topics in Signal Processing*, 11(8):1240–1253.

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.

Jialin Wu and Raymond J Mooney. 2018. Faithful multimodal explanation for visual question answering. *arXiv preprint arXiv:1809.02805*.

Zixiu Wu, Ozan Caglayan, Julia Ive, Josiah Wang, and Lucia Specia. 2019. Transformer-based cascaded multimodal speech translation. *arXiv preprint arXiv:1910.13215*.

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

Michihiro Yasunaga, Rui Zhang, Kshitijh Meelu, Ayush Pareek, Krishnan Srinivasan, and Dragomir R. Radev. 2017. Graph-based neural multi-document summarization. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 452–462.

Koichiro Yoshino, Chiori Hori, Julien Perez, Luis Fernando D'Haro, Lazaros Polymenakos, Chulaka Gunasekara, Walter S. Lasecki, Jonathan Kummerfeld, Michael Galley, Chris Brockett, Jianfeng Gao, Bill Dolan, Sean Gao, Tim K. Marks, Devi Parikh, and Dhruv Batra. 2018. The 7th dialog system technology challenge. *arXiv preprint*.

Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. 2016. Video paragraph captioning using hierarchical recurrent neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4584–4593.

Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. 2015. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702.

Matthew D Zeiler. 2012. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6720–6731.

Kuo-Hao Zeng, Tseng-Hung Chen, Juan Carlos Niebles, and Min Sun. 2016. Generation for user generated videos. In *European conference on computer vision*, pages 609–625. Springer.

Thomas Zenkel, Ramon Sanabria, Florian Metze, Jan Niehues, Matthias Sperber, Sebastian Stüker, and Alex Waibel. 2017. Comparison of decoding strategies for ctc acoustic models. *arXiv preprint arXiv:1708.04469*.

Albert Zeyer, Kazuki Irie, Ralf Schlüter, and Hermann Ney. 2018. Improved training of end-to-end attention models for speech recognition. *arXiv preprint arXiv:1805.03294*.

Jianguo Zhang, Pengcheng Zou, Zhao Li, Yao Wan, Ye Liu, Xiuming Pan, Yu Gong, and Philip S Yu. 2018. Product title refinement via multi-modal generative adversarial learning. *arXiv preprint arXiv:1811.04498*.

Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2017. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Luowei Zhou, Chenliang Xu, and Jason J. Corso. 2018a. Towards automatic learning of procedures from web instructional videos. In *Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18)*, pages 7590–7598.

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

Geoffrey Zweig, Chengzhu Yu, Jasha Droppo, and Andreas Stolcke. 2017. Advances in all-neural speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pages 4805–4809. IEEE.