Multi-head modularization to leverage generalization capability in multi-modal networks

Jun-Tae Lee, Hyunsin Park, Sungrack Yun, and Simyung Chang

Qualcomm AI Research*
{juntlee,hyunsinp,sungrack,simychan}@qti.qualcomm.com

Abstract

It has been crucial to leverage the rich information of multiple modalities in many tasks. Existing works have tried to design multi-modal networks with descent multi-modal fusion modules. Instead, we focus on improving generalization capability of multi-modal networks, especially the fusion module. Viewing the multi-modal data as *different projections of information*, we first observe that bad projection can cause poor generalization behaviors of multi-modal networks. Then, motivated by well-generalized network's *low sensitivity to perturbation*, we propose a novel multi-modal training method, *multi-head modularization (MHM)*. We modularize a multi-modal network as a series of uni-modal embedding, multi-modal embedding, and task-specific head modules. Also, for training, we exploit multiple head modules learned with different datasets, swapping each other. From this, we can make the multi-modal embedding module robust to all the heads with different generalization behaviors. In testing phase, we select one of the head modules not to increase the computational cost. Owing to the perturbation of head modules, though including one selected head, the deployed network is more well-generalized compared to the simply end-to-end learned. We verify the effectiveness of MHM on various multi-modal tasks. We use the state-of-the-art methods as baselines, and show notable performance gain for all the baselines.

1 Introduction

Human beings perceive the world through comprehensive information from multiple sensory systems. From this point, it has been a crucial problem to leverage multi-modal data obtained from different sources/structures (*modalities*) in machine learning (Baltrušaitis, Ahuja, and Morency 2018; Wang, Tran, and Feiszli 2020). Most of existing methods (Joze et al. 2020; Liu and Yuan 2018; Abavisani, Joze, and Patel 2019; Ngiam et al. 2011) have tried to obtain collaborative multi-modal representation where both cross- and uni-modal information are conveyed. To this end, diverse multi-modal fusion schemes have been designed, such as averaging (Hazirbas et al. 2016), concatenation (Ngiam et al. 2011; Lee, Uh, and Byun 2020), or attentional alignment (Lee et al. 2021; Tsai et al. 2019).

Although these fusion schemes have made fruitful progress, they have made less effort to understand the relationship between the generalization behaviors of the multi-modal networks and input modalities. To delve into the relationship, we first note that modality is a way to transmit information (Turk 2014; Caschera, Ferri, and Grifoni 2007). If so, the information can be corrupted in some unreliable modalities, and then multi-modal networks may fail to mine discriminative features (Christoudias, Urtasun, and Darrell 2008; Hori et al. 2017).

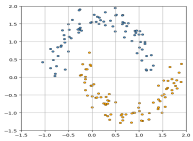
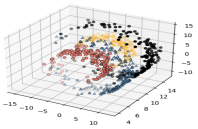
To verify this, viewing multi-modal data as multiple projections of information, we design a toy multi-modal problem (Sec. 3). We show that the generalization behavior of a multi-modal network is affected by the configuration of projections (*i.e.* input modalities). Namely, in some projections, the multi-modal network suffers from the over-fitting problem. However, in real-world problems, it is hard to find the optimal configuration of input modalities, or re-calibrate them. Hence, it is crucial to learn multi-modal networks to have strong generalization capability.

Generalization capability of deep networks mostly depends on both learning algorithms and network architectures (Goodfellow et al. 2016). Therefore, while the existing works have attempted to design multi-modal fusion architecture, we dedicate to the second point, generally applicable learning algorithm to leverage the generalization capability of deep multi-modal networks. To this end, we focus on the small sensitivity of well-generalized networks to perturbations (*e.g.* weight initialization, hyper-parameters, or noise on final weights) (Novak et al. 2018; Jiang et al. 2020; Morcos, Raghu, and Bengio 2018). Inspired by this, we aim to develop a simple but effective training technique making multi-modal representation robust to multiple classification heads with different generalization behaviors. Then, the deployed network can converge to a more general solution with small sensitivity.

More specifically, we propose the multi-head modularization (MHM) algorithm to promote the generalization capability of multi-modal networks. We consider a multi-modal network as a series of uni-modal embedding, multi-modal embedding, and (task-specific) head modules. Also, we exploit multiple head modules instead of a single head during training. The objective of our modularization is to leverage the generalization capability of the multi-modal embedding module. To this end, we split the entire training data and assign

*Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

Table 1: Testing and training errors, and generalization gap of a toy network depending on modality configuration on the two moons and swiss roll datasets, where results for early, intermediate, and late fusions are separated by '/'. Each modality is a projection line which is represented by $\mathbf{p} + t\mathbf{v}$ where $\mathbf{p}$ and $\mathbf{v}$ are a point and a directional vectors, respectively.

Toy dataset	Two moons			Swiss roll		
						
Projection lines	Cartesian	$\mathbf{p}_1$: ⟨1,0⟩, $\mathbf{v}_1$: ⟨-1,1⟩ $\mathbf{p}_2$: ⟨0,0⟩, $\mathbf{v}_2$: ⟨1,0⟩	$\mathbf{p}_1$: ⟨1,0⟩, $\mathbf{v}_1$: ⟨-1,1⟩ $\mathbf{p}_2$: ⟨0,0⟩, $\mathbf{v}_2$: ⟨2,1⟩	Cartesian	$\mathbf{p}_1$: ⟨1,0,0⟩, $\mathbf{v}_1$: ⟨0,-1,0⟩ $\mathbf{p}_2$: ⟨1,0,0⟩, $\mathbf{v}_2$: ⟨1,0,-1⟩ $\mathbf{p}_3$: ⟨0,1,0⟩, $\mathbf{v}_3$: ⟨0,1,-1⟩	$\mathbf{p}_1$: ⟨0,0,0⟩, $\mathbf{v}_1$: ⟨1,1,1⟩ $\mathbf{p}_2$: ⟨0,0,0⟩, $\mathbf{v}_2$: ⟨1,2,1⟩ $\mathbf{p}_3$: ⟨0,0,0⟩, $\mathbf{v}_3$: ⟨1,4,1⟩
Testing error (%)	7.5 / 10.5 / 8.5	17.5 / 17.5 / 27.5	27.5 / 32.5 / 18.0	0.3 / 0.5 / 0.4	22.5 / 23.7 / 23.6	33.7 / 26.5 / 27.5
Training error (%)	6.9 / 8.7 / 4.7	11.9 / 13.5 / 17.5	10.0 / 14.8 / 16.2	0.0 / 0.0 / 0.0	18.7 / 19.1 / 17.2 /	20.0 / 18.8 / 19.4
Generalization gap (%)	0.6 / 1.8 / 3.8	5.6 / 4.0 / 10.0	17.5 / 17.7 / 1.8	0.3 / 0.5 / 0.4	3.8 / 4.6 / 6.4	13.7 / 6.7 / 8.1

the different subsets to each of the multiple head modules. By training alternately these head modules and multi-modal embedding modules, we make the multi-modal embedding module robust to multiple heads which are learned to have different generalization behavior.

For the deployed network, we select one of the head modules. Hence, although using the multiple heads in training, the proposed MHM improves the generalization capability of the multi-modal network without additional computational cost in the testing phase.

In summary, we make following contributions:

- Based on a novel point of view for the multi-modal data, we explore the generalization behavior of multi-modal networks depending on input modalities.
- To promote the generalization sensitivity of multi-modal networks to unseen data, we propose a MHM algorithm to leverage the generalization capability of the multi-modal networks.
- The proposed MHM is generally applicable to various multi-modal networks without additional computational cost in testing phase.
- We conduct extensive experiment to analyze the efficacy of MHM in terms of generalization capability.
- For three multi-modal tasks (audio-visual event detection, action localization, sentiment analysis), we successfully boost the performance of the state-of-the-art methods on benchmark datsets (AVE, THUMOS14, CMU-MOSEI).

2 Related Works

Deep multi-modal learning Many deep multi-modal learning methods can be categorized into early, intermediate, and late fusion. In the early fusion, low-level features of different modalities are simply concatenated as a single input to the following deep architectures. In the intermediate fusion, each modality learns uni-modal representation, and then the uni-modal features are combined to a joint representation in the middle of the network. In the late fusion, modality-specific classifiers make individual predictions, then the final decision is obtained by combining those predictions.

In multi-modal networks, reducing the heterogeneity gap between different modalities is crucial to leverage the complementary relationship of multiple modalities (Guo, Wang, and Wang 2019). Hence, recent deep multi-modal learning methods mostly adopt the intermediate fusion framework. To bridge the different modalities, Zadeh et al. (2017) formulated the intermediate fusion with the outer product of uni-modal features. In (Fukui et al. 2016), the outer product is streamlined by the count sketch projection function. However, still many methods exploit the hidden layers to design the intermediate fusion modules to more robustly combine different modalities using cross-correlation (Lee et al. 2021), directional transformer (Tsai et al. 2019), and embracement layer (Choi and Lee 2019). These methods show the importance of the fusion modules between modalities in multi-modal tasks. We also focuses on the fusion modules. But rather than addressing where/how to fuse different modalities, we improve the generalization behavior of the fusion modules, and thus it can attain better fusion.

Generalization in neural networks There are two lines of approaches for improving the generalization capability of networks. The first focuses on regularization. In the earlier works, data augmentation, noise injection and weight decay (Nowlan and Hinton 1992), and dropout (Srivastava et al. 2014) were proposed and still widely used. In recent, for multi-branch networks, Gastaldi (2017) applied a probabilistic affine combination to typical summation of parallel branches. In (Neyshabur, Tomioka, and Srebro 2015; Neyshabur et al. 2017), the implicit regularization effect of SGD has been studied. However, these techniques are not tailored for multi-modal tasks.

As the second line of works, ensemble has long been a typical solution to improve generalization ability of a single networks (Hansen and Salamon 1990; Dietterich 2000). The ensembled network is more generalized when individual networks have diverse generalization behaviors (Goodfellow et al. 2016). Based on this, diverse basic strategies showed promising results, such as training with randomly initialized weights (Krizhevsky, Sutskever, and Hinton 2012; Simonyan and Zisserman 2015), training with different datasets (Szegedy et al. 2015), capturing multiple snapshots in a training schedule Huang et al. (2017). However, these ensemble methods inevitably require manifold inferences, which drops the computational efficiency in testing. Concurrent to ensemble, our algorithm exploits multiple classification heads on training. But, we make each head well-

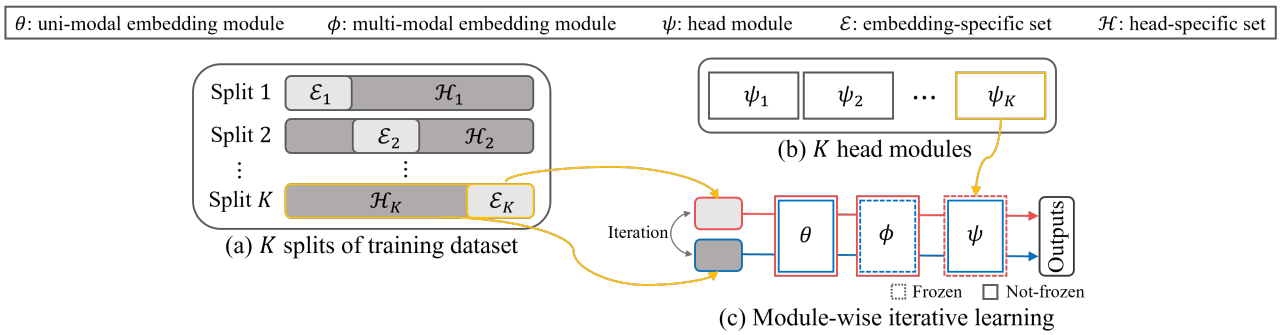


Figure 1: The proposed MHM framework exemplifying that K-th head is used. (a) K different splits of the entire training dataset. (b) K multiple head modules. (c) The multi-modal network is trained iteratively freezing the head or multi-modal embedding modules (Red and blue denote different training pathways where the frozen modules are dashed on each.)

generalized on its own when deployed, and hence don't require additional computation costs in testing.

3 Toy Analysis: Generalization Behavior of Multi-Modal Network

Multi-modal data is acquired by observing (or describing) a moment that occurred in real-world with different modalities. Here, to analyze multi-modal learning on a simple, interpretable system, we construct a scenario for the multi-modal data acquisition process as a projection of high-dimensional data to multiple low-dimensional spaces. Hence, each low-dimensional space corresponds to a modality.

Specifically, we consider two well-known toy datasets: two moons and Swiss roll. As shown in Table 1, in the two moons dataset, data points are categorized into one of two moon classes in 2D space. In the Swiss roll dataset, each data point is labeled by one of four roll classes in 3D space. For both datasets, we generate 100 data points per class. Following our scenario of multi-modal data acquisition, we project the original data points onto multiple lines (two for two moons, three for Swiss roll). In this case, each projection point, represented by a 1D coordinate on a projection line, means an observed data in a modality. Then, the multi-modal representation of a data point is a set of the 1D coordinates obtained from the multiple projection lines.

We design three toy networks for early, intermediate, and late fusions, respectively (See the supplementary materials for their details). Taking as an input the set of projection points, each toy network is learned to classify the input to its ground-truth class. For each class, we randomly select 90% of data points for training and use the remaining for testing.

To analyze the influence of modality configuration on the generalization behavior of multi-modal networks, we project the datasets to diverse sets of projection lines, *i.e.* modality sets. For each dataset, we use three different modality sets. Table 1 shows testing error, training error, and generalization gap (gap of the training and testing errors) on the three networks. In the early fusion network, for the two moon dataset, the generalization gap is increased up to 16.9% in comparison with the original 2D cartesian coordinate system. Even though two sets of projection lines show similar training errors, the generalization gaps are significantly different (5.6% vs. 17.5%). Similarly, in the Swiss roll dataset, compared to 3D cartesian coordinate system, the generalization gap is increased by 3.5% or 13.4%. We see a similar tendency in intermediate and late fusion networks as well.

Thus, we can empirically infer that, in various multi-modal fusion schemes, bad generalization behaviors can be caused by input modalities. However, in real-world scenarios, it is difficult to re-calibrate the modalities of input data. In this paper, we develop an algorithm to leverage the generalization capability of multi-modal networks for given modalities.

4 Multi-Head Modularization (MHM)

In this section, we propose the multi-head modularization (MHM) algorithm to increase the generalization capability of multi-modal networks. Fig. 1 depicts the MHM framework.

Multi-modal problem setting: We suppose two different modalities. They generate the training datasets $\mathcal{X}_1$ and $\mathcal{X}_2$, respectively. $\mathcal{X}_i$ contains data instance $\mathbf{x}_i^n$ where n indicates the instance index. In our setting, $\mathbf{x}_1^n$ and $\mathbf{x}_2^n$ take place at the same time, and share the same label y^n. For convenience, we omit the instance index n. Then, the goal is to learn the multi-modal network $f : (\mathbf{x}_1, \mathbf{x}_2) \rightarrow y$ where y is the shared label for the multi-modal inputs $\mathbf{x}_1$ and $\mathbf{x}_2$. Then, the objective function is represented by

$$\arg\min_f \mathbb{E}_{(\mathbf{x}_1,\mathbf{x}_2)\sim\mathcal{X}_1,\mathcal{X}_2} [\, l(y, f(\mathbf{x}_1, \mathbf{x}_2)\,] \qquad (1)$$

where l is a loss function for a target problem.

Modularization of multi-modal network: We consider the multi-modal network as a series of several modules: uni-modal embedding module θ, multi-modal embedding module ϕ, and (classification) head module ψ, such that $f(\mathbf{x}_1, \mathbf{x}_2) = \psi(\phi(\theta_1(\mathbf{x}_1), \theta_2(\mathbf{x}_2)))$. For each modality i, the uni-modal embedding module θ_i takes $\mathbf{x}_i$ as an input, and propagates the uni-modal feature to the multi-modal embedding module. Next, the multi-modal embedding module combines the uni-modal features of different modalities to a multi-modal feature. From the multi-modal feature, the head module yields outputs for the target problem.

Module-wise learning with multiple heads: Motivated by the low sensitivity of well-generalized networks to perturbation, we induce the individual modules are less sensitive to unseen data. As depicted in Fig. 1(b), we use multiple head modules ψ^k where $k = 1, \ldots, K$. Then, the embedding

Table 2: Comparison of MHM and end-to-end learning (ETE) on toy multi-modal examples. The results of the early, intermediate, late fusions are separated by '/'. Each modality is a projection line which is represented by $\mathbf{p} + t\mathbf{v}$ where $\mathbf{p}$ and $\mathbf{v}$ are a point and a directional vector.

Toy dataset		Two moons		Swiss roll	
	Projection lines	$\mathbf{p}_1$: ⟨1,0⟩, $\mathbf{v}_1$: ⟨-1,1⟩ $\mathbf{p}_2$: ⟨0,0⟩, $\mathbf{v}_2$: ⟨1,0⟩	$\mathbf{p}_1$: ⟨1,0⟩, $\mathbf{v}_1$: ⟨-1,1⟩ $\mathbf{p}_2$: ⟨0,0⟩, $\mathbf{v}_2$: ⟨2,1⟩	$\mathbf{p}_1$: ⟨1,0,0⟩, $\mathbf{v}_1$: ⟨0,-1,0⟩ $\mathbf{p}_2$: ⟨1,0,0⟩, $\mathbf{v}_2$: ⟨1,0,-1⟩ $\mathbf{p}_3$: ⟨0,1,0⟩, $\mathbf{v}_3$: ⟨0,1,-1⟩	$\mathbf{p}_1$: ⟨0,0,0⟩, $\mathbf{v}_1$: ⟨1,1,1⟩ $\mathbf{p}_2$: ⟨0,0,0⟩, $\mathbf{v}_2$: ⟨1,2,1⟩ $\mathbf{p}_3$: ⟨0,0,0⟩, $\mathbf{v}_3$: ⟨1,4,1⟩
ETE	Testing error (%)	17.5 / 17.5 / 17.5	27.5 / 32.5 / 18.0	22.5 / 23.7 / 23.6	33.7 / 26.5 / 27.5
	Generalization gap (%)	5.6 / 4.0 / 10.0	17.5 / 17.7 / 1.8	3.8 / 4.6 / 6.4	13.7 / 6.7 / 8.1
MHM	Testing error (%)	15.0 / 15.0 / 14.5	17.5 / 16.4 / 18.2	18.7 / 14.6 / 18.3	20.0 / 24.3 / 23.4
	Generalization gap (%)	5.6 / 4.0 / 8.5	4.4 / 5.2 / 1.7	1.7 / 1.8 / 2.3	0.3 / 0.2 / 0.3

modules are enforced to be robust to the multiple head modules conveying different generalization behaviors. Note that, rather than learning the multiple head modules together, we repeatedly change the head modules during training.

In specific, we first randomly split the entire training dataset to K same-sized folds, and then assign a head-specific training set $\mathcal{H}_i^k$, composed of $K-1$ folds except kth fold, to each head module ψ^k. Each head-specific training set is partially overlapped with others, but unique. Therefore, we can learn the head modules to show the different generalization behaviors. Then, to make the embedding modules to produce multi-modal features robust to all the head modules, we learn the multi-modal network while changing the head-module.

To further increase the generalization capability of the multi-modal embedding and head modules, for each head module ψ^k, we iteratively learn ψ^k and ϕ with different training data. In other words, the multi-modal embedding module is enforced to be optimized to the head module without observing the training data of the head module, and vice versa.

More specifically, we first freeze the multi-modal embedding module (ϕ), and optimize the uni-modal embedding modules (θ_1, θ_2) and the head (ψ^k) modules with the head-specific sets $\mathcal{H}_1^k$ and $\mathcal{H}_2^k$ as

$$\arg\min_{\theta_1,\theta_2,\psi^k} \mathbb{E}_{(\mathbf{x}_1^{\mathrm{h}},\mathbf{x}_2^{\mathrm{h}})\sim\mathcal{H}_1^k,\mathcal{H}_2^k}\left[\, l(y,\psi^k(\phi(\theta_1(\mathbf{x}_1^{\mathrm{h}}),\theta_2(\mathbf{x}_2^{\mathrm{h}})))\,\right]. \tag{2}$$

Next, we set the embedding specific-set $\mathcal{E}_i^k = \mathcal{X}_i - \mathcal{H}_i^k$, where $\mathcal{E}_i^k$ and $\mathcal{H}_i^k$ are mutually exclusive. Freezing the head module (ψ^k), we learn the uni- and multi-modal embedding modules (θ_1, θ_2, ϕ) by

$$\arg\min_{\theta_1,\theta_2,\phi} \mathbb{E}_{(\mathbf{x}_1^{\mathrm{e}},\mathbf{x}_2^{\mathrm{e}})\sim\mathcal{E}_1^k,\mathcal{E}_2^k}\left[\, l(y,\psi^k(\phi(\theta_1(\mathbf{x}_1^{\mathrm{e}}),\theta_2(\mathbf{x}_2^{\mathrm{e}})))\,\right]. \tag{3}$$

After the iterative module-wise learning of (θ_1, θ_2, ϕ, ψ^k), we replace ψ^k with ψ^{k+1}. To avoid overfitting in the multi-modal embedding and head modules to their own training sets, we consistently learn the uni-modal embedding modules.

As the union of K embedding-specific sets is equal to the entire training set, the multi-modal embedding module can be exposed to the whole training data after the iterative module-wise learning for all k. Thus, the multi-modal embedding module is learned to generate versatile multi-modal features which can maximize the performance of the network for all the head modules. Note that, we select one of the head modules for the deployed network. Hence, in our MHM, the generalization capability is increased without additional computational cost in testing phase.

To identify the efficacy of the proposed MHM, we examine it for the toy problem of Sec. 3. As demonstrated in Table 2, for most cases, MHM lowers testing errors and generalization gaps. Hence, the proposed algorithm has effect on improving the generalization capability of multi-modal networks.

5 Experiments

We provide experimental analysis and comparative evaluation to show the effectiveness of MHM. To this end, we address three multi-modal tasks: audio-visual event detection, RGB-flow action localization, multi-modal sentiment analysis. For each, we employ or design a baseline, and apply MHM.

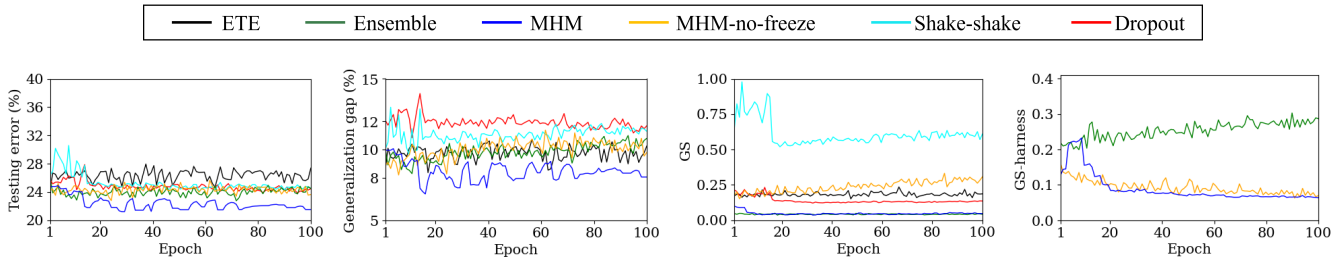
5.1 Experimental Details

More details are in the supplementary materials.

Audio-visual event detection: Audio-visual event detection is to classify each time step (snippet) into one of the event classes or background. Here, we consider the challenging weakly-supervised setting where only video-level labels are available for training, and snippet-level predictions are output in testing. We perform experiments on AVE dataset (Tian et al. 2018). It consists of 3,339 training and 804 testing videos where each lasts 10 seconds with event annotation per second. There are 28 audio-visual event categories.

For this task, we design a light version of the best performing method (Lee et al. 2021) as a baseline to show the effectiveness of the proposed MHM. In (Lee et al. 2021), the multi-modal network extracts the uni-modal feature with a fully-connected (fc) layer for each modality, generates the joint representation of audio and vision by applying the cross correlation-based attention repeatedly, and then classifies it using an open-max classifier. We decompose the learnable cross-correlation matrix as the multiplication of two low-rank vectors. Also, though Lee et al. (2021) repeatedly applied the cross-correlation matrices for audio-visual fusion, we use the low-rank vectors once in our baseline. Then, we set the earliest fc layers as the uni-modal embedding module, the low-rank vectors as the multi-modal embedding module, and the final open-max classifier as the head module. In this task, we use four head modules, empirically.

RGB-flow action localization: In this task, the start, the end, and the class of each action instance are determined given an input video. We consider the extensively studied



(a) Testing error (b) Generalization gap (c) Generalization sensitivity (GS) (d) GS-harnessed

Figure 2: Analysis on generalization capability of the proposed MHM and compared methods.

weakly-supervised setting where only video-level labels are available during the training phase. It has been known that complementary sensory information, RGB and flow, improves the performance. Though, most existing methods simply concatenate RGB and flow inputs. Hence, there are no uni-modal embedding modules. We use THUMOS14 (Jiang et al. 2014) dataset containing 200 training and 212 testing videos for 20 action classes. In a video, each action instance is labeled by start and end time with an action class.

We employ BAS-Net (Lee, Uh, and Byun 2020) which is one of the state-of-the-art methods as the baseline. In brief, BAS-Net consists of two parallel branches (base and background-suppression branches). Both branches take the concatenated uni-modal features as inputs. While the base branch is composed of three convolution layers, the background-suppression branch contains the filtering module and three convolution layers shared with the base branch. The filtering module computes the foreground weights to suppress background information in inputs. We empirically find that better results are obtained when the pre-trained filtering layers are consistently frozen on this network. Hence, we set the first convolution layer as the multi-modal embedding module, and the other layers as the head module. The number of head modules is set to 2 in training by MHM.

Multi-modal sentiment analysis: Multi-modal sentiment analysis extends the language-based sentiment analysis to a multi-modal task, where acoustic, visual, and language modalities are addressed. We evaluate our method for the multi-modal sentiment analysis on CMU-MOSEI (Zadeh et al. 2018) dataset which includes 23,454 movie review video clips. Each clip is labeled by a sentiment strength score ranging from -3 (strong negative) to +3 (strong positive). It is a challenging dataset due to the diverse speakers, a large variance in subjects, and low resource settings.

We use MulT (Tsai et al. 2019) as the baseline. In MulT, the multi-modal network first processes each modality in parallel using a modality-specified block which consists of a convolution layer, two directional cross-modal transformers, and a self-attention transformer. The cross-modal transformer calibrates the feature of the corresponding modality using another modality. Self-attention transformer collects temporal information from the outputs of the cross-modal transformers. Then, the concatenation of the features of three modalities is fed into three fc layers where the last one makes prediction. We set each block as the uni-modal embedding module, the first fc layer as the multi-modal embedding module of three modalities, and the last two fc layers as the head module. Here, K is empirically set as 3.

Table 3: Accuracy, GS, and GS-harness for MHM and compared methods using harnessed multiple baselines.

Model	Acc (%) (↑)	GS (↓)	GS-harnessed
ETE (baseline)	75.6	0.17	-
Ensemble	77.2	0.04	0.27
MHM-no-freeze	77.2	0.19	0.09
MHM (proposed)	78.5	0.04	0.08

5.2 Analysis on Generalization Capability

We analyze the generalization capability of MHM for the audio-visual event detection task. Here, we compare our MHM against to the end-to-end learned baseline (*ETE*), *MHM-no-freeze*, *Ensemble*, and *Dropout* (Srivastava et al. 2014) and *Shake-shake* (Gastaldi 2017) regularizations.

1) MHM-no-freeze: We just perform the iterative learning using four heads and split training sets, without the freezing.

2) Ensemble: We use four baselines with different training datasets, and the averaged output is the final prediction.

3) Dropout and *Shake-shake:* Both exploit randomly selected pathways to promote the generalization capability. For fair comparison with our multi-head modularization, on the baseline, we apply *Dropout* before the classification head with $p = 0.5$, or *Shake-shake* on the head.

Generalization gap: Supposing training and testing datasets share a common data generating process (Goodfellow et al. 2016), a well-generalized network yields a small performance gap between those two datasets. Thus, we first compare testing error and generalization gap varying epochs. As shown in Fig. 2(b), Ensemble and MHM-no-freeze show similar gaps to ETE, and Dropout and Shake-shake regularization are worse than ETE. Whereas, the proposed MHM yields the smallest generalization gap in most epochs. Also, for testing errors in Fig. 2(a), MHM is notably superior to others. Hence, the outstanding performance of the proposed MHM comes from its strong ability to mine the multi-modal data generating process underlain in the training dataset.

Generalization sensitivity: Here, to see whether MHM has effect on reducing the network's sensitivity to perturbation, we quantify the sensitivity of a multi-modal network to unseen dataset. Given a network, we train it multiple times, and measure Kullback-Leibler divergence (KLD) from output distributions of the multiple deployed networks for each

Table 4: Analysis on benefits of MHM in terms of the missing and hitting ratios which measure degradation and improvement against the uni-modal networks, respectively.

Method	ETE (baseline)	MHM (proposed)
Missing ratio (↓)	0.11	0.06
Hitting ratio (↑)	0.03	0.08

testing input. Then, we consider the averaged KLD as generalization sensitivity (GS).

In Fig. 2(c), MHM-no-freeze shows better GS than Shake-shake. Hence, iterative changing of the head module has more beneficial than randomly shaking (weighting) parallel pathways. Dropout yields lower GS than MHM-no-freeze, but our freezing strategy is not applied to MHM-no-freeze. Our final approach (MHM) shows significantly lower GS than Dropout. Also, although Ensemble shows consistently low GS, the harnessed four baselines make prediction as a team. Contrarily, in MHM, GS of a single baseline is comparable to Ensemble. Hence, as in Table 3, the proposed MHM shows highest accuracy (78.5%) as well as the best GS (0.04).

To study how the iterative freezing and multi-head modularization induce to promote generalization sensitivity, we measure GS among harnessed baselines (GS-harness). Namely, in MHM-no-Freeze and MHM, GS is computed among individual baselines corresponding to each head after at every epoch. Similarily, in Ensemble, we compute GS for the ensembled baselines. As in Fig. 2(d), GS-harness of MHM decreases during training, and is mostly lower than that of MHM-no-freeze. In Ensemble, GS-harness gets increasing. From these, we can infer: 1) the multi-head modularization can simulate network perturbation, and helps reducing GS. 2) the freezing strategy makes multi-modal networks more robust. 3) For better GS, whereas Ensemble increases the diversity of the harnessed baselines, MHM encourages a single baseline to be less sensitive. And, MHM is more effective in terms of testing error.

Improvement against uni-modal network: A robust multi-modal network is expected to correctly detect (hit) the testing data even missed in individual uni-modal networks, without failing to detect the ones correctly detected by uni-modal networks. In terms of this aspect, we verify the benefit of the proposed MHM. To this end, for each of two modalities, we construct the uni-modal network which consists of the uni-modal embedding and head of the baseline.

Then, for the proposed MHM and ETE (baseline), we measure missing and hitting ratios for testing data. Missing ratio is the ratio of data which are detected incorrectly in multi-modal network but correctly in any uni-modal networks. Hitting ratio is the ratio of data detected correctly in multi-modal network but incorrectly in both uni-modal ones. As in Table 4, our MHM yields higher hitting and lower missing ratios than ETE by 0.03 for both. Thus, the proposed MHM exploits the richer information of multi-modal data, faithfully.

5.3 Ablation Studies

We study the impact of multi-modal embedding and classification head modules on MHM.

Table 5: Adequacy of the number of folds, K.

No. fold (K)	2	3	4	5
Acc (%)	76.0	77.8	**78.5**	78.0
GS	0.11	0.05	0.04	0.04

Hyper-parameter K: We analyze the adequacy of K. To this end, by varying the number of folds from 2 to 5, we report accuracy and GS in Table 5. Even 2-fold improves the ETE (baseline) (75.6% to 76.0%) with lower GS (0.11 vs. 0.17). Reducing GS further (0.11 to 0.05), 3-fold gives further improvement as the multi-modal embedding module can observe more diverse statistics. Then, the performance is the highest at $K = 4$ with a bit better GS (0.04). Hence, the relationship between accuracy and GS means that MHM has effect on promoting generalization sensitivity. When $K = 5$, the performance is slightly lower than $K = 4$. As K gets bigger, the multi-modal embedding module is trained with a smaller embedding specific set at each iteration. This causes saturated performance.

Multiple head modules: To see the effect of the multiple head modules, we compare MHM with four variants: 1 head module with 1 or 4 head-specific sets, 4 head modules with 1 head-specific set, and ensemble of 4 head modules. In 1 head module with 1 head-specific set, the head module is not changed during the module-wise iterative learning. In 1 head module with 4 head-specific sets, the head module is not changed, but the head-specific set does after an epoch. In 4 head modules with 1 head-specific set, only a single head-specific set is used for all of the head modules. In 4 head ensemble, whereas head modules are learned independently with different head-specific sets similarly to MHM, the multi-modal embedding module is learned using the ensemble of the heads (the objective loss is computed using average of head outputs). In testing, all head outputs are averaged.

Table 6 reports accuracy scores of the proposed MHM and the four variants. As in Table 6, all four variants degrade the performance of the MHM by 3.0%, 2.9%, and 2.5%, respectively. Hence, we see that using multiple head modules with different head-specific sets induces more different generalization behaviors among the head modules. Though the head ensemble is comparable to our MHM, it requires larger computational cost in testing.

Multi-modal embedding module: Proposed MHM exploits the combination of multiple head modules and one multi-modal embedding module. To see the adequacy of this topology, we compare the proposed MHM with a different modularization with multiple multi-modal embedding modules (MMM). In MMM, dissimilar with the proposed MHM, the multi-modal embedding modules are changed during the module-wise iterative learning. For fair comparison with MHM, we set the number of multi-modal embedding modules as 4 in MMM. The proposed MHM shows a higher accuracy score than MMM by 5.1% (78.5% MHM vs. 73.4% MMM). Moreover, MMM is even inferior to ETE (75.6% ETE vs. 73.4% MMM). Hence, combination of a single multi-modal embedding and multiple head modules generates more robust multi-modal features.

Table 6: Ablation for the numbers of head modules and head-specific sets.

Method	1 head module		4 head modules		Ensemble of 4 head modules
	1 head specific-set	4 head specific-sets	1 head specific-set	4 head specific-sets (MHM)	4 head specific-sets
Acc (%)	75.5	75.6	76.0	**78.5**	78.4

Table 7: Comparison of the proposed and existing weakly-supervised audio-visual event detection methods on AVE. Against baseline + ETE, performance gain is also provided.

Method	Acc (%)	Gain (%)
(Tian et al. 2018)	73.1	-
(Lin, Li, and Wang 2019)	74.2	-
(Xuan et al. 2020)	75.7	-
(Lee et al. 2021)	<u>77.1</u>	-
baseline + ETE	75.6	-
baseline + MHM	**78.5**	2.9

Table 8: Comparison of the proposed MHM with the existing methods for the weakly-supervised action localization on THUMOS14. † denotes the reproduced scores with the authors' source code.

Method	Avg. mAP (%)	Gain (%)
(Liu, Jiang, and Wang 2019)	32.4	-
(Narayan et al. 2019)	31.9	-
(Nguyen, Ramanan, and Fowlkes 2019)	36.3	-
(Shi et al. 2020)	**37.0**	-
Bas-Net (ETE)† (Lee, Uh, and Byun 2020)	34.6	-
Bas-Net + MHM (Ours)	<u>36.4</u>	1.8

5.4 Comparative Evaluation

Audio-visual event detection: In Table 7, we compare the proposed 'baseline + MHM' with the recent methods. Following the compared methods, we report snippet-wise event prediction accuracy on the AVE dataset.

Note that our baseline is much smaller than the network of the existing state-of-the-art method (Lee et al. 2021). Nevertheless, contrary to the 'baseline + ETE' (the end-to-end learned baseline), the proposed 'baseline + MHM' outperforms over all the existing methods. Moreover, in the view of boosting the generalization capability, the most important point is that 'baseline + MHM' shows significantly higher accuracy than 'baseline + ETE' by 2.9%. Hence, the proposed MHM is very useful to increase the generalization capability of the multi-modal network.

RGB-flow action localization: For the quantitative evaluation, we compute the mean average precision (mAP) scores for action segments at different intersection over union (IoU) thresholds [0.1:0.1:0.7]. Table 8 compares the recent methods including our baseline, and the proposed methods for the averaged mAP. For the baseline BAS-Net, we reproduce its performance using the authors' source code.

Although the baseline 'Bas-Net' shows a decent performance (34.6% avg. mAP), it is lower than (Shi et al. 2020) and (Nguyen, Ramanan, and Fowlkes 2019) with large gaps by 2.4% and 1.7%, respectively. Contrarily, the proposed 'Bas-Net + MHM' notably boosts the baseline by 1.8% for avg. mAP. The boosted performance (36.4% avg. mAP) achieves the second best score, and is comparable to the highest performing method (Shi et al. 2020). From this, we successfully show that the proposed MHM algorithm is simple, but highly effective to increase the generalization capability of the multi-modal networks in this task, as well.

Table 9: Comparison of the proposed method with the previous multi-modal sentiment analysis methods in terms of accuracy for seven classes (Acc_7), and accuracy (Acc_2) and F1-score for binary classes on CMU-MOSEI. † denotes the reproduced score with the authors' source code.

Model	Acc_7 (%)	Acc_2 (%)	F1-score (%)
CTC + RAVEN (Wang et al. 2019)	45.5	75.4	75.7
CTC + MCTN (Pham et al. 2019)	48.2	79.3	79.7
MulT (Tsai et al. 2019) †	49.1	80.6	81.0
MulT + MHM (Ours)	**50.4**	**80.9**	**81.4**

Multi-modal sentiment analysis: As in the existing works, we use 7-class accuracy (Acc_7), binary accuracy (Acc_2), and F1-score as the evaluation metrics. In Acc_7, the predicted sentiment score is mapped to 7 integer scores in [-3:1:3]. In Acc_2, the sentiment score is categorized into the binary classes (positive or negative sentiments). The F1-score is computed for the binary classification. Table 9 reports the results of the proposed 'MulT + MHM,' the baseline 'MulT,' and the existing methods, 'CTC + RAVEN' (Wang et al. 2019) and 'CTC + MCTN' (Pham et al. 2019). For all metrics, the proposed 'MulT + MHM' achieves the highest performance. Also, even with the same model structure and computation, 'MulT + MHM' improves the baseline which is the existing state-of-the-art method, for all of the metrics. Especially, 'MulT + MHM' exceeds the baseline for the most challenging metric, Acc_7, by a large gap 1.3%.

6 Conclusions

We first observed the relationship between generalization behavior and input modalities in a toy problem. Then, inspired by the low sensitivity of well-generalized network to perturbation, we proposed a novel algorithm (MHM) to boost the generalization capability of the multi-modal networks. Exploiting multiple classification head modules, we iteratively learn the multi-modal embedding module and the head module. Varying the head module, we use different head- and embedding-specific training sets in iterative learning. From this, the multi-modal embedding module can produce multi-modal features robust to all the head modules with different generalization behaviors. By selecting one of the multiple head modules, we do not increase the computational cost in the deployed network. We extensively studied the generalization capability of the proposed MHM according to generalization gap and sensitivity, and showed notable performance gain in three multi-modal tasks.

References

Abavisani, M.; Joze, H. R. V.; and Patel, V. M. 2019. Improving the performance of unimodal dynamic hand-gesture recognition with multimodal training. In *CVPR*.

Baltrušaitis, T.; Ahuja, C.; and Morency, L.-P. 2018. Multimodal machine learning: A survey and taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(2): 423–443.

Caschera, M. C.; Ferri, F.; and Grifoni, P. 2007. Multimodal interaction systems: information and time features. *Int. J. Web Grid Serv.*, 3(1): 82–99.

Choi, J.-H.; and Lee, J.-S. 2019. EmbraceNet: A robust deep learning architecture for multimodal classification. *Inf. Fusion*, 51: 259–270.

Christoudias, C. M.; Urtasun, R.; and Darrell, T. 2008. Multiview learning in the presence of view disagreement. In *Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*.

Dietterich, T. G. 2000. Ensemble methods in machine learning. In *International Workshop on Multiple Classifier Systems*.

Fukui, A.; Park, D. H.; Yang, D.; Rohrbach, A.; Darrell, T.; and Rohrbach, M. 2016. Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding. In *EMNLP*.

Gastaldi, X. 2017. Shake-shake regularization. Forthcoming, arXiv:1705.07485.

Goodfellow, I.; Bengio, Y.; Courville, A.; and Bengio, Y. 2016. *Deep learning*, volume 1. MIT press Cambridge.

Guo, W.; Wang, J.; and Wang, S. 2019. Deep multimodal representation learning: A survey. *IEEE Access*, 7: 63373–63394.

Hansen, L. K.; and Salamon, P. 1990. Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(10): 993–1001.

Hazirbas, C.; Ma, L.; Domokos, C.; and Cremers, D. 2016. Fusenet: Incorporating depth into semantic segmentation via fusion-based CNN architecture. In *ACCV*.

Hori, C.; Hori, T.; Lee, T.-Y.; Zhang, Z.; Harsham, B.; Hershey, J. R.; Marks, T. K.; and Sumi, K. 2017. Attention-based multimodal fusion for video description. In *ICCV*.

Huang, G.; Li, Y.; Pleiss, G.; Liu, Z.; Hopcroft, J. E.; and Weinberger, K. Q. 2017. Snapshot ensembles: Train 1, get m for free. In *ICLR*.

Jiang, Y.; Neyshabur, B.; Mobahi, H.; Krishnan, D.; and Bengio, S. 2020. Fantastic generalization measures and where to find them. In *ICLR*.

Jiang, Y.-G.; Liu, J.; Roshan Zamir, A.; Toderici, G.; Laptev, I.; Shah, M.; and Sukthankar, R. 2014. THUMOS Challenge: Action Recognition with a Large Number of Classes. http://crcv.ucf.edu/THUMOS14/.

Joze, H. R. V.; Shaban, A.; Iuzzolino, M. L.; and Koishida, K. 2020. MMTM: Multimodal transfer module for CNN fusion. In *CVPR*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *NeurIPS*.

Lee, J.-T.; Jain, M.; Park, H.; and Yun, S. 2021. Cross-attentional audio-visual fusion for weakly-supervised action localization. In *ICLR*.

Lee, P.; Uh, Y.; and Byun, H. 2020. Background suppression network for weakly-supervised temporal action localization. In *AAAI*.

Lin, Y.; Li, Y.; and Wang, Y. F. 2019. Dual-modality Seq2Seq network for audio-visual event localization. In *ICASSP*.

Liu, D.; Jiang, T.; and Wang, Y. 2019. Completeness modeling and context separation for weakly supervised temporal action localization. In *CVPR*.

Liu, M.; and Yuan, J. 2018. Recognizing human actions as the evolution of pose estimation maps. In *CVPR*.

Morcos, A. S.; Raghu, M.; and Bengio, S. 2018. Insights on representational similarity in neural networks with canonical correlation. In *NeurIPS*.

Narayan, S.; Cholakkal, H.; Khan, F. S.; and Shao, L. 2019. 3C-Net: Category count and center loss for weakly-supervised action localization. In *ICCV*.

Neyshabur, B.; Bhojanapalli, S.; McAllester, D.; and Srebro, N. 2017. Exploring generalization in deep learning. In *NeurIPS*.

Neyshabur, B.; Tomioka, R.; and Srebro, N. 2015. In search of the real inductive bias: On the role of implicit regularization in deep learning. In *ICLR Workshop*.

Ngiam, J.; Khosla, A.; Kim, M.; Nam, J.; Lee, H.; and Ng, A. Y. 2011. Multimodal deep learning. In *ICML*.

Nguyen, P. X.; Ramanan, D.; and Fowlkes, C. C. 2019. Weakly-supervised action localization with background modeling. In *ICCV*.

Novak, R.; Bahri, Y.; Abolafia, D. A.; Pennington, J.; and Sohl-Dickstein, J. 2018. Sensitivity and generalization in neural networks: an empirical study. In *ICLR*.

Nowlan, S. J.; and Hinton, G. E. 1992. Simplifying neural networks by soft weight-sharing. *Neural Comput.*, 4(4): 473–493.

Pham, H.; Liang, P. P.; Manzini, T.; Morency, L.-P.; and Póczos, B. 2019. Found in translation: Learning robust joint representations by cyclic translations between modalities. In *AAAI*.

Shi, B.; Dai, Q.; Mu, Y.; and Wang, J. 2020. Weakly-supervised action localization by generative attention modeling. In *CVPR*.

Simonyan, K.; and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. Forthcoming, arXiv:1409.1556.

Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: A simple way to prevent neural networks from overfitting. volume 15, 1929–1958.

Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; and Rabinovich, A. 2015. Going deeper with convolutions. In *CVPR*.

Tian, Y.; Shi, J.; Li, B.; Duan, Z.; and Xu, C. 2018. Audio-Visual Event Localization in Unconstrained Videos. In *ECCV*.

Tsai, Y.-H. H.; Bai, S.; Liang, P. P.; Kolter, J. Z.; Morency, L.-P.; and Salakhutdinov, R. 2019. Multimodal transformer for unaligned multimodal language sequences. In *ACL*.

Turk, M. 2014. Multimodal interaction: A review. *Pattern Recognit. Lett.*, 36: 189–195.

Wang, W.; Tran, D.; and Feiszli, M. 2020. What makes training multi-modal classification networks hard? In *CVPR*.

Wang, Y.; Shen, Y.; Liu, Z.; Liang, P. P.; Zadeh, A.; and Morency, L.-P. 2019. Words can shift: Dynamically adjusting word representations using nonverbal behaviors. In *AAAI*.

Xuan, H.; Zhang, Z.; Chen, S.; Yang, J.; and Yan, Y. 2020. Cross-Modal Attention Network for Temporal Inconsistent Audio-Visual Event Localization. In *AAAI*.

Zadeh, A.; Chen, M.; Poria, S.; Cambria, E.; and Morency, L.-P. 2017. Tensor fusion network for multimodal sentiment analysis. In *EMNLP*.

Zadeh, A. B.; Liang, P. P.; Poria, S.; Cambria, E.; and Morency, L.-P. 2018. Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph. In *ACL*.