

Hallucination Is All You Need: Using Generative Models for Test Time Data Augmentation

Dhruv Jhamb
David Chan, Ed.
John F. Canny, Ed.
Avideh Zakhor, Ed.

Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2022-85

<http://www2.eecs.berkeley.edu/Pubs/TechRpts/2022/EECS-2022-85.html>

May 12, 2022



Copyright © 2022, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

Acknowledgement

I would like to acknowledge my advisor Professor John F. Canny and Professor Avidah Zakhor for their support with my research. I would like to acknowledge my mentor David Chan for his continuous guidance and feedback over the past couple years. Last but not least, I would like to thank my family and friends for their support during my entire academic journey at UC Berkeley.

Hallucination Is All You Need: Using Generative Models for Test Time Data Augmentation

by

Dhruv Jhamb

A dissertation submitted in partial satisfaction of the
requirements for the degree of

Master of Science

in

Electrical Engineering and Computer Science

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor John F. Canny, Chair
Professor Avidoh Zakhor

Spring 2022

**Hallucination Is All You Need: Using Generative Models for Test Time
Data Augmentation**

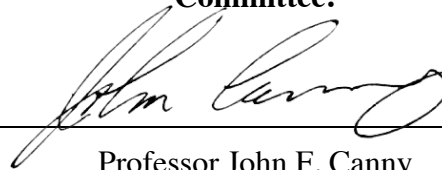
by Dhruv Jhamb

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences,
University of California at Berkeley, in partial satisfaction of the requirements for the
degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

Committee:



Professor John F. Canny
Research Advisor

5/12/2022

(Date)



Professor Avidah Zakhori
Second Reader

5/12/2022

(Date)

Hallucination Is All You Need: Using Generative Models for Test Time Data Augmentation

Copyright 2022

by

Dhruv Jhamb

Abstract

Hallucination Is All You Need: Using Generative Models for Test Time Data Augmentation

by

Dhruv Jhamb

Master of Science in Electrical Engineering and Computer Science

University of California, Berkeley

Professor John F. Canny, Chair

Multimodal learning, which consists of building models that can take information from different modalities as input, is growing in popularity due to its potential. Deep learning-based multimodal models can be applied to a variety of downstream tasks such as video description, sentiment analysis, event detection, cross-modal translation, and cross-modal retrieval. Inherently, we can expect multimodal models to outperform unimodal models because the additional modalities provide more information. The way humans experience and learn is multimodal, as we combine multiple senses to experience the world around us. In the ideal case, we assume completeness of data, meaning that all modalities are entirely present. However, this assumption is not always guaranteed at test time, meaning that it is necessary to create multimodal models robust to missing modalities in real-world applications. We choose to address this missing modality problem during test time by comparing several feature reconstruction methods on multimodal emotion recognition datasets.

Dedication

To my parents and sister for their love and support.

Contents

Contents	ii
List of Figures	iii
List of Tables	iv
1 Introduction	1
2 Related Work	2
2.1 Multimodal Learning	2
2.2 Modality Generation	3
3 Methods	5
3.1 Feature Extraction	5
3.2 Baseline for Classification	6
3.3 Feature Reconstruction	7
3.4 Oracle Models	11
4 Experimental Design	14
4.1 Datasets	14
4.2 Data Processing	15
4.3 Experiment Design	15
5 Results and Analysis	17
5.1 Results	17
5.2 Analysis	20
6 Future Work	24
7 Conclusion	25
Bibliography	26

List of Figures

2.1	Example of how a video can be split into different modalities for emotion recognition [37].	2
3.1	Feature extraction pipeline from raw video data to multimodal features. Vision and audio features are concatenated to get a feature vector of length 912.	5
3.2	MLP architecture. The variable num_classes depends on the choice of dataset.	7
3.3	GAN architecture. For AV-GAN: in_shape = 100, in_shape_2 = 912. For V-GAN: in_shape = 612, in_shape_2 = 1424. For A-GAN: in_shape = 500, in_shape_2 = 1312.	10
3.4	VAE architecture. For AV-VAE: in_shape = 256. For V-VAE: in_shape = 768. For A-VAE: in_shape = 656.	12

List of Tables

5.1	Legend	17
5.2	Baseline Accuracies for Trained MLP	18
5.3	RAVDESS Feature Reconstruction Method Accuracies	18
5.4	RAVDESS Oracle Methods	18
5.5	eINTERFACE Feature Reconstruction Method Accuracies	18
5.6	eINTERFACE Oracle Methods	18
5.7	CMU-MOSI (7 Classes) Feature Reconstruction Method Accuracies	19
5.8	CMU-MOSI (7 Classes) Oracle Methods	19
5.9	CMU-MOSI (2 Classes) Feature Reconstruction Method Accuracies	19
5.10	CMU-MOSI (2 Classes) Oracle Methods	19

Acknowledgments

I would like to acknowledge my advisor Professor John F. Canny and Professor Avidesh Zakhor for their support with my research. I would like to acknowledge my mentor David Chan for his continuous guidance and feedback over the past couple years. Last but not least, I would like to thank my family and friends for their support during my entire academic journey at UC Berkeley.

Chapter 1

Introduction

Within the field of artificial intelligence (AI), multimodal learning is becoming a popular tool for solving different tasks. This is in part due to the abundance of available data that comes from different modalities. While combining different modalities increases the amount of knowledge models have access to, data incompleteness is a problem that diminishes these gains in information. In many multimodal datasets and the real world, there will not be complete modalities for each sample. In other work, generative modeling approaches, such as autoencoders and GANs, have been used to reconstruct the missing modality. Other more simple approaches have also been used as baselines, such as zero padding, which is padding feature representations of the missing modality with the value zero.

The purpose of this project is to investigate the commonly used approaches for handling missing modalities by measuring performance on a downstream task. This will allow us to see the robustness of commonly used approaches in dealing with missing modalities during test time. This project will investigate modality reconstruction, namely which approaches can generate missing modality better than others, and see if trends hold across different datasets.

The downstream task we will focus on is emotion recognition, using the multimodal datasets: RAVDESS, eNTERFACE'05, and CMU-MOSI. Making emotion recognition our task stems from it being an important prerequisite for AI systems in the future, with it being a central part of natural human-computer interactions. Emotion recognition can be used for tasks such as security measures, HR assistance, customer service, audience engagement, video game testing, healthcare [9]. Human emotions can range from being simple to being complex, with humans relying on visual and auditory cues to distinguish a person's sentiment. Therefore, we aim to achieve the highest performance we can and present approaches that achieve state-of-the-art on RAVDESS and close to the state-of-the-art on eNTERFACE'05.

For feature reconstruction, several methods will be explored: zero padding, replacement with training mean, replacement with random uniform values, gaussian mixture models (GMMs), generative adversarial networks (GANs), variational autoencoders (VAEs), and other sampling baselines that will be discussed in Chapter 3.

Chapter 2

Related Work

2.1 Multimodal Learning

One of the biggest reasons for the relevance of multimodal learning is the number of tasks that it can be applied to. Some popular applications are image description, video description, visual question answering, speech synthesis, event detection, and emotion recognition [31].

Missing Modalities

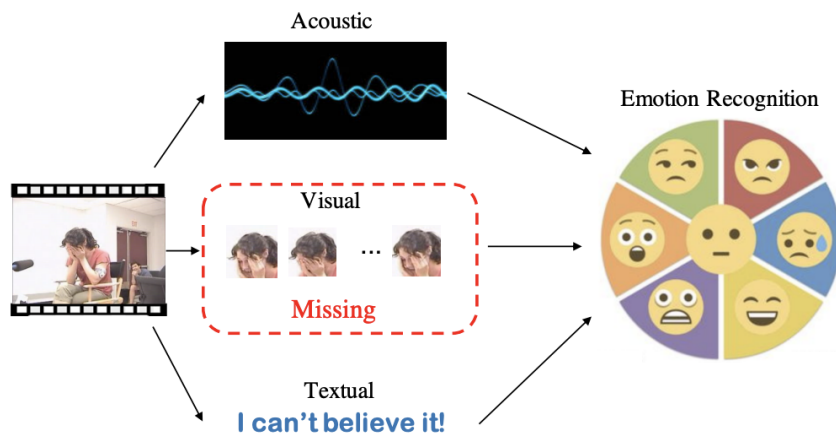


Figure 2.1: Example of how a video can be split into different modalities for emotion recognition [37].

Du et al. propose a semi-supervised multiview deep generative framework for multimodal emotion recognition with incomplete data [7]. Our work is different because Du et al. only consider the scenario where a certain modality is missing or incomplete. We allow for different

modalities to be missing at the same time, which is closer to real-world situations. Ma et al. address the problem of audio-visual emotion recognition with missing labels and missing modalities [19]. They identify two challenges: 1) large amounts of emotional data have missing labels, and 2) emotional data often has missing modalities. We specifically focus on the second challenge of missing modalities and go more in depth than [19].

There has been a wide range of research into handling missing modalities. Tsai et al. focus on addressing missing or noisy modalities during testing using a Multimodal Factorization Model [34]. Shi et al. use a contrastive framework to make use of unpaired multimodal data [28]. Mengmeng Ma et al. offer a flexible solution to handle severely missing modalities during train time and test time using a feature reconstruction network to approximate the missing modality [21]. Fei Ma et al. use an approach based on maximum likelihood estimation to incorporate knowledge in modality-missing data [20].

Methods to deal with missing modalities can mainly be divided into three groups. The first group features the data augmentation approach, which randomly ablates the inputs to mimic missing modality cases [26]. The second group is based on generative methods to directly predict the missing modalities given the available modalities [17, 5, 32, 7]. The third group aims to learn the joint multimodal representations that can contain related information from these modalities [1, 27, 13, 35]. Our work falls under the second group as we use generative methods to reconstruct missing modalities.

We apply some of the common approaches from the above literature to our specific datasets; however, we focus solely on missing modality during test time to better simulate real-world circumstances. We combine the notion of feature reconstruction methods grounded in statistics like [21, 20] and re-use some of the baselines discussed in the above papers.

Emotion Recognition

We specifically look into audio-visual emotion recognition since our chosen multimodal datasets are focused on emotion classification. [37] uses multimodal fusion by applying a Missing Modality Imagination Network that learns robust joint multimodal representations. It can predict the representation of any missing modality given available modalities under different missing modality conditions. In contrast with [37], we do not jointly learn multimodal representations and instead separately concatenate vision and audio representations to create our features. However, we do use the same principle of “imagining” the missing modality given available modalities.

2.2 Modality Generation

To address the problem of missing modality during test time, we employ feature reconstruction techniques to generate modalities. The data this project is concerned with are videos, which by nature include both a visual and auditory modality. Vision and sound are closely

related, sparking the question of whether accurate audio can be generated from a sequence of image frames and whether accurate vision can be generated from audio.

Image Generation

Gregor et al. propose the architecture DRAW which uses a variational autoencoder to improve upon state-of-the-art on MNIST [11]. Other work such as Oord et al.’s Pixel CNN Decoder falls under conditional image generation. Their model can be conditioned on any vector or latent embeddings generated by other networks [25]. Bodla et al. implement a Fused GAN which fused a generator for unconditional image generation and a generator for conditional image generation [4].

To generate the vision features, we implement a variety of methods, including a conditional Variational Autoencoder (VAE) and a conditional Generative Adversarial Network (GAN). We reason that generating the visual aspect given the audio would perform better than using a non-conditional approach. In a sense, we hope these models can “imagine” the visual data from the audio data.

Audio Generation

Zhou et al. use a hierarchical RNN to predict raw audio signals from inputs videos [40]. Generative models such as GANs are also commonly used to generate audio. Donahue et al. use WaveGAN to generate audio suitable for sound effect generation [6].

Intuitively, we expect that audio would be related to vision. In terms of a video, the sound should align with what is occurring in each image frame. Zhu et al. define audio-visual generation as trying to synthesize one modality (audio or visual) from the other modality [41]. We pursue audio-visual generation through the methods we implement which are discussed in Chapter 3, including a conditional VAE and a conditional GAN. Similar to the case of image generation, we hope these models can “imagine” the audio data from the visual data.

Chapter 3

Methods

In this chapter, we will discuss how we extract multimodal features from raw video data, our baseline with complete data, and feature reconstruction methods we use to handle missing modalities at test time.

3.1 Feature Extraction

We use separate neural network architectures for vision features and audio features. We concatenate our vision and audio features to get our multimodal features for each video. Refer to 3.1 for a diagram of our feature extraction architecture.

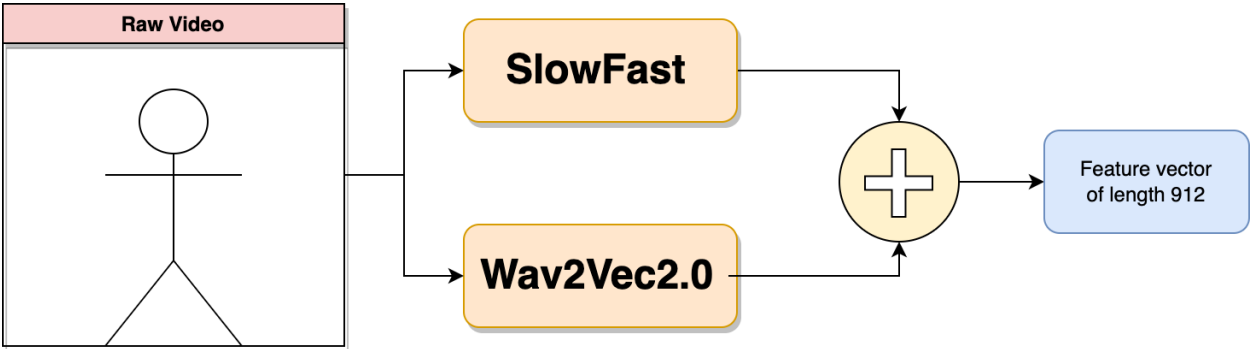


Figure 3.1: Feature extraction pipeline from raw video data to multimodal features. Vision and audio features are concatenated to get a feature vector of length 912.

All feature extraction networks we use are zero-shot, meaning they have not been exposed to our data at all. With additional fine-tuning of these networks, we would expect overall accuracy improvements, although the general trends between methods should still hold.

Vision Features

To extract vision features, we use a PyTorchVideo model (SlowFast [8]) pre-trained on the Kinetics 400 dataset. For the datasets RAVDESS and CMU-MOSI, we use SlowFast R50. For the dataset eNTERFACE, we use SlowFast R101. Model-specific input transforms were applied to the data following Torch Hub documentation. The output of the SlowFast models is a tensor of length 400.

The decision to use SlowFast networks for extracting vision features stems from their strong performance on the tasks action classification and detection in video. They achieve state-of-the-art accuracy on major video recognition benchmarks: Kinetics, Charades, and AVA [8].

We use SlowFast networks pre-trained on the Kinetics 400 dataset because we expect the information that the network learns about videos to carry over to our task of sentiment classification. Since we achieved close to state-of-the-art accuracy on RAVDESS and eNTERFACE while doing feature extraction in a zero-shot manner, it seems that this hypothesis was correct.

Audio Features

To extract audio features, we use a Wav2Vec2.0 model (XLSR-Wav2Vec2) pre-trained in 53 languages and fine-tuned on English using the Common Voice dataset [12, 3]. The necessary data pre-processing which is done before feeding the data into the Wav2Vec2.0 processor and model is detailed in Chapter 4. The output of the Wav2Vec2.0 model is a tensor of length 512.

The decision to use Wav2Vec2.0 is because it achieves state of the art on the full Librispeech benchmark for noisy speech and works well for speech recognition with a limited amount of labeled data [3].

We use fine-tuned Wav2Vec2.0 because we expect the information the network learns about English speech to carry over to our task of sentiment classification. Since we achieved close to state-of-the-art accuracy on RAVDESS and eNTERFACE while doing feature extraction in a zero-shot manner, it seems that this hypothesis was correct.

3.2 Baseline for Classification

We use a basic feedforward neural network, or multilayer perceptron (MLP), for classification on our chosen datasets. The MLP takes in the dataset features from the feature extraction networks detailed above as input and outputs a classification.

Refer to 3.2 for a diagram of the MLP architecture. We train our MLP for 1000 epochs using cross-entropy loss, the Adam optimizer with a learning rate of $1e-4$ and a weight decay of $1e-5$, and a learning rate scheduler with a step size of 250 and a gamma value of 0.1.

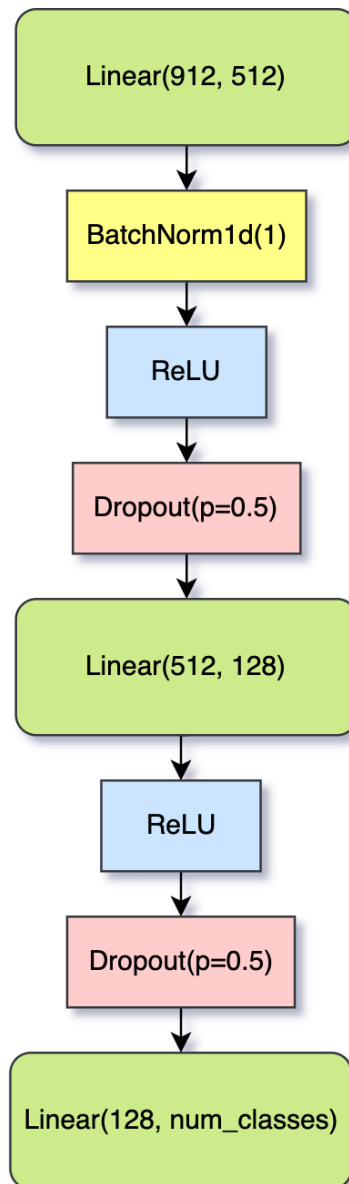


Figure 3.2: MLP architecture. The variable num_classes depends on the choice of dataset.

3.3 Feature Reconstruction

Zero Padding

Perhaps one of the most straightforward ways to deal with missing data is to simply replace the missing values with the value 0, which is referred to as zero padding. We use zero padding

to serve as our baseline for our feature reconstruction methods detailed below. Intuitively, we expect other feature reconstruction methods to outperform, or at least match, zero padding because replacing missing feature values with 0 does not add any information that the classification MLP can use.

Replacement with Training Mean

As a potential improvement over zero padding, we replace missing audio feature values in our test data with the mean of the audio features of the training data and we replace missing vision feature values in our test data with the mean of the vision features of the training data.

Replacement with Random Values

As an alternative to zero padding, we replace missing feature values in our test data with random uniform values between 0 and 1. Intuitively, we expect this to perform worse than the previously mentioned approaches (zero padding and replacement with training mean) because missing modalities are replaced with random noise, which could be misleading compared to replacement with a constant value.

Gaussian Mixture Model

We use a Gaussian Mixture Model (GMM) with variational inference algorithms (Variational Bayesian Gaussian Mixture). The model infers an approximate posterior distribution over the parameters of a Gaussian mixture model. Variational inference is an extension of expectation-maximization that maximizes a lower bound on model evidence instead of data likelihood. A Variational Bayesian Gaussian Mixture model avoids singularities often found in expectation-maximization solutions.

We use scikit-learn’s implementation and specify the number of mixture components to be 2 and the covariance type to be diagonal so that each component has its own diagonal covariance matrix. We fit the model on our train dataset and then sample from it to reconstruct missing feature data in our test dataset.

We choose to use a GMM because it is an unsupervised approach that clusters data into multiple Gaussian distributions. GMMs are used when there is uncertainty about the number of clusters in the data and are useful for generating synthetic data points for data augmentation. In our case, it serves as a more advanced model than the replacement techniques described above.

Generative Adversarial Networks

Generative Adversarial Networks (GANs) are a type of generative model introduced by Goodfellow et al. [10]. GANs are inspired by the two-player zero-sum game and are trained

under the notion of adversarial learning. A GAN is comprised of a generator and a discriminator that are typically neural networks. The role of the generator is to generate new data samples that capture the potential distribution of real samples. The role of the discriminator is to classify the real samples as real and generated samples as fake. In essence, both networks are trained simultaneously and the generator tries to fool the discriminator.

The optimization process of GANs is a minimax game process and the goal is for the generator to capture the distribution of real data samples. Typically, GANs are used to generate realistic images, address the problem of insufficient training examples for supervised learning, and they are even applied to speech and language processing.

GANs are used for modality generation (image generation, voice generation, etc.) and are a very popular generative approach. We choose to use them because of their popularity for generative tasks and because other similar missing modality works also use them.

In our case, we choose to implement a standard GAN and conditional GAN, where we generate one modality by conditioning on the other modality. In this sense, if we have the audio modality but are missing the visual modality, then we will generate the visual modality by sampling from a GAN conditioning on the audio modality. The process of conditioning is done by feeding the conditioned variable c into the generator and discriminator with the input data x . x and c are concatenated and then used as input to the generator and discriminator networks. When sampling from the conditional GAN, we concatenate c with random noise z to feed into the generator. The generator output is then returned.

We train a total of 3 different GANs for each dataset and seed:

1. AV-GAN (Standard GAN): in the case where both the visual and audio modality are missing, sample from AV-GAN to generate both.
2. V-GAN (Conditional GAN): in the case where the visual modality is missing, sample from V-GAN and use the audio modality as the conditional variable to generate the visual modality.
3. A-GAN (Conditional GAN): in the case where the audio modality is missing, sample from A-GAN and use the visual modality as the conditional variable to generate the audio modality.

Refer to 3.3 for the architecture for the GANs we use. We train our GANs for 200 epochs using binary cross-entropy loss and the Adam optimizer with a learning rate of $2e-4$.

Variational Autoencoders

Variational Autoencoders (VAEs) are a type of generative model introduced by [15] that are used for learning complex data distributions. VAEs are composed of an encoder and a decoder that parametrize the variational approximate posterior and the conditional data distributions in a latent variable generative model [16]. Typically, both the encoder and decoder are neural networks.

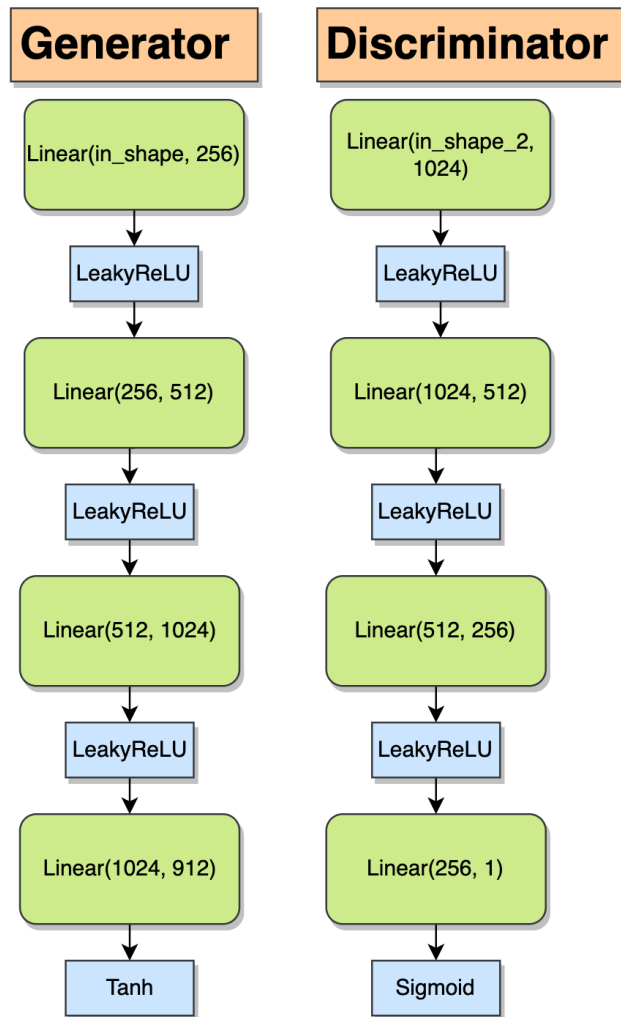


Figure 3.3: GAN architecture. For AV-GAN: $\text{in_shape} = 100$, $\text{in_shape_2} = 912$. For V-GAN: $\text{in_shape} = 612$, $\text{in_shape_2} = 1424$. For A-GAN: $\text{in_shape} = 500$, $\text{in_shape_2} = 1312$.

VAEs are deep Bayesian generative models that rely on the principles of amortized variational inference to approximate the complex distributions from which the observed data originate. The ground-truth distribution $p(x)$ is modeled by a parametric distribution $p_\theta(x)$ with a latent variable generative process that the encoder learns as seen below [16]. z is the latent variable.

$$p_\theta(x) = \int p_\theta(x|z)p(z)dz \quad (3.1)$$

Like GANs, VAEs are typically used for modality generation and are popular deep generative models. We choose to use them because of their popularity for generative tasks and

because other similar missing modality works also use autoencoders.

In our case, we choose to implement a standard VAE and conditional VAE, where we generate one modality by conditioning on the other modality. In this sense, if we have the audio modality but are missing the visual modality, then we will generate the visual modality by sampling from a VAE conditioning on the audio modality. The process of conditioning is done by feeding the conditioned variable c into the decoder along with the Gaussian random noise z . z and c are concatenated and then used as input to the decoder network. When sampling from the conditional VAE, we concatenate c with random noise z to feed into the decoder. The decoder output is then returned.

We train a total of 3 different VAEs for each dataset and seed:

1. AV-VAE (Standard VAE): in the case where both the visual and audio modality are missing, sample from AV-VAE to generate both.
2. V-VAE (Conditional VAE): in the case where the visual modality is missing, sample from V-VAE and use the audio modality as the conditional variable to generate the visual modality.
3. A-VAE (Conditional VAE): in the case where the audio modality is missing, sample from A-VAE and use the visual modality as the conditional variable to generate the audio modality.

Refer to 3.4 for the architecture for the VAEs we use. We train our VAEs for 2000 epochs using a loss function composed of KL divergence and Gaussian likelihood. We use the Adam optimizer with a learning rate of $1e-4$.

3.4 Oracle Models

The motivation behind implementing oracle models, that is models that assume complete knowledge of the test data, is to establish an upper bound and lower bound for the aforementioned feature reconstruction methods. Beyond this, the oracle models also serve to facilitate an understanding of how feature reconstruction works across different possible scenarios: replacement with features from the same class, replacement with features from a different class, and replacement with the class-conditioned mean. The results from using these oracle models have implications on the total maximum and minimum performance we can expect from feature reconstruction on our datasets.

Oracle: Sample From Same Class

To establish an upper bound for feature reconstruction, we create a method that assumes complete knowledge of the test data. Assume a data sample d with label l has missing feature data. This oracle model replaces the missing data with another data sample d' with

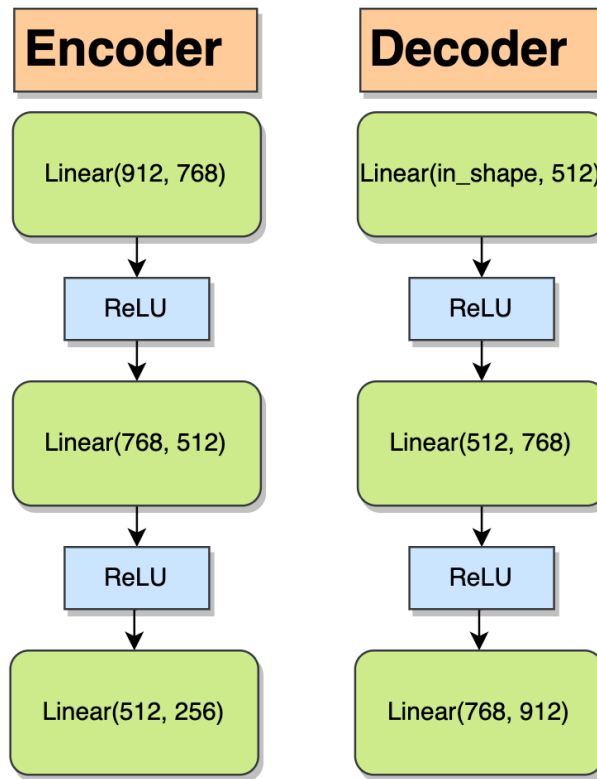


Figure 3.4: VAE architecture. For AV-VAE: $\text{in_shape} = 256$. For V-VAE: $\text{in_shape} = 768$. For A-VAE: $\text{in_shape} = 656$.

the same label l . The sample d' is selected at random and both d and d' are in the test dataset.

Oracle: Sample From Different Class

As another guideline, we create a method that assumes complete knowledge of the test data but behaves differently from the previous method by serving as a lower bound. Assume a data sample d with label l has missing feature data. This oracle model replaces the missing data with another data sample d' with some label l' where $l' \neq l$. The sample d' is selected at random and both d and d' are in the test dataset.

Oracle: Class-Conditioned Means

As an improvement over the method Replacement with Training Mean, we create another method that assumes complete knowledge of the test data. For each class (label), the means of the vision and audio features are computed. Let $\mu_{\text{vision}}(l)$ be the mean of the vision

features with label l from the test data and $\mu_{audio}(l)$ be the mean of the audio features with label l from the test data. Assume a data sample d with label l has missing feature data. This oracle model replaces the missing vision feature data with the value $\mu_{vision}(l)$ and the missing audio feature data with value $\mu_{audio}(l)$.

Chapter 4

Experimental Design

4.1 Datasets

RAVDESS

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) contains 356 files (total size: 24.8 GB). The database contains 24 professional actors (12 female, 12 male), vocalizing two lexically-matched statements in a neutral North American accent. Speech includes calm, happy, sad, angry, fearful, surprised, and disgusted expressions, and song contains calm, happy, sad, angry, and fearful emotions. Each expression is produced at two levels of emotional intensity (normal and strong), with an additional neutral expression. All conditions are available in three modality formats: Audio-only (16bit, 48kHz .wav), Audio-Video (720p H.264, AAC 48kHz, .mp4), and Video-only (no sound) [18].

We choose to use the audio-video files (visual and audio modalities) with the task of classifying emotion from 8 categories: calm, happy, sad, angry, fearful, surprise, and disgust.

eNTERFACE'05

The final version of the database contains 42 subjects, coming from 14 different nationalities. Among the 42 subjects, 81% were men, while the remaining 19% were women. 31% of the total set wore glasses, while 17% of the subjects had a beard. Each subject was told to listen to six successive short stories, each of them eliciting one of the following emotions: happiness, sadness, surprise, anger, disgust, and fear. They had then to react to each of the situations and two human experts judged whether the reaction unambiguously expressed the emotion. If this was the case, the sample was added to the database. If not, it was discarded [22].

We used the video files (visual and audio modalities) with the task of classifying emotion from 6 categories: happiness, sadness, surprise, anger, disgust, and fear.

CMU-MOSI

The Multimodal Opinion-level Sentiment Intensity dataset (CMU-MOSI) is a standard benchmark for multimodal sentiment analysis. It is a collection of 2199 opinion video clips, with each video being annotated with a sentiment in the range $[-3, 3]$. The dataset is rigorously annotated with labels for subjectivity, sentiment intensity, per-frame, and per-opinion annotated visual features, and per-milliseconds annotated audio features [36].

While the dataset contains visual, audio, and text modalities, we choose to only use the visual and audio modalities. We address the task of classifying emotion from 7 classes (-3, -2, -1, 0, 1, 2, 3) and the simpler task of classifying samples as one of two classes: positive (≥ 0) or negative (< 0).

4.2 Data Processing

The raw multimodal data are video files (.avi or .mp4 files depending on the dataset). The following sections detail how the multimodal features (concatenation of vision and audio features) are created.

Vision Features

To extract vision features, we use the SlowFast model and follow the PyTorch documentation for how to load the video and define the input transform (which is model-specific). A clip from the beginning of the video is loaded (clip duration is defined in PyTorch documentation). That clip is then transformed and then fed to the SlowFast model to get a feature vector of length 400. This process is repeated for every video in the dataset and these features are saved.

Audio Features

To extract audio features, we use Wav2Vec2.0. We use librosa to load in the video with a sampling rate of 16000 and convert the signal to mono to get an audio time series. We feed this into the Wav2Vec2.0 processor and then feed the processor's output into the Wav2Vec2.0 model. We take the mean of the model's output across dimension 1 to get a feature vector of length 512. This process is repeated for every video in the dataset and these features are saved.

4.3 Experiment Design

To split our datasets into a train and test set, we assign 20% of the videos to be in our test set and the remaining 80% to make up our train set.

To simulate missing modality during test time, we will drop data from the test set of each of our datasets. For each data sample, modality $m \in \{vision, audio\}$ is dropped with probability p_m . We drop a modality by dropping its entire part of the feature representation. For example, if we drop the vision modality, then we remove the 400 elements corresponding to the vision features from the complete multimodal feature vector. We run experiments with different values of p_m for $m \in \{vision, audio\}$. We evaluate the feature reconstruction methods detailed in Chapter 3 by evaluating the trained MLP on the test set.

For the feature reconstruction methods using GANs and VAEs, we follow a specific process for replacing the missing modality. Depending on the modality/modalities missing, we sample from one of our GANs/VAEs to replace the modality. We do this sampling process 20 times and use our classification MLP to predict each sample. We then take the max vote of the 20 predictions and have that be our predicted label. Refer to Algorithm 1 for the pseudocode of this sampling process.

Algorithm 1 GAN/VAE Sampling Algorithm

```

for inputs  $\in$  dataset do
   $mask_{audio} \leftarrow \text{random}() < p_a$   $\triangleright p_a$  is the probability of masking the audio
   $mask_{vision} \leftarrow \text{random}() < p_v$   $\triangleright p_v$  is the probability of masking the vision
  for  $i \in [0, 20)$  do
    if  $mask_{audio}$  and  $mask_{video}$  then
      inputs  $\leftarrow$  AV_Model.sample()
    end if
    if  $mask_{audio}$  and not  $mask_{video}$  then
      inputs  $\leftarrow$  A_Model.sample(vision_features)
    end if
    if not  $mask_{audio}$  and  $mask_{video}$  then
      inputs  $\leftarrow$  V_Model.sample(audio_features)
    end if
    predictions.append(mlp(inputs))
  end for
  if max_vote(predictions) == correct_label then
    running_corrects  $\leftarrow$  running_corrects + 1
  end if
end for

```

When running experiments, we run each experiment with 5 different seeds: 42, 1, 2, 3, 4. In our results in Chapter 5, we report accuracy with a 95% confidence interval.

Chapter 5

Results and Analysis

Before we begin, we will define a legend to explain the terms used in our results section. Refer to 5.1 for this legend.

5.1 Results

We run multiple experiments to measure the performance of each of our methods on our datasets. We vary the probability of dropping a modality for each test data sample and record the accuracy of our MLP (trained on complete training data) on the partially corrupted test data.

Results can be found in the following tables: 5.2, 5.3, 5.4, 5.5, 5.6.

Table 5.1: Legend

Term	Definition
p_a	Probability of dropping audio modality
p_v	Probability of dropping vision modality
ZP	Zero Padding
MR	Replacement with Training Mean
Random	Replacement with Random Uniform Values
GMM	Gaussian Mixture Model
GAN	Generative Adversarial Network
VAE	Variational Autoencoder

Table 5.2: Baseline Accuracies for Trained MLP

Dataset	Baseline Accuracy With Complete Test Data
RAVDESS	91.85 ± 0.50
eNTERFACE'05	77.07 ± 1.43
CMU-MOSI (7 classes)	29.05 ± 1.24
CMU-MOSI (2 classes)	67.05 ± 1.24

Table 5.3: RAVDESS Feature Reconstruction Method Accuracies

p_a	p_v	ZP	MR	Random	GMM	GAN	VAE
0.25	0.25	73.52 ± 1.98	73.73 ± 1.82	70.22 ± 1.67	73.28 ± 1.19	73.32 ± 1.54	79.14 ± 1.42
0.50	0.50	53.93 ± 3.71	53.85 ± 3.75	49.61 ± 1.89	53.52 ± 1.77	54.83 ± 3.52	62.28 ± 1.31
0.75	0.75	32.46 ± 4.26	32.59 ± 4.05	31.20 ± 1.86	34.42 ± 1.58	33.52 ± 3.05	40.45 ± 1.32
1.00	1.00	11.00 ± 3.67	11.00 ± 3.67	14.38 ± 0.80	14.18 ± 1.25	11.00 ± 3.67	14.99 ± 2.02

Table 5.4: RAVDESS Oracle Methods

p_a	p_v	Sample From Same Class	Sample From Different Class	Class-Conditioned Means
0.25	0.25	89.08 ± 0.67	68.39 ± 1.70	93.97 ± 0.14
0.50	0.50	85.17 ± 1.43	47.05 ± 1.81	95.52 ± 0.48
0.75	0.75	85.42 ± 1.60	28.92 ± 1.40	97.47 ± 0.35
1.00	1.00	83.87 ± 1.24	13.60 ± 0.26	100.00 ± 0.00

Table 5.5: eNTERFACE Feature Reconstruction Method Accuracies

p_a	p_v	ZP	MR	Random	GMM	GAN	VAE
0.25	0.25	62.01 ± 2.59	61.70 ± 2.62	59.07 ± 2.83	61.78 ± 1.88	61.39 ± 2.11	65.71 ± 2.18
0.50	0.50	45.71 ± 2.46	45.25 ± 1.98	40.93 ± 2.01	45.95 ± 1.70	45.41 ± 2.84	49.58 ± 2.15
0.75	0.75	29.65 ± 2.78	29.58 ± 2.56	28.96 ± 2.77	29.58 ± 1.90	30.58 ± 3.28	32.59 ± 3.09
1.00	1.00	17.99 ± 3.08	17.99 ± 3.08	16.06 ± 4.21	17.76 ± 2.13	17.99 ± 3.08	15.44 ± 1.71

Table 5.6: eNTERFACE Oracle Methods

p_a	p_v	Sample From Same Class	Sample From Different Class	Class-Conditioned Means
0.25	0.25	74.98 ± 2.83	59.77 ± 1.49	81.00 ± 1.46
0.50	0.50	70.27 ± 1.94	43.17 ± 2.85	86.33 ± 0.70
0.75	0.75	68.26 ± 2.90	29.34 ± 2.24	91.58 ± 1.16
1.00	1.00	67.41 ± 2.92	15.91 ± 1.34	100.00 ± 0.00

Table 5.7: CMU-MOSI (7 Classes) Feature Reconstruction Method Accuracies

p_a	p_v	ZP	MR	Random	GMM	GAN	VAE
0.25	0.25	25.27 \pm 1.43	25.36 \pm 1.44	25.86 \pm 1.00	25.23 \pm 0.71	23.95 \pm 1.10	26.73 \pm 1.31
0.50	0.50	21.73 \pm 2.42	21.82 \pm 2.67	21.00 \pm 0.48	23.14 \pm 0.89	19.73 \pm 2.11	25.00 \pm 1.23
0.75	0.75	17.59 \pm 4.39	17.86 \pm 4.79	19.23 \pm 1.90	19.86 \pm 0.61	16.68 \pm 3.42	20.14 \pm 1.57
1.00	1.00	12.91 \pm 1.43	12.91 \pm 6.69	17.05 \pm 1.51	18.50 \pm 1.60	12.91 \pm 6.69	18.05 \pm 1.17

Table 5.8: CMU-MOSI (7 Classes) Oracle Methods

p_a	p_v	Sample From Same Class	Sample From Different Class	Class-Conditioned Means
0.25	0.25	26.91 \pm 1.10	24.59 \pm 1.27	28.59 \pm 1.99
0.50	0.50	26.23 \pm 1.84	21.27 \pm 1.40	31.41 \pm 3.44
0.75	0.75	23.77 \pm 0.53	19.23 \pm 1.81	36.05 \pm 6.54
1.00	1.00	22.91 \pm 2.44	18.77 \pm 2.80	41.68 \pm 11.54

Table 5.9: CMU-MOSI (2 Classes) Feature Reconstruction Method Accuracies

p_a	p_v	ZP	MR	Random	GMM	GAN	VAE
0.25	0.25	59.73 \pm 1.58	59.41 \pm 1.38	61.27 \pm 1.05	62.36 \pm 1.09	62.14 \pm 1.74	64.68 \pm 0.96
0.50	0.50	55.91 \pm 2.38	55.77 \pm 2.44	53.27 \pm 1.23	57.64 \pm 1.85	55.14 \pm 1.08	64.59 \pm 1.34
0.75	0.75	51.32 \pm 1.34	51.32 \pm 1.39	49.45 \pm 1.86	53.05 \pm 2.08	51.09 \pm 2.42	56.82 \pm 1.98
1.00	1.00	47.68 \pm 3.78	47.68 \pm 3.78	46.05 \pm 0.99	50.64 \pm 0.99	47.68 \pm 3.78	50.55 \pm 1.67

Table 5.10: CMU-MOSI (2 Classes) Oracle Methods

p_a	p_v	Sample From Same Class	Sample From Different Class	Class-Conditioned Means
0.25	0.25	64.32 \pm 1.86	62.55 \pm 2.10	63.09 \pm 1.33
0.50	0.50	62.05 \pm 0.87	57.41 \pm 0.69	60.05 \pm 1.99
0.75	0.75	63.68 \pm 2.77	53.64 \pm 1.30	54.27 \pm 1.95
1.00	1.00	62.77 \pm 3.03	50.0 \pm 1.83	47.68 \pm 3.78

5.2 Analysis

Baseline Accuracies

Before analyzing our feature reconstruction methods, we will briefly discuss our baseline accuracies with complete test data and compare our approach with other work.

RAVDESS

We achieve state-of-the-art performance on the RAVDESS dataset. We report an accuracy of 91.85%. Middya et al. use separate feature extractor networks for audio and video data, and then fuse these features in a multimodal model to achieve 86% accuracy [23]. 81%. [30, 2] only use audio and an MLP to achieve 81% accuracy and 85% accuracy respectively.

eNTERFACE'05

We achieve close to state-of-the-art performance on the eNTERFACE'05 dataset. We report an accuracy of 77.07%. Tiwari et al. use S-DLA to extract features and feed those into a 1-D CNN, resulting in an accuracy of 86.41% [33]. Zhi et al. use three attention modules inserted into a backbone network to achieve 88.33% [39].

CMU-MOSI

Unfortunately, our feature extraction and emotion classification architecture does not perform as well on the CMU-MOSI dataset compared to existing work. We report an Acc-7 of 29.05% and an Acc-2 of 67.05%. Song et al. use ALBERT and PANNs on the raw text and audio respectively to create multimodal features and achieve an accuracy of 84.98% in the 2 class scenario [29]. Zhao et al. use MAG+ (multimodal adaptation gate attached to BERT and XLNet) to get 87.60% in the 2 class scenario [38]. Miyazawa et al. use pre-trained Transformer models to achieve an Acc-7 of 56.27% and an Acc-2 of 86.89% [24].

Dataset Trends for Feature Reconstruction

Across all three datasets (RAVDESS, eNTERFACE, and CMU-MOSI), the same general trends hold with regard to the feature reconstruction methods. The only noticeable difference across datasets is that for CMU-MOSI with 2 classes, the results of the Oracle methods differ.

In terms of the feature reconstruction methods, the only method that offers a significant improvement in performance after feature reconstruction for all datasets is the VAE. For the dataset RAVDESS, across different values of p_a and p_v , the VAE achieved performance gains over the other methods of between 6 to 12%. For eNTERFACE, the VAE achieved performance gains over other methods of 3 to 4%. The oracle models performed identically across datasets. For CMU-MOSI (7 Classes), the VAE achieved performance gains over other

methods of 2 to 4%. For CMU-MOSI (2 Classes), the VAE achieved performance gains over other methods of 5 to 9%.

For CMU-MOSI specifically, the GMM and GANs achieved small performance gains over zero padding, mean replacement, and random replacement for different masking probabilities. These performance gains ranged from 2 to 3%.

Replacement with 0, training mean, random values

These methods are simple approaches for feature reconstruction that are good baselines for other more complicated approaches. Across all datasets, replacing the missing modality with 0 results in the same performance as replacing the missing modality with the training mean. This makes intuitive sense as our trained MLP contains a batch normalization layer, which standardizes layer inputs by keeping track of the mean and standard deviation of input variables. Therefore, replacing missing modalities with a constant value results in the same performance regardless of the value since inputs are standardized. Replacing the missing modality with uniform random values results in worse performance across different values of p_a and p_v , around a 3-5% drop compared to replacement with 0 or the training mean for RAVDESS and eNTERFACE. Replacement with random values achieves comparable performance to zero padding and mean replacement for CMU-MOSI, perhaps indicating that our multimodal feature extraction architecture does not perform as well on this dataset.

GMM

For RAVDESS and eNTERFACE, sampling from a GMM results in similar performance to replacement with 0 and the training mean. This is likely because the GMM learns a normal distribution with a mean close to the modality-specific training means. So sampling from this distribution should result in values close to the training means and therefore similar performance to replacement with training mean.

In the case of CMU-MOSI (7 Classes), the GMM achieves a 2% improvement for $p_a, p_v = 0.50$ and $p_a, p_v = 0.75$. For CMU-MOSI (2 Classes), the GMM achieves a 2 to 3% improvement across all values of p_a and p_v .

VAE

Interestingly enough, the only method with a significant improvement over replacing the missing modality with 0's was using VAEs. This is perhaps due to the VAE architecture being able to better learn representations of our feature data. If the quality of latent representation of our multimodal data is good, then it makes sense that sampling the VAEs results in modality representations that are more accurate than the other methods.

VAE vs. GAN

While using the VAEs for feature reconstruction resulted in a performance improvement, the GANs were unable to replicate this. This could be because for our use case, the multimodal data we have is more suited to the Gaussian distribution that the VAE’s encoder learns. We hypothesize that the VAEs are easier to train, hence why our VAEs can learn how to generate missing modalities better than the GANs. In order to achieve a similar performance boost, we would have to tune the GANs and potentially modify our training process, which is more difficult.

Oracle Models

For the model Oracle: Class-Conditioned Means, performance on RAVDESS and eINTERFACE improves as p_a and p_v increase. This is the only method for which this phenomenon occurs. What this means is that replacing missing modalities with their class-conditioned means performs very well. This approach removes any sampling variance and as a result the trained MLP successfully classifies a majority of the test data (achieving 100% accuracy in the case where all modalities are dropped). Therefore, we can say that the MLP has trouble with variance in the feature data.

On CMU-MOSI (7 Classes), Oracle: Class-Conditioned Means achieves better performance as p_a and p_v increase. However, on CMU-MOSI (2 Classes), the class-conditioned means model has worse performance as p_a and p_v increase. Furthermore, the model Oracle: Sample From Same Class outperforms Oracle: Class-Conditioned Means across all values of p_a, p_v . The reasons for this are not obvious but we hypothesize that in the case of having only 2 classes, the class-conditioned means are too similar and not distinct enough for the MLP to distinguish them.

For the dataset RAVDESS, our feature reconstruction methods all outperform Oracle: Sample From Different Class, which is our theoretical minimum performance. The worst method is replacement with random uniform values, which still outperforms Oracle: Sample From Different Class. Intuitively, we expect a method that adds no new information (or simply noise) to still outperform a method that adds misleading information (the case for this specific Oracle model). The feature reconstruction methods all achieve worse performance than Oracle: Sample From Same Class, which is our theoretical maximum performance. We expect none of our methods to reach this Oracle model’s performance because it assumes knowing the labels of the test data. However, it does tell us that while the best feature reconstruction methods cannot exactly match the original MLP accuracy, they can get within a reasonable delta. This is an incentive to continue working on the problem of missing modality reconstruction since theoretically, it can result in close to the original performance.

For the dataset eINTERFACE, almost all of our feature reconstruction methods outperform Oracle: Sample From Different Class. Surprisingly, replacing with random uniform values performs worse than sampling from a different class. However, unlike in RAVDESS, the difference in accuracy between Oracle: Sample From Different Class and the feature

reconstruction methods is smaller. While the performance improvement is smaller, it is encouraging that the methods we present are still above the theoretical minimum bound of performance. This decreased performance delta could be dataset-dependent, as perhaps it is tougher for our generative models to learn representations for the eNTERFACE data. Hence why using a GMM and GANs performs worse than zero padding.

For the dataset CMU-MOSI (7 Classes), our feature reconstruction methods generally fall between Oracle: Sample From Different Class and Oracle: Sample From Same Class. However, perhaps because the baseline accuracy on this dataset is low, the difference in performance between the two theoretical bounds is small. Some methods are below our theoretical lower bound for certain probability values.

For the dataset CMU-MOSI (2 Classes), our feature reconstruction methods generally fall below our theoretical lower bound. Every method except for the VAEs is slightly below Oracle: Sample From Different Class. Surprisingly, the VAEs outperform our theoretical upper bound for $p_a, p_v = 0.25$ and $p_a, p_v = 0.50$. These trends differ from our results on RAVDESS and eNTERFACE. We expect that this is because our feature extraction pipeline does not result in accurate features for CMU-MOSI, hence why we get atypical performance. However, even on this dataset, we show the success of our VAEs in handling missing modality at test time as we outperform the other feature reconstruction methods and even the theoretical upper bound.

Chapter 6

Future Work

While we cover multiple multimodal datasets and generative methods, there are still some intriguing extensions that can be explored. This work is solely concerned with multimodal emotion recognition datasets, but it would be interesting to see the addition of more datasets and compare the results. A dataset like Kinetics [14] would be an ideal choice because it is a human action video dataset. Videos are multimodal in nature, containing vision and sound modalities. Kinetics-400 contains 400 human action classes, and it would be interesting seeing if the feature reconstruction methods presented in this work perform well for other dataset choices.

Additionally, we focus on missing modality strictly during test time. Other work has investigated training models with less data by removing portions of modalities in the training data. Removing portions of the data at train time should result in a more robust model that can handle missing modality during test time more effectively. Training with less data, for example dropping 50% of the input data, and then running the same experiments described in this work to see how that affects results would be a logical next step.

One limitation of this current work is that our feature extraction process is zero shot, meaning that neither our vision nor audio feature extraction networks are trained on any of the multimodal data we use from RAVDESS, eNTERFACE, or CMU-MOSI. While we still present close to state-of-the-art results despite this, fine-tuning our feature extraction networks would improve our MLP baseline accuracy. A second limitation is that the performance of our GANs/VAEs could likely be improved further with more tuning. While the VAEs achieve the best performance for feature reconstruction, the GANs are essentially equivalent to zero padding. Intuitively, with more tuning, we would expect our GAN method to outperform other methods such as zero padding, replacement with training mean, replacement with random values, and sampling from a GMM.

Chapter 7

Conclusion

In this work, we examined missing data at test time in the multimodal setting. We investigate multimodal datasets with the task of emotion recognition: RAVDESS, eNTERFACE'05, and CMU-MOSI. Real-world AI systems are dependent on taking in multiple streams of data that often come from different modalities. In systems being used for tasks that are necessary to be performed correctly, such as autonomous vehicles, a data collection device failing should not drastically impact the system. We examine several feature reconstruction methods to deal with losing modalities with varying probabilities. As we show, VAEs are able to minimize the loss in performance from losing modality data. Additionally, we present state-of-the-art performance on RAVDESS and close to state-of-the-art performance on eNTERFACE. Our novel classification architecture (feature extraction networks and MLP) outperforms existing work in the domain of multimodal emotion recognition. More work still needs to be done and pursuing the extensions from Chapter 6 is a logical next step. Handling loss of modality in multimodal deep learning is still very much an open problem that if solved can lead to the more widespread acceptance of multimodal AI systems.

Bibliography

- [1] Gustavo Aguilar et al. “Multimodal and Multi-view Models for Emotion Recognition”. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 991–1002. DOI: 10.18653/v1/P19-1095. URL: <https://aclanthology.org/P19-1095>.
- [2] Abeer Alnuaim et al. “Human-Computer Interaction for Recognizing Speech Emotions Using Multilayer Perceptron Classifier”. In: *Journal of Healthcare Engineering 2022* (Mar. 2022), pp. 1–12. DOI: 10.1155/2022/6005446.
- [3] Alexei Baevski et al. “wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 12449–12460. URL: <https://proceedings.neurips.cc/paper/2020/file/92d1e1eb1cd6f9fba3227870bb6d7f07-Paper.pdf>.
- [4] Navaneeth Bodla, Gang Hua, and Rama Chellappa. “Semi-supervised FusedGAN for Conditional Image Generation”. In: *CoRR* abs/1801.05551 (2018). arXiv: 1801.05551. URL: <http://arxiv.org/abs/1801.05551>.
- [5] Lei Cai et al. “Deep Adversarial Learning for Multi-Modality Missing Data Completion”. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*. KDD '18. London, United Kingdom: Association for Computing Machinery, 2018, pp. 1158–1166. ISBN: 9781450355520. DOI: 10.1145/3219819.3219963. URL: <https://doi.org/10.1145/3219819.3219963>.
- [6] Chris Donahue, Julian J. McAuley, and Miller S. Puckette. “Synthesizing Audio with Generative Adversarial Networks”. In: *CoRR* abs/1802.04208 (2018). arXiv: 1802.04208. URL: <http://arxiv.org/abs/1802.04208>.
- [7] Changde Du et al. “Semi-supervised Deep Generative Modelling of Incomplete Multi-Modality Emotional Data”. In: *Proceedings of the 26th ACM international conference on Multimedia*. ACM, Oct. 2018. DOI: 10.1145/3240508.3240528. URL: <https://doi.org/10.1145/3240508.3240528>.
- [8] Christoph Feichtenhofer et al. “SlowFast Networks for Video Recognition”. In: *CoRR* abs/1812.03982 (2018). arXiv: 1812.03982. URL: <http://arxiv.org/abs/1812.03982>.

- [9] N Fragopanagos and J.G. Taylor. “Emotion recognition in human-computer interaction”. In: *Neural networks : the official journal of the International Neural Network Society* 18 (June 2005), pp. 389–405. DOI: 10.1016/j.neunet.2005.03.006.
- [10] Ian J. Goodfellow et al. *Generative Adversarial Networks*. 2014. DOI: 10.48550/ARXIV.1406.2661. URL: <https://arxiv.org/abs/1406.2661>.
- [11] Karol Gregor et al. “DRAW: A Recurrent Neural Network For Image Generation”. In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by Francis Bach and David Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, July 2015, pp. 1462–1471. URL: <https://proceedings.mlr.press/v37/gregor15.html>.
- [12] Jonas Grosman. *XLSR Wav2Vec2 English by Jonas Grosman*. <https://huggingface.co/jonatasgrosman/wav2vec2-large-xlsr-53-english>. 2021.
- [13] Jing Han et al. “Implicit Fusion by Joint Audiovisual Training for Emotion Recognition in Mono Modality”. In: *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2019), pp. 5861–5865.
- [14] Will Kay et al. *The Kinetics Human Action Video Dataset*. 2017. DOI: 10.48550/ARXIV.1705.06950. URL: <https://arxiv.org/abs/1705.06950>.
- [15] Diederik P Kingma and Max Welling. *Auto-Encoding Variational Bayes*. 2013. DOI: 10.48550/ARXIV.1312.6114. URL: <https://arxiv.org/abs/1312.6114>.
- [16] Frantzeska Lavda, Magda Gregorová, and Alexandros Kalousis. *Improving VAE generations of multimodal data through data-dependent conditional priors*. 2019. DOI: 10.48550/ARXIV.1911.10885. URL: <https://arxiv.org/abs/1911.10885>.
- [17] Yitong Li et al. “Video Generation From Text”. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 32.1 (Apr. 2018). URL: <https://ojs.aaai.org/index.php/AAAI/article/view/12233>.
- [18] Steven R. Livingstone and Frank A. Russo. “The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English”. In: *PLOS ONE* 13.5 (May 2018), pp. 1–35. DOI: 10.1371/journal.pone.0196391. URL: <https://doi.org/10.1371/journal.pone.0196391>.
- [19] Fei Ma, Shao-Lun Huang, and Lin Zhang. “An Efficient Approach for Audio-Visual Emotion Recognition With Missing Labels And Missing Modalities”. In: *2021 IEEE International Conference on Multimedia and Expo (ICME)*. 2021, pp. 1–6. DOI: 10.1109/ICME51207.2021.9428219.
- [20] Fei Ma et al. “Maximum Likelihood Estimation for Multimodal Learning with Missing Modality”. In: *CoRR* abs/2108.10513 (2021). arXiv: 2108.10513. URL: <https://arxiv.org/abs/2108.10513>.

- [21] Mengmeng Ma et al. “SMIL: Multimodal Learning with Severely Missing Modality”. In: *CoRR* abs/2103.05677 (2021). arXiv: 2103.05677. URL: <https://arxiv.org/abs/2103.05677>.
- [22] Olivier Martin et al. “The eNTERFACE’05 Audio-Visual Emotion Database.” In: *ICDE Workshops*. Ed. by Roger S. Barga and Xiaofang Zhou. IEEE Computer Society, 2006, p. 8. ISBN: 0-7695-2571-7. URL: <http://dblp.uni-trier.de/db/conf/icde/icdew2006.html#MartinKMP06>.
- [23] Asif Iqbal Middy, Baibhav Nag, and Sarbani Roy. “Deep learning based multimodal emotion recognition using model-level fusion of audio–visual modalities”. In: *Knowledge-Based Systems* 244 (2022), p. 108580. ISSN: 0950-7051. DOI: <https://doi.org/10.1016/j.knosys.2022.108580>. URL: <https://www.sciencedirect.com/science/article/pii/S0950705122002593>.
- [24] Kazuki Miyazawa, Yuta Kyuragi, and Takayuki Nagai. “Simple and Effective Multimodal Learning Based on Pre-Trained Transformer Models”. In: *IEEE Access* 10 (2022), pp. 29821–29833. DOI: 10.1109/ACCESS.2022.3159346.
- [25] Aaron van den Oord et al. “Conditional Image Generation with PixelCNN Decoders”. In: *Advances in Neural Information Processing Systems*. Ed. by D. Lee et al. Vol. 29. Curran Associates, Inc., 2016. URL: <https://proceedings.neurips.cc/paper/2016/file/b1301141feffabac455e1f90a7de2054-Paper.pdf>.
- [26] Srinivas Parthasarathy and Shiva Sundaram. *Training Strategies to Handle Missing Modalities for Audio-Visual Expression Recognition*. 2020. DOI: 10.48550/ARXIV.2010.00734. URL: <https://arxiv.org/abs/2010.00734>.
- [27] Hai Pham et al. *Found in Translation: Learning Robust Joint Representations by Cyclic Translations Between Modalities*. 2018. DOI: 10.48550/ARXIV.1812.07809. URL: <https://arxiv.org/abs/1812.07809>.
- [28] Yuge Shi et al. “Relating by Contrasting: A Data-efficient Framework for Multimodal Generative Models”. In: *CoRR* abs/2007.01179 (2020). arXiv: 2007.01179. URL: <https://arxiv.org/abs/2007.01179>.
- [29] Yunfeng Song et al. “Large Pretrained Models on Multimodal Sentiment Analysis”. In: *Artificial Intelligence in China*. Ed. by Qilian Liang et al. Singapore: Springer Singapore, 2022, pp. 506–513. ISBN: 978-981-16-9423-3.
- [30] G. Sowmya et al. “Speech2Emotion: Intensifying Emotion Detection Using MLP through RAVDESS Dataset”. In: *2022 International Conference on Electronics and Renewable Systems (ICEARS)*. 2022, pp. 1–3. DOI: 10.1109/ICEARS53579.2022.9752022.
- [31] Jabeen Summaira et al. “Recent Advances and Trends in Multimodal Deep Learning: A Review”. In: *CoRR* abs/2105.11087 (2021). arXiv: 2105.11087. URL: <https://arxiv.org/abs/2105.11087>.

- [32] Qiuling Suo et al. “Metric Learning on Healthcare Data with Incomplete Modalities”. In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, July 2019, pp. 3534–3540. DOI: 10.24963/ijcai.2019/490. URL: <https://doi.org/10.24963/ijcai.2019/490>.
- [33] Pradeep Tiwari and A. D. Darji. “A Novel S-LDA Features for Automatic Emotion Recognition from Speech using 1-D CNN”. English. In: *International Journal of Mathematical, Engineering and Management Sciences* 7.1 (2022). Copyright - © 2022. This work is published under <https://creativecommons.org/licenses/by/4.0/> (the “License”). Notwithstanding the ProQuest Terms and Conditions, you may use this content in accordance with the terms of the License; Last updated - 2022-04-04, pp. 49–67. URL: <https://www.proquest.com/scholarly-journals/novel-s-lda-features-automatic-emotion/docview/2631686819/se-2>.
- [34] Yao-Hung Hubert Tsai et al. “Learning Factorized Multimodal Representations”. In: *International Conference on Learning Representations*. 2019. URL: <https://openreview.net/forum?id=rygqqsA9KX>.
- [35] Zilong Wang, Zhaohong Wan, and Xiaojun Wan. “TransModality: An End2End Fusion Method with Transformer for Multimodal Sentiment Analysis”. In: *CoRR* abs/2009.02902 (2020). arXiv: 2009.02902. URL: <https://arxiv.org/abs/2009.02902>.
- [36] Amir Zadeh et al. “MOSI: Multimodal Corpus of Sentiment Intensity and Subjectivity Analysis in Online Opinion Videos”. In: *CoRR* abs/1606.06259 (2016). arXiv: 1606.06259. URL: <http://arxiv.org/abs/1606.06259>.
- [37] Jinming Zhao, Ruichen Li, and Qin Jin. “Missing modality imagination network for emotion recognition with uncertain missing modalities”. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 2021, pp. 2608–2618.
- [38] Xianbing Zhao et al. “MAG+: An Extended Multimodal Adaptation Gate for Multimodal Sentiment Analysis”. In: *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2022, pp. 4753–4757. DOI: 10.1109/ICASSP43922.2022.9746536.
- [39] Junnan Zhi et al. “Multi-Attention Module for Dynamic Facial Emotion Recognition”. In: *Information* 13.5 (2022). ISSN: 2078-2489. DOI: 10.3390/info13050207. URL: <https://www.mdpi.com/2078-2489/13/5/207>.
- [40] Yipin Zhou et al. “Visual to Sound: Generating Natural Sound for Videos in the Wild”. In: *CoRR* abs/1712.01393 (2017). arXiv: 1712.01393. URL: <http://arxiv.org/abs/1712.01393>.

- [41] Hao Zhu et al. “Deep Audio-visual Learning: A Survey”. In: *International Journal of Automation and Computing* 18.3 (2021), pp. 351–376. DOI: 10.1007/s11633-021-1293-0. URL: <https://doi.org/10.1007/s11633-021-1293-0>.