Deep Multimodal Representation Learning from Temporal Data

Xitong Yang*[1], Palghat Ramesh[2], Radha Chitta*[3], Sriganesh Madhvanath*[3], Edgar A. Bernal*[4] and Jiebo Luo[5]

[1]University of Maryland, College Park [2]PARC [3]Conduent Labs US
[4]United Technologies Research Center [5]University of Rochester

[1]xyang35@cs.umd.edu, [2]Palghat.Ramesh@parc.com, [3]{Radha.Chitta, Sriganesh.Madhvanath}@conduent.com, [4]bernalea@utrc.utc.com, [5]jluo@cs.rochester.edu

(Paper to appear in CVPR 2017)
https://arxiv.org/abs/1704.03152
Presented by Dustin Doss

Introduction

Problem

- Temporal Multimodal Learning (TML)
- Previous attempts:
  - Non-temporal models applied to concatenated data (deep autoencoders, etc.)
  - More recently, temporal models (Recurrent RBMs, multimodal LSTMs)
- Goals of a TML Model:
  - Joint representation for multimodal input and temporal structure
  - Dynamic weighting of input modalities
  - Generalize to different multimodal datasets
  - Efficient/tractable training

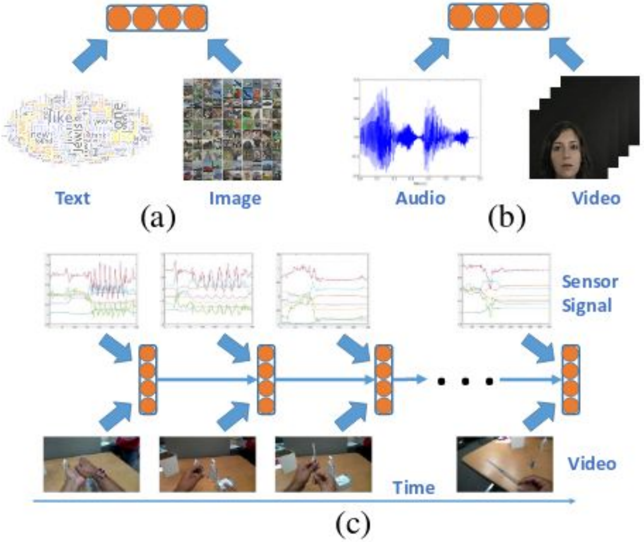


Figure 1. Different multimodal learning tasks. (a) Non-temporal model for non-temporal data [21]. (b) Non-temporal model for temporal data [13]. (c) Proposed CorrRNN model: temporal model for temporal data.

Introduction
Main Contributions/Claims

- Correlational Recurrent Neural Network (CorrRNN) uses assumption of correlation between modalities
    - Encoder/Decoder RNN framework with multimodal GRUs
    - Multi-aspect learning objective
    - Dynamic weighting of modes
- Improvements over state-of-the-art for video/sensor activity classification and audio-visual speech recognition
- More efficient training than previous TML models

CorrRNN Model
Overview

1. Input vectors mapped to hidden layers
2. Multimodal hidden inputs combined into fusion layer via multimodal GRU
   - Correlation and mode weighting used here
3. Final feature vector fed to decoder layer
4. Fairly standard reconstruction loss used here, re-extracting original inputs

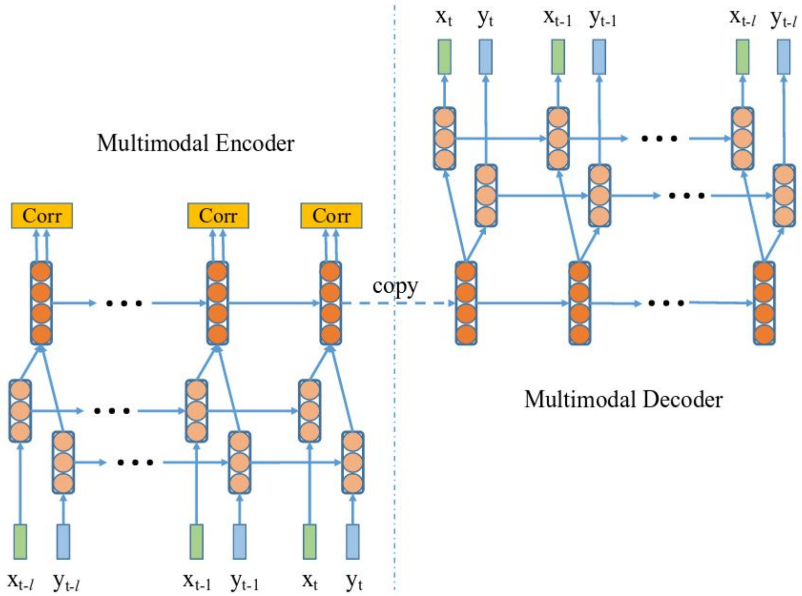


Figure 2. Basic architecture of the proposed model

CorrRNN Model

Encoder Overview

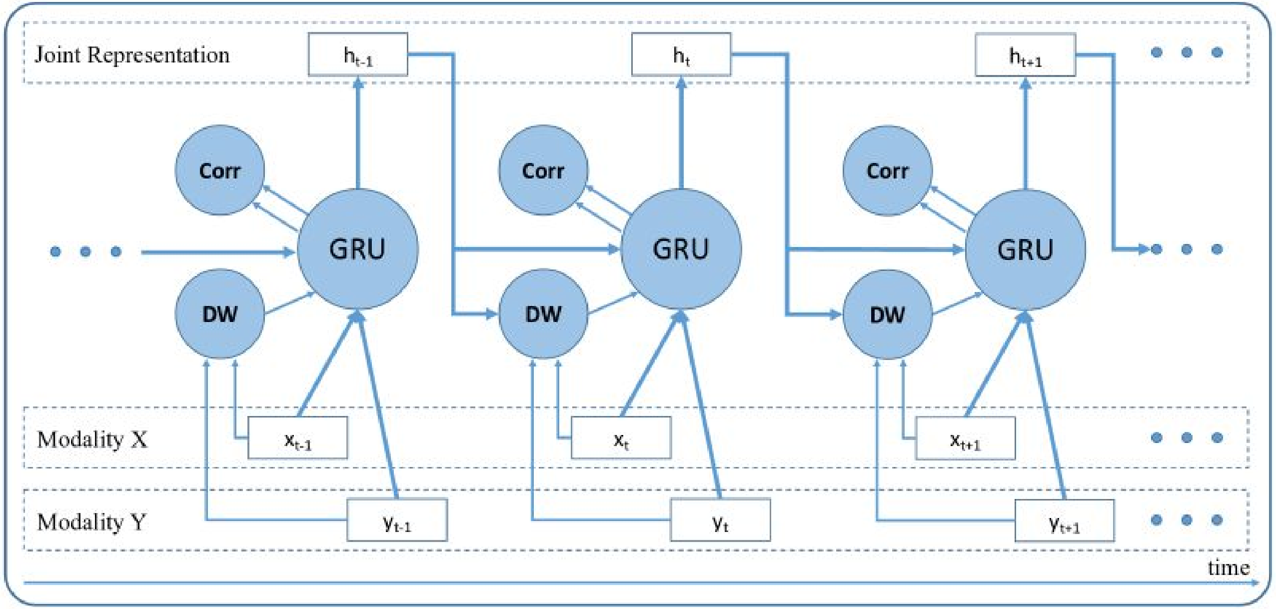


Figure 3. The structure of the multimodal encoder. It includes three modules: Dynamic Weighting module (DW), GRU module (GRU) and Correlation module (Corr).

CorrRNN Model

Encoder Dynamic Weighting Module

- “Soft-attention” mechanism to shift focus on most useful modality
- Based on coherence scores between time-steps of modalities:

$$\alpha_t^1 = x_t A_1 h_{t-1}^T, \quad \alpha_t^2 = y_t A_2 h_{t-1}^T,$$

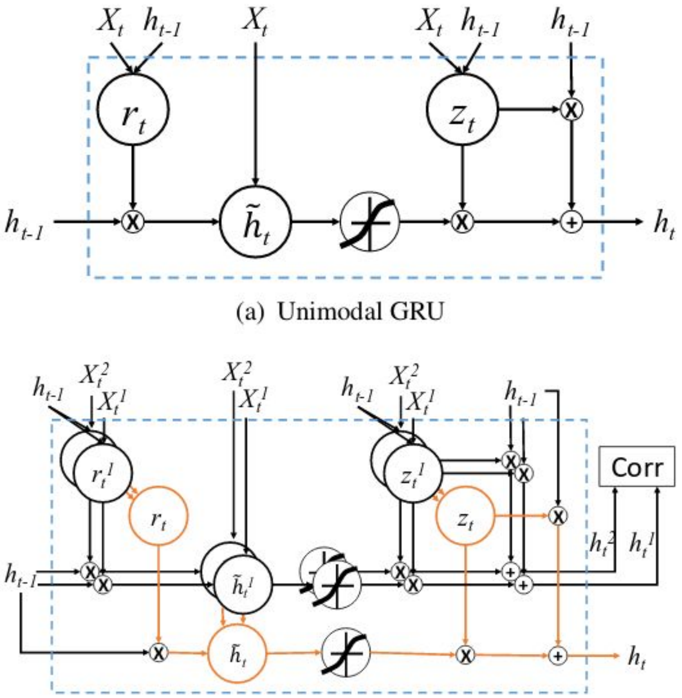
- Normalized using Laplace smoothing

$$w_t^i = \frac{1 + \exp(\alpha_t^i)}{2 + \sum_k \exp(\alpha_t^k)}, \quad i = 1, 2$$

CorrRNN Model

GRU module

- Multimodal GRU extends standard GRU
- Keeps track of 3 quantities:
  - Fused representation h_t
  - Individual representations h^1_t and h^2_t
- Uses different weights for different modalities



(a) Unimodal GRU

(b) Multimodal GRU

Figure 4. Block diagram illustrations of unimodal and multimodal GRU modules.

CorrRNN Model

Correlation module

- Compute correlation loss across individual representations from GRU

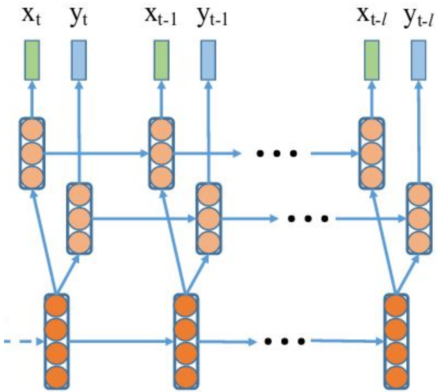
$$corr(H_t^1, H_t^2) = \frac{\sum_{i=1}^{N}(h_{ti}^1 - \overline{H_t^1})(h_{ti}^2 - \overline{H_t^2})}{\sqrt{\sum_{i=1}^{N}(h_{ti}^1 - \overline{H_t^1})^2 \sum_{i=1}^{N}(h_{ti}^2 - \overline{H_t^2})^2}}$$

where $\overline{H_t^1} = \frac{1}{N}\sum_i^N h_{ti}^1$ and $\overline{H_t^2} = \frac{1}{N}\sum_i^N h_{ti}^2$.

- Maximize correlation as part of feature learning

CorrRNN Model
Decoder

- Attempt to reconstruct individual modality sequences X and Y from h_t
- Uses three component loss terms



Multimodal Decoder

- **Fused-reconstruction loss.** The error in reconstructing $\tilde{x_i}$ and $\tilde{y_i}$ from joint representation $\tilde{h}_i = f(\tilde{x_i}, \tilde{y_i})$.

$$L_{\text{fused}} = L(g(f(\tilde{x_i}, \tilde{y_i})), \tilde{x_i}) + \beta L(g(f(\tilde{x_i}, \tilde{y_i}), \tilde{y_i})$$

- **Self-reconstruction loss.** The error in reconstructing $\tilde{x_i}$ from $\tilde{x_i}$, and $\tilde{y_i}$ from $\tilde{y_i}$.

$$L_{\text{self}} = L(g(f(\tilde{x_i})), \tilde{x_i}) + \beta L(g(f(\tilde{y_i}), \tilde{y_i})$$

- **Cross-reconstruction loss.** The error in reconstructing $\tilde{x_i}$ from $\tilde{y_i}$, and $\tilde{y_i}$ from $\tilde{x_i}$.

$$L_{\text{cross}} = L(g(f(\tilde{y_i}), \tilde{x_i}) + \beta L(g(f(\tilde{x_i})), \tilde{y_i})$$

$$\mathcal{L} = \sum_{i=1}^{N} (L_{\text{fused}} + L_{\text{cross}} + L_{\text{self}}) - \lambda L_{\text{corr}}$$

Experiments

Setups and Data

- Two domains:
  - Video-sensor data (ISI dataset)
    - Subjects inject insulin while wearing Google Glass and a wrist sensor
    - Manually labeled to correspond to one of seven actions
  - Audio-Video data (AVLetters and CUAVE)
    - Subjects pronounce the English alphabet and digits 0-9 respectively
    - Video cropped to mouth; audio represented with Mel-Frequency Cepstrum Coefficients
- Each used five multimodal learning settings:

	Feature Learning	Supervised Training	Testing
Multimodal Fusion	$X + Y$	$X + Y$	$X + Y$
Cross Modality Learning	$X + Y$ $X + Y$	X Y	X Y
Shared Represe-ntation Learning	$X + Y$ $X + Y$	X Y	Y X

Experiments

Video-Sensor Data Results

Config	Description
Baseline	Single-layer GRU RNN per modality
Fused	Objective uses only L_{fused} term
Self	Objective uses L_{fused} & L_{self}
Cross	Objective uses L_{fused} & L_{cross}
All	Objective uses L_{fused},L_{self} & L_{cross}
Corr	Objective uses all loss terms
Corr-DW	Objective uses all loss terms & dyn. weights

Table 2. CorrRNN model configurations evaluated

Configuration	Correlation
Fused	0.46
Self	0.67
Cross	0.76
Corr	0.95
Corr-DW	0.93

Table 3. Normalized correlation for different model configurations

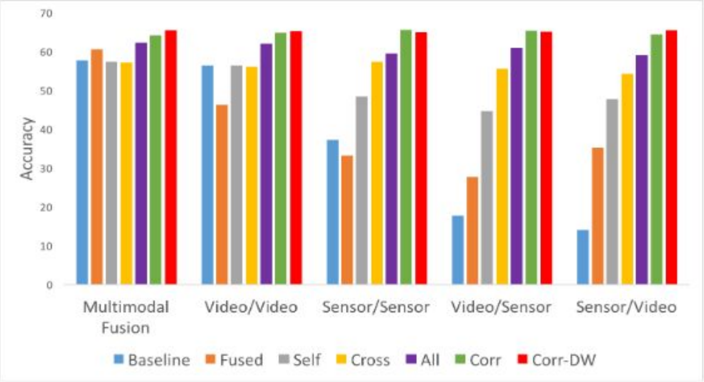


Figure 5. Classification accuracy on the ISI dataset for different model configurations

Experiments

Audio-Video Data Results

Method	Accuracy	
	Clean Audio	Noisy Audio
MDAE	94.4	77.3
Audio RBM	95.8	75.8
MDAE + Audio RBM	94.4	82.2
CorrRNN	**96.11**	**90.88**

Table 6. Classification accuracy for audio-visual speech recognition on the CUAVE dataset, under clean and noisy audio conditions. White Gaussian noise is added to the audio signal at 0dB SNR. Baseline results from [13].

Method	Accuracy	
	AVLetters	CUAVE
MDAE [13]	62.04	66.70
MDBN [21]	63.2	67.20
MDBM [21]	64.7	69.00
RTMRBM [7]	66.04	-
CRBM [1]	67.10	69.10
CorrRNN	**83.40**	**95.9**

Table 4. Classification performance for audio-visual speech recognition on the AVLetters and CUAVE datasets, compared to the best published results in literature, using the fused representation of the two modalities.

	Train /Test	Method	Accuracy	
			AVLetters	CUAVE
Cross-modality learning	Video /Video	Raw	38.08	42.05
		CorrRNN	81.85	96.22
	Audio /Audio	Raw	57.31	88.32
		CorrRNN	85.33	96.11
Shared representation learning	Video /Audio	MDAE	-	24.30
		CorrRNN	85.33	96.77
	Audio /Video	MDAE	-	30.70
		CorrRNN	81.85	96.33

Table 5. Classification accuracy for the cross-modality and shared representation learning settings. MDAE results from [13].

Conclusions

- Good
  - Well-written and fairly easy-to-follow paper
  - Multi-domain experiments
  - Promising results
- Questions
  - Does it actually perform as well for >2 modalities?
  - How much does dynamic weighting add?
  - No comparison on the video/sensor task?
  - Baselines look weird for the audio-video task
  - No argument for efficiency of training
  - Training on asynchronous inputs?