HighMMT: Quantifying Modality & Interaction Heterogeneity for High-Modality Representation Learning

Anonymous authors
Paper under double-blind review

Abstract

Many real-world problems are inherently multimodal, from the communicative modalities humans use to express social and emotional states such as spoken language, gestures, and paralinguistics to the force, proprioception, and visual sensors ubiquitous on robots. While there has been an explosion of interest in multimodal representation learning, these methods are still largely focused on a small set of modalities, primarily in the language, vision, and audio space. In order to accelerate generalization towards diverse and understudied modalities, this paper studies efficient representation learning for *high-modality scenarios* involving a large set of diverse modalities. Since adding new models for every new modality or task becomes prohibitively expensive, a critical technical challenge is *heterogeneity quantification*: how can we measure which modalities encode *similar information* and *interactions* in order to permit parameter sharing with previous modalities? This paper proposes two new information theoretic metrics for heterogeneity quantification: (1) *modality heterogeneity* studies how similar 2 modalities $\{X_1, X_2\}$ are by measuring how much information can be transferred from X_1 to X_2, while (2) *interaction heterogeneity* studies how similarly pairs of modalities $\{X_1, X_2\}, \{X_3, X_4\}$ interact by measuring how much interaction information can be transferred from $\{X_1, X_2\}$ to $\{X_3, X_4\}$. We show the importance of these 2 proposed metrics in high-modality scenarios as a way to automatically prioritize the fusion of modalities that contain unique information or unique interactions. The result is a single model, HighMMT, that scales up to 10 modalities (text, image, audio, video, sensors, proprioception, speech, time-series, sets, and tables) and 15 tasks from 5 different research areas. Not only does HighMMT outperform prior methods on the tradeoff between performance and efficiency, it also demonstrates a crucial scaling behavior: performance continues to improve with each modality added, and it transfers to entirely new modalities and tasks during fine-tuning. We release our code and benchmarks, which we hope will present a unified platform for subsequent theoretical and empirical analysis.

1 Introduction

Multimodal machine learning brings unique challenges for both computational and theoretical research given the heterogeneity of various data sources (Liang et al., 2022). While there have been impressive advances in modeling language, vision, and audio (Agrawal et al., 2017; Ramesh et al., 2021), advances in sensing technologies have resulted in many real-world platforms such as cellphones, smart devices, self-driving cars, healthcare technologies, and robots now integrating a much larger number of sensors such as time-series, proprioception, sets, tables, and high-frequency sensors (Frantzidis et al., 2010; Lee et al., 2019; Leiva et al., 2020; Liang et al., 2021a; Belpaeme et al.,

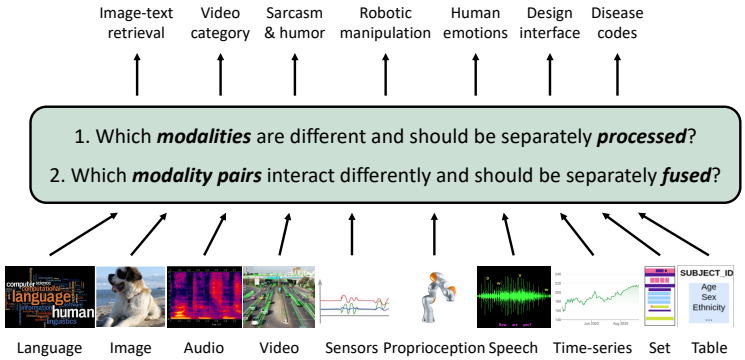

Figure 1: **Heterogeneity quantification**: Efficiently learning from many modalities requires measuring (1) *modality heterogeneity*: which modalities are different and should be separately processed, and (2) *interaction heterogeneity*: which modality pairs interact differently and should be separately fused. Our proposed HighMMT model uses these new measurements to dynamically group parameters to balance both performance and efficiency.

2018; Yeong et al., 2021). This new setting of *high-modality learning* involves learning representations over many diverse modality inputs. As more modalities are introduced, adding new model parameters for every new modality or task (Tsai et al., 2019; Jayakumar et al., 2020; Lu et al., 2019) becomes prohibitively expensive and not scalable (Liang et al., 2022). A critical technical challenge for efficient high-modality learning, therefore, is *heterogeneity quantification*: how can we measure which modalities encode *similar information* and *similar interactions* in order to permit parameter sharing with previous modalities (see Figure 1)? For example, how can one determine whether the same modality encoder can be shared when processing language and speech, or that the same fusion network can be shared when fusing human speech and gestures as well as robot visual and force sensors?

In this paper, we propose a principled approach for heterogeneity quantification via modality information transfer, an information-theoretic approach that measures the amount of transferable usable information (Xu et al., 2019) from one modality to another. Our first proposed metric, (1) *modality heterogeneity* studies how similar 2 modalities $\{X_1, X_2\}$ are by measuring how much usable information can be transferred from X_1 to X_2, and our second metric, (2) *interaction heterogeneity* studies how similarly 2 modality pairs $\{X_1, X_2\}, \{X_3, X_4\}$ interact by measuring how much usable interaction information can be transferred from $\{X_1, X_2\}$ to $\{X_3, X_4\}$. We show the importance of these 2 proposed metrics in high-modality scenarios as a way to automatically prioritize the fusion of modalities that contain unique information or unique interactions, and otherwise sharing parameters across similar modalities displaying similar information or interactions.

Operationalizing these ideas on a suite of 10 modalities, 15 prediction tasks, and 5 research areas, we show how to train a single model, HighMMT, that (1) improves the tradeoff between performance and efficiency over task-specific state-of-the-art models (Liang et al., 2021b; Jayakumar et al., 2020), and general multimodal models with full parameter sharing (Jaegle et al., 2021b; Hu and Singh, 2021; Akbari et al., 2021; Reed et al., 2022), (2) enables cross-modal transfer by pretraining on source multimodal tasks before transferring to new target modalities and tasks, and (3) is especially beneficial for low-resource scenarios (less training data and partially-observable modalities). Beyond these empirical results, we believe that our insights on quantifying heterogeneity and information sharing in multimodal models are independently useful for future work. Our implementations and benchmarks are publicly available which we hope will present a unified platform for subsequent theoretical and empirical analysis.

2 High-Modality Multimodal Transformer

In this section, we describe our overall approach for high-modality representation learning (see Figure 2). In §2.1, we formalize modality and interaction heterogeneity to understand whether modalities should be processed similarly or differently. Using these insights, §2.2 describes our proposed HighMMT model with dynamic parameter sharing based on heterogeneity measurements.

2.1 Measuring Heterogeneity via Modality Information Transfer

Modality heterogeneity seeks to answer the question: how differently should I encode modality X_1 versus X_2? *Interaction heterogeneity* aims to answer: how differently should I fuse modalities $\{X_1, X_2\}$ versus $\{X_3, X_4\}$? Together, they help us appropriately design both unimodal and crossmodal components of a multimodal model. We will formalize heterogeneity via *modality transfer*, an information-theoretic approach that measures the amount of transferable usable information from one modality to another.

Background: Information theory and usable information. Information theory is a useful framework to study the utility of data for prediction tasks. Specifically, the mutual information (Shannon, 1948) $I(X;Y)$ measures the amount of uncertainty reduced from $H(Y)$ to $H(Y|X)$ when given X as input, and its estimation has been central to studying (Ethayarajh et al., 2022) and improving (Tishby et al., 2000; Tschannen et al., 2019) representation learning in both unimodal (Vera et al., 2018; Xu et al., 2019) and multimodal settings (Colombo et al., 2021; Tian et al., 2020; Tosh et al., 2021). Recently, $\mathcal{V}$-usable information extends traditional Shannon information theory to account for computational constraints (Ethayarajh et al., 2022; Xu et al., 2019): $I_{\mathcal{V}}(X \to Y)$ reflects the ease with which a model family $\mathcal{V}$ can predict outcomes Y

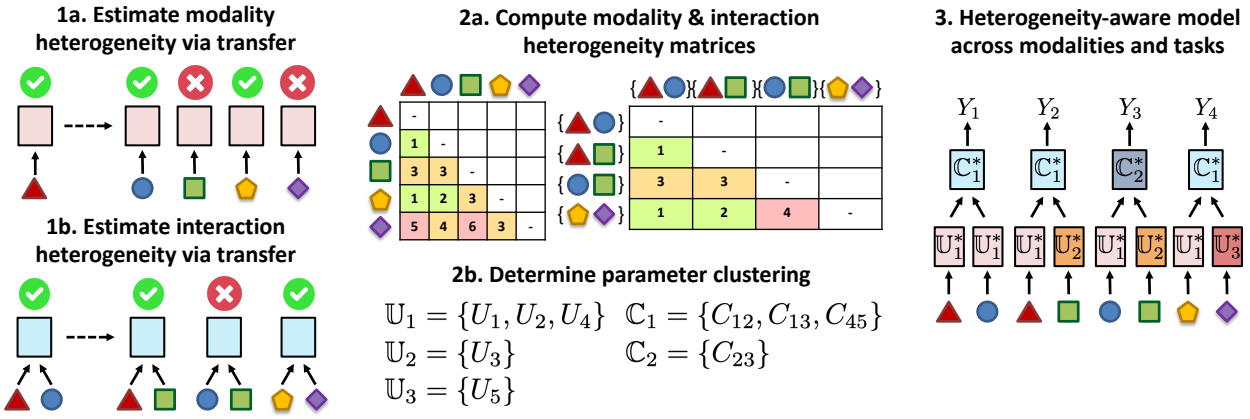

Figure 2: **HighMMT workflow**: (1) We estimate modality and interaction heterogeneity via modality transfer to determine which modalities should be processed and fused differently. (2) Using the inferred heterogeneity, we determine the optimal grouping of parameters balancing both total performance and parameter efficiency, which (3) informs our design of a heterogeneity-aware model with dynamic parameter sharing across many modalities and tasks. HighMMT enables statistical strength sharing, efficiency, and generalization to new modalities and tasks.

given inputs X:

$$H_{\mathcal{V}}(Y) = \inf_{f \in \mathcal{V}} \mathbb{E}[-\log_2 f[\varnothing](Y)], \quad H_{\mathcal{V}}(Y|X) = \inf_{f \in \mathcal{V}} \mathbb{E}[-\log_2 f[X](Y)], \tag{1}$$

$$I_{\mathcal{V}}(X \to Y) = H_{\mathcal{V}}(Y) - H_{\mathcal{V}}(Y|X). \tag{2}$$

where $f[X]$ and $f[\varnothing]$ are both models that produce a probability distribution over the labels, and the goal is to find $f \in \mathcal{V}$ that maximizes the log-likelihood of the data with ($H_{\mathcal{V}}(Y|X)$) and without ($H_{\mathcal{V}}(Y)$) the input. Conditioning information on a specific model family $\mathcal{V}$ can help to explain the difficulties of encryption applied to X, or the benefits of feature representation learning applied to X. For example, processing the input with τ (e.g., decryption or representation learning) can make prediction easier, allowing $I_{\mathcal{V}}(\tau(X) \to Y) \geq I_{\mathcal{V}}(X \to Y)$, whereas traditional information theory measures of $I(X;Y)$ would be invariant to these processing steps. Finally, setting $\mathcal{V}$ as the set of all functions (i.e., under unbounded computation) reduces to Shannon information.

Estimating modality heterogeneity via unimodal information transfer. We propose to measure heterogeneity between modalities X_1 and X_2 via unimodal transfer. Given a task Y defined over X_1 and X_2, how well does an unimodal model trained on $(X_1;Y)$ transfer to $(X_2;Y)$? We choose model transfer as our focus of heterogeneity since it is captured at the level of features extracted via representation learning, rather than at the data-level. Even though the input data may be very different (e.g., images from different cameras or paraphrased sentences), effective feature extractors may be able to learn similar representations from them. Furthermore, it directly models task-relevance: the degree of heterogeneity depends on the end task, which enables using these heterogeneity measures subsequently for end-task optimization.

$\mathcal{V}$-usable information provides a nice formalism to compute unimodal transfer, via the difference in usable information between unimodal models trained on X_1 before transfer to X_2, versus those trained directly on X_2. Specifically, first set the model family $\mathcal{V}$ as the family of unimodal networks on target task $(X_2;Y)$, and the model family $\mathcal{V}(X_1)$ as the family of unimodal networks initialized with pre-trained parameters on task $(X_1;Y)$. Then, the transfer difficulty can be defined as $T(X_1 \to X_2;Y) = I_{\mathcal{V}}(X_2 \to Y) - I_{\mathcal{V}(X_1)}(X_2 \to Y)$. Intuitively, $I_{\mathcal{V}}(X_2 \to Y)$ measures the (baseline) usable information in $\mathcal{V}$ to predict Y given X_2, while $I_{\mathcal{V}(X_1)}(X_2 \to Y)$ measures the usable information in $\mathcal{V}(X_1)$ to predict Y given X_2. In our experiments, we find that both $\mathcal{V}(X_1)$ and $\mathcal{V}$ are both expressive enough such that $H_{\mathcal{V}(X_1)}(Y) \approx H_{\mathcal{V}}(Y)$ reduce to the label entropy, so the simplified form

$$T(X_1 \to X_2;Y) = I_{\mathcal{V}}(X_2 \to Y) - I_{\mathcal{V}(X_1)}(X_2 \to Y) \approx H_{\mathcal{V}(X_1)}(Y|X_2) - H_{\mathcal{V}}(Y|X_2) \tag{3}$$

measures the difficulty of transferring a model trained on the source task $(X_1;Y)$ to a target task $(X_2;Y)$. Note that computing $T(X_1 \to X_2;Y)$ only requires the training or fine-tuning of 2 models across the source and target modalities, which is efficient.

What are some properties of $T(X_1 \to X_2; Y)$? For very different modalities X_1 and X_2, we typically expect $\mathcal{V}(X_1)$ to contain less usable information than $\mathcal{V}$ for a target task $(X_2; Y)$, which would imply that $T(X_1 \to X_2; Y) \geq 0$ (i.e., positive difficulty). This is consistent with work demonstrating negative transfer across different modalities (Liang et al., 2021b;c; Tulving and Watkins, 1974; Wang et al., 2019). Under these scenarios, the larger the positive magnitude of $T(X_1 \to X_2; Y)$, the more different modalities X_1 and X_2 are in the context of task Y (more difficult to transfer). However, there can also be cases of zero or even positive transfer (i.e., $T(X_1 \to X_2; Y) \leq 0$), even in the surprising case of very different modalities (Lu et al., 2021). These cases reinforce the benefits of feature-based approaches to measure heterogeneity: while the raw modalities themselves seem very different, they can still be processed by similar models resulting in positive transfer, and should be assigned a difference of 0. Our final heterogeneity measure $d(X_1; X_2)$ aggregates the absolute value (to account for positive transfer) of transfer difficulty statistics across tasks $Y \in \mathcal{Y}$ and across both transfer directions $X_1 \to X_2$ and $X_2 \to X_1$:

$$d(X_1; X_2) = \sum_{Y \in \mathcal{Y}} |T(X_1 \to X_2; Y)| + \sum_{Y \in \mathcal{Y}} |T(X_2 \to X_1; Y)|. \tag{4}$$

We note that our modality heterogeneity measure $d(X_1; X_2)$ is not a strict metric space: while it satisfies *non-negativity*: $d(X_1; X_2) \geq 0$, with $d(X_1; X_2) = 0$ when $X_1 = X_2$, *symmetry*: $d(X_1; X_2) = d(X_2; X_1)$, and *triangle inequality*: $d(X_1; X_3) \leq d(X_1; X_2) + d(X_2; X_3)$, it does not satisfy *positivity*: there are a few cases of $X_1 \neq X_2$ but $d(X_1; X_2) = 0$ due to positive transfer.

Estimating interaction heterogeneity via crossmodal information transfer. We are also interested in interaction heterogeneity: specifically, how differently should I fuse modalities $\{X_1, X_2\}$ versus $\{X_3, X_4\}$? We therefore extend to crossmodal transfer by comparing the difference in usable information between a pretrained multimodal model on $(X_1, X_2; Y)$ before transfer to $(X_3, X_4; Y)$, versus those trained directly on the target task $(X_3, X_4; Y)$. In other words, we measure the quantity

$$T(X_1, X_2 \to X_3, X_4; Y) = I_{\mathcal{V}}(X_3, X_4 \to Y) - I_{\mathcal{V}(X_1, X_2)}(X_3, X_4 \to Y) \tag{5}$$

$$\approx H_{\mathcal{V}(X_1, X_2)}(Y | X_3, X_4) - H_{\mathcal{V}}(Y | X_3, X_4). \tag{6}$$

The resulting set of distances $d(X_1, X_2; X_3, X_4)$ after aggregation over tasks and transfer directions estimates the interaction heterogeneity between $\{X_1, X_2\}$ and $\{X_3, X_4\}$.

Modality and interaction heterogeneity matrix. Finally, we construct a modality heterogeneity matrix $M_U(i, j) = d(X_i; X_j)$ and an interaction heterogeneity matrix (technically 4D-tensor) $M_C(i, j, k, \ell) = d(X_i, X_j; X_k, X_\ell)$. As a side note, observe that these matrices are highly structured due to distances satisfying the triangle inequality, which implies that we do not need to compute all entries and instead rely on low-rank reconstruction from partial entries in practice (Drineas et al., 2006; Tasissa and Lai, 2018) (see Appendix A for details, and see an example in §3.1).

Determining parameter groupings to balance both total performance and parameter efficiency can be solved via agglomerative hierarchical clustering where modalities are nodes and heterogeneity measurements are edges. The number of clusters k is treated as a hyperparameter dependent on the parameter budget (see Appendix A for details, and see clustering examples in §3.1). Clustering on the modality heterogeneity matrix M_U results in a grouping of modalities based on similarity (e.g., $\mathcal{U}_1 = \{X_1, X_2, X_4\}, \mathcal{U}_2 = \{X_3\}, \mathcal{U}_3 = \{X_5\}$), and likewise for the crossmodal matrix M_C (e.g., $\mathcal{C}_1 = \{\{X_1, X_2\}, \{X_1, X_3\}, \{X_4, X_5\}\}, \mathcal{C}_2 = \{\{X_2, X_3\}, \mathcal{C}_3 = \{\{X_4, X_6\}, \{X_5, X_6\}\}$, and so on.

2.2 Capturing Heterogeneity and Homogeneity in HighMMT

Using these insights, we now describe our approach for a general model HighMMT suitable for high-modality representation across many modalities and tasks. Our approach takes 2 main steps (see Figure 3): (1) *homogeneous pre-training* of a fully shared model across all modalities, before (2) *heterogeneity-aware fine-tuning* to respect modality and interaction heterogeneity.

Homogeneous pre-training. We first design a homogeneous multimodal model fully shared across all modalities and tasks with the following key components (see more details in Appendix B).

1. Standardized input sequence: We first standardize modalities as a sequence of embeddings, as is already done for sequential data such as text, audio, and time series, and recently adapted for image patches

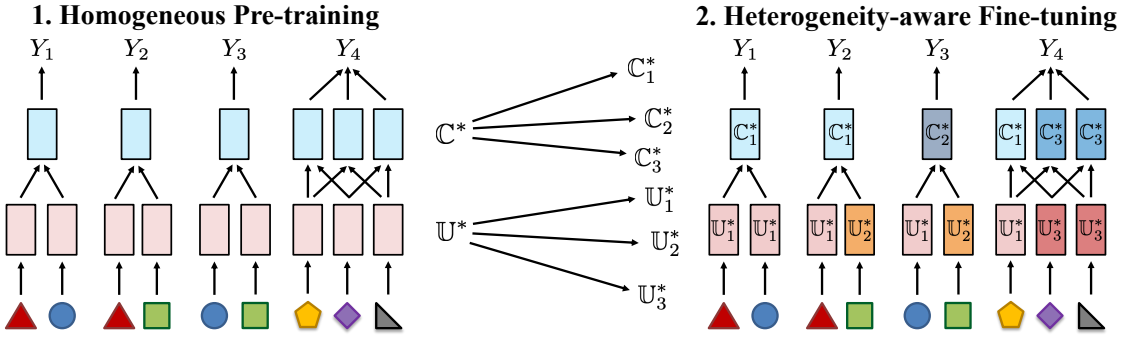

Figure 3: **HighMMT training** involves 2 steps: (1) *homogeneous pre-training* of a fully shared model across all modalities, before (2) *heterogeneity-aware fine-tuning* of modality and interaction parameters in different groups to respect modality and interaction heterogeneity respectively.

too (Dosovitskiy et al., 2021). For tables, sets, and graphs we treat each element in the table/set/graph as an element in the sequence. The end result is a standardized input data format of dimension $\mathbf{x}_m \in \mathbb{R}^{n \times t_m \times d_m}$, where n is the common batch-size, t_m is a modality and task-specific input sequence length, and d_m is a modality and task-specific input dimension.

2. Modality-specific embedding and positional encoding. For each distinct modality $m \in M$ (which may appear across multiple tasks), we define a one-hot modality embedding $\mathbf{e}_m \in \mathbb{R}^{|M|}$, where $|M|$ is the total number of distinct modalities, to identify common modalities across different tasks for information sharing. We also introduce Fourier feature positional encodings $\mathbf{p}_m \in \mathbb{R}^{t_m \times d_{pm}}$, where d_{pm} is the positional encoding dimension, to capture temporal and positional information across each modality. For multimodal tasks where a common dimension is shared across time (e.g., videos/time series), we apply a common positional encoding to capture the common time dimension.

3. Shared unimodal networks. Given modality-specific embeddings and positional encodings, the final input representation can now be processed by a general unimodal encoder with parameters $\mathbb{U}$ via a Transformer-based Perceiver block (Jaegle et al., 2021b). The input layer query is first set with a latent $d_{LN} \times d_{LS}$ block, and the context is set as $\mathbf{x}_m$ concatenated with modality embeddings and positional encodings along the last dimension. Consequently, self-attention learns relationships between elements in each modality, resulting in unimodal contextualized representations $\mathbf{z}_m \in \mathbb{R}^{n \times d_{LN} \times d_{LS}}$.

4. Shared crossmodal networks. To learn multimodal representations, we use a shared Crossmodal Transformer block with parameters $\mathbb{C}$ (Tsai et al., 2019; Lu et al., 2019). Given 2 unimodal representations $\mathbf{z}_1$ and $\mathbf{z}_2$, a Crossmodal Transformer (CT) block uses crossmodal self-attention by setting the input layer query $Q = \mathbf{z}_1$ and keys and values $K, V = \mathbf{z}_2$ to learn attention from X_1 to X_2, and a separate block to capture the attention in the opposite direction. A Crossmodal Transformer block using $\mathbf{z}_1$ to attend to $\mathbf{z}_2$ (and vice-versa) results in a final multimodal representation $\mathbf{z}_{\text{mm}} = [\mathbf{z}_{1\to2}, \mathbf{z}_{2\to1}] = [\text{CT}(\mathbf{z}_1, \mathbf{z}_2), \text{CT}(\mathbf{z}_2, \mathbf{z}_1)]$. For tasks with more than 2 modalities, a Crossmodal block is applied for each pair of modalities before concatenating.

5. Task-specific classifier and multitask pre-training. Finally, on top of concatenated and multimodal representations $\mathbf{z}_{\text{mm}}$, we use a separate linear classification layer per task for task-specific prediction. To enable information sharing across modalities and tasks, homogeneous pre-training is performed across a diverse set of datasets in a multitask manner by optimizing a weighted sum of losses over tasks. The result is a single set of shared unimodal parameters $\mathbb{U}^*$ that encodes all modalities, and a single set of shared crossmodal parameters $\mathbb{C}^*$ that captures all pairwise interactions between modality pairs, along with all modality-specific embeddings $\mathbb{E}^*$ and task-specific classifiers $\mathbb{T}^*$.

Heterogeneity-aware fine-tuning. Finally, we account for heterogeneity by grouping unimodal parameters based on modalities that we know to be similar from §2.1 (e.g., setting $\mathbb{U}_1 = \{U_1, U_2\}, \mathbb{U}_2 = \{U_3\}, \mathbb{U}_3 = \{U_4, U_5, U_6\}$), and likewise for the crossmodal parameters (e.g., $\mathbb{C}_1 = \{C_{12}, C_{13}, C_{14}\}, \mathbb{C}_2 = \{C_{23}, C_{15}\}, \mathbb{C}_3 = \{C_{24}, ...\}$). These groups of parameters are first initialized with the homogeneous model $\mathbb{U}^*$ and $\mathbb{C}^*$ before separate fine-tuning, which results in final parameters $\mathbb{U}^* \to \{\mathbb{U}_1^*, \mathbb{U}_2^*, ...\}$ and $\mathbb{C}^* \to \{\mathbb{C}_1^*, \mathbb{C}_2^*, ...\}$. The modality

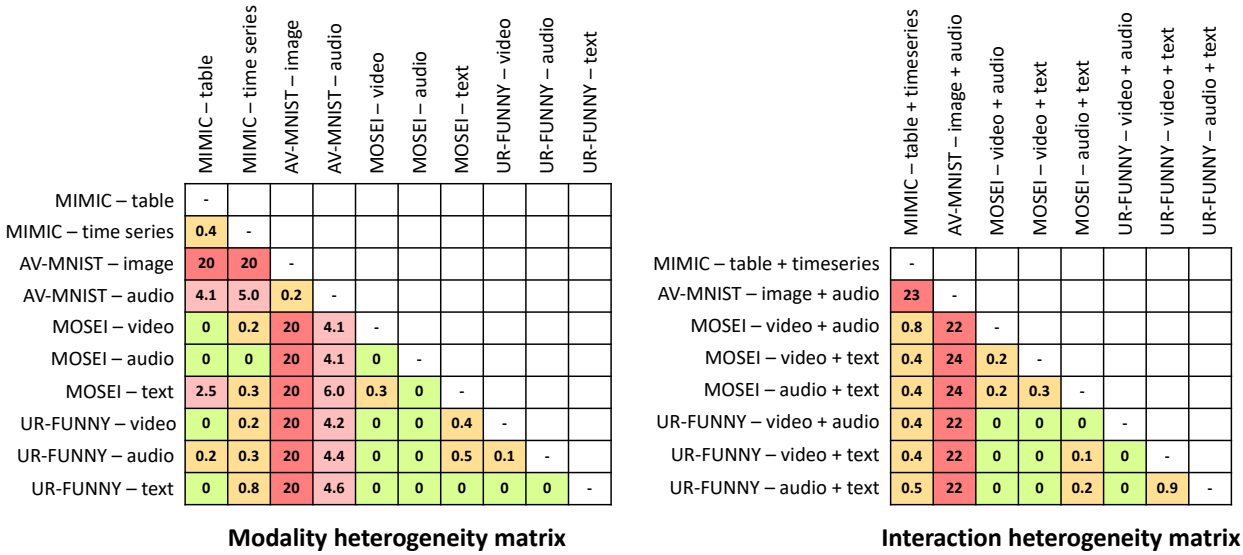

Figure 4: Modality and interaction heterogeneity matrices color coded by distances. We find clear task outliers (AV-MNIST only transfer well to each other and has high difficulty transferring to other modalities from the other datasets). Otherwise, the same modality across different tasks is generally similar to each other (e.g., text between AV-MNIST and MOSEI, audio between AV-MNIST and MOSEI). There is generally more interaction heterogeneity than unimodal, implying that while the modality features are similar, the crossmodal interactions between modality pairs are more unique. We also find that the same modality pairs (video+text) and (video+audio) shows crossmodal similarity across both datasets they appear in: MOSEI and UR-FUNNY.

embeddings $\mathbb{E}^*$ and task classifiers $\mathbb{T}^*$ are jointly fine-tuned as well. Fine-tuning is also performed in a multitask manner by optimizing a weighted sum of supervised losses across all modalities and tasks.

3 Experiments

Setup: In this section, we design experiments to analyze the multitask, transfer, and generalization capabilities of HighMMT. We use a large collection of multimodal datasets provided in MultiBench (Liang et al., 2021b) spanning 10 modalities, 15 prediction tasks, and 5 research areas. We trained 3 multitask models across combinations of these datasets (see Table 6 in Appendix C for details). Overall, the total size of datasets involved in our experiments exceeds $370,000$ and covers diverse modalities such as images, video, audio, text, time-series, various robotics sensors, sets, and tables, prediction tasks spanning the prediction of matching images and captions, robot pose, object pose, robot contact, design interfaces, digits, humor, sentiment, emotions, mortality rate, and ICD-9 codes, as well as multiple research areas of affective computing, healthcare, multimedia, robotics, and HCI.

3.1 Heterogeneity Measurements and Parameter Groups

We begin with a study of the heterogeneity measurements (see modality and interaction heterogeneity matrices in Figure 4) and the resulting parameter groups.

Modality heterogeneity: We first notice that the modalities from AV-MNIST only transfer well to each other and has high difficulty transferring to other modalities from the other datasets. The same modality across different tasks is generally similar to each other (e.g., text between UR-FUNNY and MOSEI, audio between UR-FUNNY and MOSEI). The text modality in UR-FUNNY seems to be close to most other modalities, and likewise for the tabular modality in MIMIC. It is also worth noting that the video and audio modalities are not the most informative in MOSEI, and predictions are dominated by language (Zadeh et al., 2017), which may explain their general homogeneity with respect to other modalities. Using these measurements, the final groups of modalities (and therefore unimodal parameters) we obtain after clustering are: $\mathcal{U}_1$ = {MIMIC table, MOSEI video, UR-FUNNY video}, $\mathcal{U}_2$ = {AV-MNIST image, AV-MNIST

audio}, $\mathcal{U}_3$ = {MIMIC timeseries, MOSEI text, UR-FUNNY text}, and $\mathcal{U}_4$ = {MOSEI audio, UR-FUNNY audio}.

Interaction heterogeneity: At a high level, there is generally more interaction heterogeneity than unimodal, implying that while the modality features are similar, the crossmodal interactions between modality pairs are more unique. Again, we notice the general poor transfer from the modality pair (image+audio) in AV-MNIST to all other pairs, and the general strong transfer from (audio+text) in UR-FUNNY to the rest, which shows a higher-order relationship between modality and interaction heterogeneity. We also find that the same modality pairs (video+text) and (video+audio) shows crossmodal similarity across both datasets they appear in: MOSEI and UR-FUNNY. Finally, while the triplet of crossmodal pairs in MOSEI are quite different from each other, those in UR-FUNNY are more similar. Using these measurements, the final groups of crossmodal pairs (and therefore crossmodal parameters) we obtain after clustering are: $\mathcal{C}_1$ = {MIMIC table+timeseries, MOSEI video+text, UR-FUNNY video+text}, $\mathcal{C}_2$ = {AV-MNIST image+audio}, $\mathcal{C}_3$ = {MOSEI video+audio}, and $\mathcal{C}_4$ = {MOSEI audio+text, UR-FUNNY video+audio, UR-FUNNY audio+text}.

3.2 Qualitative Results

We now present our results on the multitask, transfer, and generalization capabilities of HighMMT using performance and efficiency metrics. Henceforth, we will refer to the following models:

(1) **HighMMT share none** refers to individual copies of HighMMT models, one for each task.

(2) **HighMMT share all** refers to one single HighMMT model fully shared across all modalities and tasks.

(3) **HighMMT** refers to the full heterogeneity-aware HighMMT model across all modalities and tasks with learned parameter groupings based on heterogeneity measurements.

Multitask performance and efficiency. In Figure 5, we summarize the overall tradeoff between performance and efficiency using existing task-specific models and variants of HighMMT. The blue dots represent all possible combinations of task-specific models across multiple datasets (summarized in MultiBench (Liang et al., 2021b), $> 10^5$ total combinations) with their overall performance (scaled to a $0-1$ range before averaging across datasets) and overall efficiency (inverted total number of parameters). The red dots represent the state-of-the-art Pareto front: points that are not strictly dominated in both performance and efficiency. In light green, separate single-task HighMMT models (share none) already improve parameter efficiency as compared to standard Multimodal Transformers (Lu et al., 2019; Tsai et al., 2019). In dark green is HighMMT (share all) trained in a homogeneous multitask manner (i.e., with full parameter sharing across unimodal and multimodal layers within and across tasks), which further pushes forward the Pareto front by improving both performance and efficiency. Finally, in orange, HighMMT with heterogeneity-aware fine-tuning achieves significantly better tradeoffs between performance and efficiency, with a controlled increase in parameters but much higher increases in performance across multiple modalities and tasks. The suite of HighMMT models is obtained by tuning k, the number of parameter groups (i.e., number of clusters when clustering heterogeneity matrices).

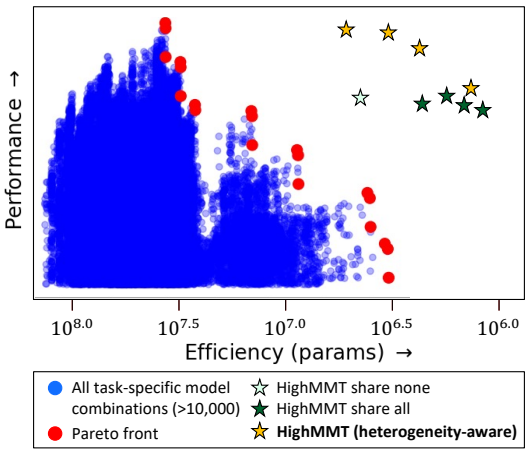

Figure 5: **Overall tradeoff.** HighMMT pushes forward the Pareto front of performance and efficiency as compared to all possible ($> 10^5$) combinations of task-specific models across multiple datasets (Liang et al., 2021b). The x-axis denotes (inverted) total parameters and y-axis denotes performance scaled to a $0-1$ range before averaging across datasets.

Positive transfer to new modalities and tasks. HighMMT also offers opportunities to study whether we can *transfer* knowledge between completely different modalities and tasks. We pre-trained a fully-shared HighMMT model on 1/2/3 of the 4 tasks before fine-tuning on the fourth task only (e.g., train on

Table 1: **Cross-modal transfer to new modalities and tasks.** We train a fully-shared multitask HighMMT on 1/2/3 datasets and find that it generalizes to new modalities and tasks on the 4th dataset, with improved performance over single-task training on the 4th dataset. Cross-modal transfer improves with more pretraining tasks and works best on the smallest target tasks (UR-FUNNY).

# Source tasks	Target task			
	UR-FUNNY	MOSEI	MIMIC	AV-MNIST
0 (no transfer)	63.3	79.4	67.7	70.4
1	64.1	79.4	68.3	70.4
2	65.5	80.0	68.5	70.5
3	**65.7**	**80.5**	**68.5**	**70.5**

Table 2: HighMMT achieves strong performance on overall performance and efficiency, sometimes even beating (shown in **bold**) the task-specific state-of-the-art, especially on the relatively understudied modalities (time-series, robotics sensors, and sets) from the robotics (Push, V&T) HCI (ENRICO), and healthcare (MIMIC) research areas, while using **10× fewer parameters** due to parameter sharing and multitask learning. SOTA captures the max performance and parameters of more than 20 recent multimodal models implemented in MultiBench (Liang et al., 2021b).

Model	ENRICO ↑	Push ↓	V&T ↑	UR-FUNNY ↑	MOSEI ↑	MIMIC ↑	AV-MNIST ↑	Params (M) ↓
SOTA	51.0 ± 1.4	0.290 ± 0.1	93.6 ± 0.1	$\mathbf{66.7 \pm 0.3}$	82.1 ± 0.5	$\mathbf{68.9 \pm 0.5}$	72.8 ± 0.2	32.3
HighMMT	$\mathbf{52.7 \pm 0.6}$	$\mathbf{0.277 \pm 0.1}$	$\mathbf{96.3 \pm 0.2}$	$\mathbf{66.2 \pm 0.4}$	80.2 ± 0.2	$\mathbf{68.2 \pm 0.3}$	71.1 ± 0.2	**3.01**

UR-FUNNY, MOSEI, MIMIC and transfer to AV-MNIST). From Table 1, we found that on all four combinations of multitask pretraining and fine-tuning, weights learned from other multimodal tasks generalize well to new modalities and tasks, improving performance over single target-task training. When we increase the number of pretraining datasets, we observe a consistent improvement in fine-tuned target task performance. There is an inverse correlation between target task size and performance improvement: the smallest dataset, UR-FUNNY, benefited the most (+2.4%) from transfer learning from 0 to 3 multitask datasets. This implies that our multimodal pretraining-fine-tuning paradigm is useful for low-resource target modalities and tasks. Finally, we compare transfer learning performance across different levels of partial observability. While one would expect the transfer to MIMIC to be the hardest due to its modality set $\{\text{time-series}, \text{table}\}$ being completely disjoint from the remaining 3 datasets, we still observe a +0.8% gain as compared to single-task training. Therefore, HighMMT can generalize to new modalities and tasks. Unsurprisingly, for datasets with more overlap (e.g., UR-FUNNY with complete overlap in $\{\text{text}, \text{video}, \text{audio}\}$ with respect to pretraining), we find larger improvements using transfer learning over single-task models (+2.4%).

Comparison with task-specific state-of-the-art. In Table 2, we compare multitask performance and efficiency with task-specific state-of-the-art models. We achieve performance within the range of published models (and usually close to the individual task-specific state-of-the-art) in MultiBench, which tallies more than 20 recent multimodal models in each task's literature (Liang et al., 2021b). In fact, HighMMT even sets new state-of-the-art results on several datasets, especially on the relatively understudied modalities (time-series, force and proprioception sensors, and sets) from the robotics (Push, V&T) and HCI (ENRICO) research areas. On top of strong performance, the main benefit lies in using fewer total parameters as compared to separate task-specific models - more than 10× reduction. Since this reduction grows with the number of tasks, our approach is scalable to high-modality scenarios.

Partial-observability. Observe HighMMT performance on partially-observable modality subsets (i.e., target task involving modalities not present in the other tasks): from Table 2, we find that the model performs well on the MIMIC dataset despite its modality set $\{\text{time-series}, \text{table}\}$ being completely disjoint from the remaining 3 datasets - we obtain similar performance across both multitask and single-task models ($68.2 \pm 0.3\%$ vs $68.9 \pm 0.5\%$). We find that HighMMT multitask also works on ENRICO dataset in the HCI domain ($52.7 \pm 0.6\%$ multitask vs $51.0 \pm 1.4\%$ single-task) despite it having completely disjoint modality inputs.

Multitask fusion and retrieval. We perform multitask training over multimodal fusion in AV-MNIST and retrieval in CIFAR-ESC. While fusion emphasizes information integration from complementary data sources, retrieval focuses on aligning corresponding elements expressed through different views of the data (Liang et al., 2022). Even across these vastly different prediction tasks, we find that multitask training (60.5% retrieval

Table 3: We conduct in-depth **ablation studies** on the architecture design, parameter sharing settings, and fine-tuning strategies in HighMMT, and find strong evidence for (1) having separate unimodal and interaction architecture layers, (2) determining parameter sharing via feature transfer rather than having parameters fully separate, fully shared, or computed via feature difference across modalities and tasks, and (3) homogeneous pre-training before heterogeneity-aware fine-tuning into parameter groups rather than directly training for heterogeneity.

	Model	UR-FUNNY ↑	MOSEI ↑	MIMIC ↑	AV-MNIST ↑	Ave ↑
	HighMMT	**66.3**	**80.2**	**68.5**	**71.3**	**71.6**
Architecture ablations	- w/o embeddings	62.5	78.4	67.9	69.5	69.6
	- w/o unimodal	57.6	61.8	63.0	59.1	60.4
	- w/o crossmodal	61.3	80.3	67.7	69.4	69.7
Param sharing ablations	- share none	63.4	79.7	68.5	69.0	70.2
	- share unimodal	63.5	79.5	65.3	70.0	69.6
	- share crossmodal	64.6	79.9	65.4	69.3	69.9
	- share all	63.0	79.9	67.8	70.4	70.3
	- random difference	62.4	79.5	67.6	70.4	70.4
	- feature difference	63.0	79.7	68.1	70.5	70.3
Training ablations	- direct training	61.2	78.5	64.8	71.1	69.9

accuracy) improves upon single-task training (58.8%). Not only have the unimodal networks simultaneously processed different modalities, but the crossmodal network has also learned to capture correspondences useful for both fusion and retrieval.

3.3 Ablation Studies

In this subsection, we carefully ablate each part of the model, between the model architectures, various ways of performing parameter sharing, and training decisions.

Architectural ablations. We first analyze each architectural component of HighMMT: (1) *w/o embeddings* removes the only modality-specific component in the model - the modality embeddings. We set embeddings for all modalities to be the same to test whether a modality-specific component is necessary to capture heterogeneity across input data sources, (2) *w/o unimodal* removes the unimodal encoder and directly applies the cross-attention layer, and *w/o crossmodal* replaces the crossmodal layer with a concatenation of unimodal features and a linear classification layer. The latter resembles the most direct multimodal extension of existing work in shared unimodal encoders like Perceiver (Jaegle et al., 2021b), MultiModel (Kaiser et al., 2017), ViT-BERT (Li et al., 2021) or PolyViT (Likhosherstov et al., 2022). From Table 3, removing any of the 3 components in HighMMT results in worse performance. The unimodal encoder is particularly important for best performance.

Param sharing ablations. We further ablate with respect to possible parameter sharing settings in HighMMT. Beyond fully single-task and multitask variants, (1) *share none* uses separate unimodal and multimodal layers reminiscent of typical single-task multimodal transformers (Tsai et al., 2019; Lu et al., 2019; Hendricks et al., 2021), (2-3) *share unimodal (crossmodal)* only shares the unimodal (crossmodal) layer during multitask training, (4) *share all* shares all parameters without accounting for possible heterogeneity (Reed et al., 2022), (5) *random difference* determines k parameter groups randomly rather than via heterogeneity measurements, (6) *feature difference* is a simple baseline using feature-level divergences on jointly trained unimodal encoders (i.e., $\|U(X_1) - U(X_2)\|_2^2$) rather than transfer performance to measure heterogeneity as is commonly done in transfer learning and domain adaptation (Daumé III, 2007; Sun et al., 2016). From Table 3, using separate parameters for unimodal or multimodal layers also decreases performance, which implies that parameter sharing learns improved generalizable representations. Furthermore, our proposed heterogeneity-aware parameter grouping results in the best overall performance as compared to fully shared, fully separate, or parameter grouping informed by other heterogeneity measures such as random or feature distance.

Training ablations. Finally, we explore *direct training* of the learned parameter groups as opposed to performing homogeneous pre-training before fine-tuning them into parameter groups. From Table 3, we find that this ablation underperforms - training multiple parameter groups from scratch overfits quickly to smaller datasets which hurts overall generalization performance.

3.4 Understanding homogeneity and heterogeneity in HighMMT

The above subsections have demonstrated the strong abilities of HighMMT in multitask, transfer, and few-shot learning settings across diverse modalities and tasks. We now take a deeper empirical analysis to better understand the extent of homogeneity and heterogeneity captured by HighMMT. To do so, we design two experiments investigating parameter overlap and parameter interference in a fully trained HighMMT model. We also investigate several other model properties and visualizations in Appendix D.4.

Parameter overlap. We first investigate the extent of parameter overlap in a fully shared HighMMT model across modalities and tasks. Starting with a trained multitask HighMMT model, we use a gradient-based analysis method (in the same vein of studying gradients to look at prediction influence (Han et al., 2020)) to determine how much each parameter is involved in a specific task. For each task T and parameter $\theta \in \Theta$ in multitask model M_Θ, we compute the involvement $I_T(\theta) = \mathbb{E}_{(x,y)\epsilon T}|\nabla_\theta M_\Theta(y|x)|$ where $M_\Theta(y|x)$ is the predicted probability of correct target y by M_Θ given x as input. In other words, this measures the absolute gradient with respect to θ when predicting y given x in task T. A higher absolute gradient implies "activated" neurons and vice-versa for gradients closer to 0. This enables us to compute the extent a parameter θ is involved for each task. The *number of tasks* a given parameter θ is involved in can then be approximated by thresholding and summing up $n(\theta) = \sum_T (\mathbb{1}\{I_T(\theta) > \epsilon \max(I_1(\theta), I_2(\theta), I_3(\theta), I_4(\theta)\})$ which returns an integer from 1 to 4. We chose a threshold ϵ such that parameters are classified as active about half the time on average, which occurs at $\epsilon = 0.2$.

Since we are interested in the level of parameter overlap in the shared unimodal encoder and multimodal layer, we set θ as these 2 modules and report results in Table 4. There is evidence of significant parameter overlap across unimodal encoders: more than 92% of neurons are involved in at least 3 of the 4 tasks. On the other hand, there is not nearly as much parameter overlap in the multimodal layer: only 10% of neurons are involved in 3 or 4 tasks. Hence, it seems like the shared unimodal encoders learn task-agnostic representations, but the subsequent multimodal layers (closer to task-specific classifiers) capture more task-specific information. This also reinforces our observation in §3.1 that there is generally more interaction heterogeneity than modality heterogeneity, which suggests using fewer unimodal parameter groups and more crossmodal parameter groups.

Table 4: We find evidence of significant **parameter overlap** across unimodal encoders: > 92% of neurons are involved in at least 3 of the 4 tasks. On the other hand, the multimodal layers are more task-specific: only 10% of neurons are involved in 3 or 4 tasks.

Component	Number of involved tasks			
	1	2	3	4
Unimodal layers	2.8%	5.1%	**61.1**%	**31.1**%
Crossmodal layers	**48.8**%	**39.7**%	9.9%	1.6%

Parameter interference. Another empirical proof for parameter sharing in multitask models is the phenomenon of *parameter interference*: to what extent do parameters across tasks interfere with each other? Parameter interference is inspired by catastrophic forgetting (French, 1999; Toneva et al., 2018) - where pre-trained models fine-tuned on a new dataset tend to 'forget' how to perform on their pre-trained datasets due to the same group of parameters interfering with each other as the task changes. We perform a similar experiment to investigate parameter interference: we pick one task and flip the labels in its training set, train the multitask model on the modified training set, and see how the incorrectly labeled task affects performance on other tasks. This experiment provides evidence of information sharing: if the multitask model does not share information (i.e., the model learns independent subspaces for each task), then one would not observe a negative interference phenomenon from one noisy dataset. We study negative interference under 3 configurations of training (a) the whole model; (b) only the unimodal encoder, and (c) only the multimodal layer on the flipped training set.

From Table 5, we observe that certain tasks are more affected by negative interference (e.g., AV-MNIST), while some tasks are not influenced as much (e.g., UR-FUNNY). Again, this exactly reflects our heterogeneity measurements in §3.1, where we find that AV-MNIST displays high heterogeneity to other modalities and tasks from the other datasets. Furthermore, we observe that performance drops due to training the shared unimodal encoders are the most significant, which corroborates with our parameter overlap and heterogeneity analysis that general unimodal encoders contain more entangled parameters which are more sensitive to task changes. On the other hand, multimodal layers contain more disentangled parameters that share less information across tasks, which results in higher heterogeneity measurements and suitability for more separate parameter groups.

Table 5: **Parameter interference**: we observe different degrees of performance drops on each task (columns) after training on one task with flipped labels (rows). Training the shared unimodal encoders causes the most harm, which implies that unimodal encoders contain more shared neurons sensitive to task changes. Red for drops greater than 20%, yellow for drops between 10 and 20%, and green for drops below 10%.

(a) Training entire model

Flipped task	UR-FUNNY	MOSEI	MIMIC	AV-MNIST
UR-FUNNY	−24.6	−8.83	−10.6	−57.7
MOSEI	−4.07	−59.7	−20.3	−53.2
MIMIC	−3.59	−5.83	−33.1	−37.5
AV-MNIST	−3.50	−1.23	−4.87	−68.9

(b) Only training unimodal encoder

Flipped task	UR-FUNNY	MOSEI	MIMIC	AV-MNIST
UR-FUNNY	−23.8	−10.1	−12.8	−58.4
MOSEI	−5.77	−57.6	−21.1	−52.7
MIMIC	−3.03	−3.54	−35.0	−56.3
AV-MNIST	−2.94	−7.82	−53.6	−69.3

(c) Only training multimodal layer

Flipped task	UR-FUNNY	MOSEI	MIMIC	AV-MNIST
UR-FUNNY	−25.2	−8.34	−2.67	−8.16
MOSEI	0.47	−59.6	−19.8	−8.19
MIMIC	0.19	−0.76	−35.2	−4.87
AV-MNIST	−1.61	−1.48	−2.23	−69.1

4 Related Work

Multimodal Transformers have emerged as strong general-purpose models for representation learning. Building upon the initial text-based Transformer model (Vaswani et al., 2017), these multimodal extensions typically use either full self-attention over modalities concatenated across the sequence dimension (Li et al., 2019; Sun et al., 2019; Su et al., 2020; Chen et al., 2020) or a cross-modal attention layer (Lu et al., 2019; Tsai et al., 2019; Tan and Bansal, 2019), and are useful for sequential data by automatically aligning and capturing complementary features at different time-steps (Tsai et al., 2019; Yao and Wan, 2020; Lee et al., 2020c). Self-supervised multimodal pretraining has emerged as an effective way to train these powerful architectures, with the aim of learning general-purpose representations from larger-scale unlabeled multimodal data before transferring to specific downstream tasks via supervised fine-tuning (Lu et al., 2019; Li et al., 2019; Su et al., 2020). These pretraining objectives typically consist of unimodal masked prediction, crossmodal masked prediction, and multimodal alignment prediction (Hendricks et al., 2021).

Unified encoder for unimodal learning. Several works such as Perceiver (Jaegle et al., 2021a;b), MultiModel (Kaiser et al., 2017), ViT-BERT (Li et al., 2021), and PolyViT (Likhosherstov et al., 2022) have explored the possibility of using the same unimodal encoder architecture for different inputs on unimodal tasks (i.e., language, image, video, or audio-only). The Transformer architecture has emerged as a popular choice due to its suitability for serialized inputs such as text (Devlin et al., 2019), images (Dosovitskiy et al., 2021), video (Sun et al., 2019), and time-series data (Lim et al., 2021), a phenomenon further observed by Lu et al. (2021) where a single Transformer pretrained on text transfers to other unimodal tasks including sequence modeling and image classification. While these serve as building blocks in our model, our focus is on a general-purpose multimodal model for multitask and transfer learning across different subsets of modalities rather than unimodal tasks. We summarize some of these differences in Figure 6.

Multimodal multitask and transfer learning. There have also been several attempts to build a single model that works well on a suite of multimodal tasks (Li et al., 2019; Lu et al., 2019; Su et al., 2020; Cho et al., 2021; Reed et al., 2022). For example, UniT (Hu and Singh, 2021), VLBERT (Su et al., 2020), ViLBERT (Lu et al., 2019), and VL-T5 (Cho et al., 2021) are all unifying models for vision-and-language tasks, with some models also possessing generalization to vision-only and language-only tasks. VATT (Akbari et al., 2021) jointly trains a shared model on video, audio, and text data to perform audio-only, video-only,

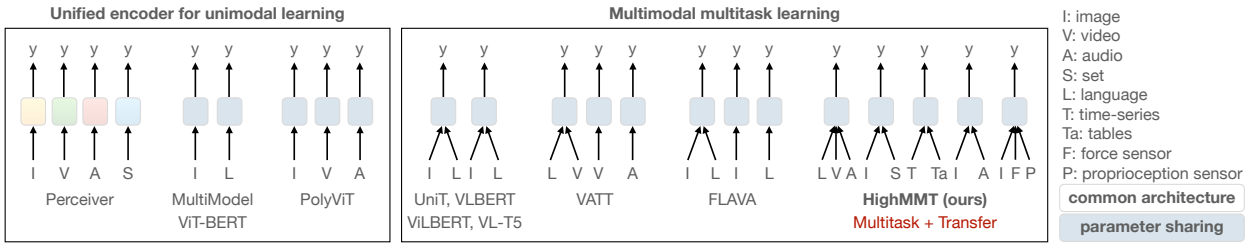

Figure 6: In contrast to current work in unified unimodal encoders (Jaegle et al., 2021b; Kaiser et al., 2017; Li et al., 2021; Likhosherstov et al., 2022) and multimodal multitask learning (Hu and Singh, 2021; Lu et al., 2019; Su et al., 2020; Cho et al., 2021; Akbari et al., 2021; Singh et al., 2021), we aim to design a general-purpose model for high-modality and partially-observable scenarios, which presents two main technical challenges: (1) *scalability*, since adding new parameters for every new modality/task becomes prohibitively expensive, and (2) *partial observability*, since each task is defined only over a small subset of all modalities we are interested in modeling.

and image-text retrieval tasks. FLAVA (Singh et al., 2021) found that pretraining a shared model with unpaired images, unpaired text, and image-text pairs results in strong performance on image-only, text-only, and image-text multimodal tasks, while Reed et al. (2022) scales up a single Transformer model for image, text, and decision-making tasks. However, all of these train a single model for all tasks, without investigating how varying degrees of modality and interaction heterogeneity can necessitate partial parameter sharing for the best tradeoff between performance and efficiency. On the transfer side, while more research has focused on transfer within the same modality with external information (Socher et al., 2013; Dunnmon et al., 2019; Xing et al., 2019; Zadeh et al., 2020), Liang et al. (2021c) is the only work that studies transfer to completely new modalities. However, they require paired data collection and modality-specific modeling for specific unimodal transfer tasks. Our work goes beyond the commonly studied language, vision, and audio modalities to relatively understudied ones (e.g., tabular data, time-series, sensors, graphs, and set data). Furthermore, we show the possibility of generalizing to new modality subsets using general models, which can further enable quick generalization to new domains and tasks. Finally, our work also complements existing empirical and theoretical studies of transfer learning in a single modality (Standley et al., 2020; Wu et al., 2022; Zamir et al., 2018), where insights from task heterogeneity have informed multitask approaches.

Multimodal benchmarks have emerged as a crucial component testing the generalization of models in multitask and transfer settings. Ideal benchmarks should reflect real-world distribution shifts between modalities, tasks, and research areas. In language and vision, these have consolidated into a suite of datasets spanning image-text retrieval (Young et al., 2014; Plummer et al., 2015), image-text question answering (Agrawal et al., 2017; Zellers et al., 2019; Kazemzadeh et al., 2014), and image-text fusion (Kiela et al., 2020; Vielzeuf et al., 2018; Kay et al., 2017; Arevalo et al., 2017). Beyond these two modalities, MultiBench is a large-scale benchmark with many modalities and tasks while evaluating generalization, complexity, and robustness (Liang et al., 2021b). Several benchmarks have also focused on a single research area such as affective computing (Gkoumas et al., 2021), human multimodal language (Zadeh, 2019), language and vision-based question answering (Ferraro et al., 2015; Sharif et al., 2020), text classification with multimodal information (Gu, 2020), and multimodal learning for education (Hassan et al., 2021).

5 Conclusion

In conclusion, this paper proposes an information transfer approach for estimating modality and interaction heterogeneity, a key component towards automatically determining which modalities should be processed and fused jointly for efficient representation learning in high-modality scenarios. Our resulting model, HighMMT dynamically determines the optimal parameter groupings balancing total performance and parameter efficiency, simultaneously achieves strong results on 10 modalities (text, image, video, audio, time-series, sensors, tables, and sets) and 15 tasks from 5 different research areas, and transfers to entirely new modalities and tasks during fine-tuning. We release our code and benchmarks which we hope will present a unified platform for subsequent theoretical and empirical analysis.

References

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. VQA: Visual question answering. *International Journal of Computer Vision*, 2017.

Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *arXiv preprint arXiv:2104.11178*, 2021.

Paras Malik Amisha, Monika Pathania, and Vyas Kumar Rathaur. Overview of artificial intelligence in medicine. *Journal of family medicine and primary care*, 8(7):2328, 2019.

John Arevalo, Thamar Solorio, Manuel Montes-y Gómez, and Fabio A González. Gated multimodal units for information fusion. In *5th International conference on learning representations 2017 workshop*, 2017.

Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443, 2018.

Jonathan Baxter. A bayesian/information theoretic model of learning to learn via multiple task sampling. *Machine learning*, 28(1):7–39, 1997.

Tony Belpaeme, James Kennedy, Aditi Ramachandran, Brian Scassellati, and Fumihide Tanaka. Social robots for education: A review. *Science robotics*, 3(21), 2018.

Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120. Springer, 2020.

Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. Unifying vision-and-language tasks via text generation. In *ICML*, 2021.

Pierre Colombo, Emile Chapuis, Matthieu Labeau, and Chloé Clavel. Improving multimodal fusion via mutual dependency maximisation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 231–245, 2021.

Hal Daumé III. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, 2007.

Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. Rico: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, pages 845–854, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*, 2019.

Alan Dix, Janet Finlay, Gregory D Abowd, and Russell Beale. Human-computer interaction. *Harlow ua*, 2000.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

Petros Drineas, Asif Javed, Malik Magdon-Ismail, Gopal Pandurangan, Reino Virrankoski, and Andreas Savvides. Distance matrix reconstruction from incomplete distance information for sensor network localization. In *2006 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, volume 2, pages 536–544. IEEE, 2006.

Jared Dunnmon, Alexander Ratner, Nishith Khandwala, Khaled Saab, Matthew Markert, Hersh Sagreiya, Roger E. Goldman, Christopher Lee-Messer, Matthew P. Lungren, Daniel L. Rubin, and Christopher Ré. Cross-modal data programming enables rapid medical machine learning. *CoRR*, abs/1903.11101, 2019. URL `http://arxiv.org/abs/1903.11101`.

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. Understanding dataset difficulty with $\mathcal{V}$-usable information. In *International Conference on Machine Learning*, pages 5988–6008. PMLR, 2022.

Francis Ferraro, Nasrin Mostafazadeh, Ting-Hao Huang, Lucy Vanderwende, Jacob Devlin, Michel Galley, and Margaret Mitchell. A survey of current datasets for vision and language research. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 207–213, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1021. URL https://www.aclweb.org/anthology/D15-1021.

Christos A Frantzidis, Charalampos Bratsas, Manousos A Klados, Evdokimos Konstantinidis, Chrysa D Lithari, Ana B Vivas, Christos L Papadelis, Eleni Kaldoudi, Costas Pappas, and Panagiotis D Bamidis. On the classification of emotional biosignals evoked while viewing affective pictures: an integrated data-mining-based approach for healthcare applications. *IEEE Transactions on Information Technology in Biomedicine*, 14(2):309–318, 2010.

Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.

Dimitris Gkoumas, Qiuchi Li, Christina Lioma, Yijun Yu, and Dawei Song. What makes the difference? an empirical comparison of fusion strategies for multimodal language analysis. *Information Fusion*, 66:184–197, 2021.

Ken Gu. Multimodal toolkit. https://github.com/georgian-io/Multimodal-Toolkit, 2020.

Xiaochuang Han, Byron C Wallace, and Yulia Tsvetkov. Explaining black box predictions and unveiling data artifacts through influence functions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5553–5563, 2020.

Md Kamrul Hasan, Wasifur Rahman, AmirAli Bagher Zadeh, Jianyuan Zhong, Md Iftekhar Tanveer, Louis-Philippe Morency, and Mohammed Ehsan Hoque. Ur-funny: A multimodal language dataset for understanding humor. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2046–2056, 2019.

Javaria Hassan, Jovin Leong, and Bertrand Schneider. *Multimodal Data Collection Made Easy: The EZ-MMLA Toolkit: A Data Collection Website That Provides Educators and Researchers with Easy Access to Multimodal Data Streams.*, page 579–585. Association for Computing Machinery, New York, NY, USA, 2021. ISBN 9781450389358. URL https://doi.org/10.1145/3448139.3448201.

Lisa Anne Hendricks, John Mellor, Rosalia Schneider, Jean-Baptiste Alayrac, and Aida Nematzadeh. Decoupling the role of data, attention, and losses in multimodal transformers. *arXiv preprint arXiv:2102.00529*, 2021.

Ronghang Hu and Amanpreet Singh. Transformer is all you need: Multimodal multitask learning with a unified transformer. *arXiv preprint arXiv:2102.10772*, 2021.

Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, et al. Perceiver io: A general architecture for structured inputs & outputs. *arXiv preprint arXiv:2107.14795*, 2021a.

Andrew Jaegle, Felix Gimeno, Andrew Brock, Andrew Zisserman, Oriol Vinyals, and Joao Carreira. Perceiver: General perception with iterative attention. *arXiv preprint arXiv:2103.03206*, 2021b.

Siddhant M. Jayakumar, Wojciech M. Czarnecki, Jacob Menick, Jonathan Schwarz, Jack Rae, Simon Osindero, Yee Whye Teh, Tim Harley, and Razvan Pascanu. Multiplicative interactions and where to find them. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rylnK6VtDH.

Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-Wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.

Lukasz Kaiser, Aidan N Gomez, Noam Shazeer, Ashish Vaswani, Niki Parmar, Llion Jones, and Jakob Uszkoreit. One model to learn them all. *arXiv preprint arXiv:1706.05137*, 2017.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798, 2014.

Douwe Kiela, Suvrat Bhooshan, Hamed Firooz, Ethan Perez, and Davide Testuggine. Supervised multimodal bitransformers for classifying images and text. *arXiv preprint arXiv:1909.02950*, 2019.

Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. The hateful memes challenge: Detecting hate speech in multimodal memes. *Advances in Neural Information Processing Systems*, 33, 2020.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Michelle A Lee, Yuke Zhu, Krishnan Srinivasan, Parth Shah, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette Bohg. Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8943–8950. IEEE, 2019.

Michelle A Lee, Brent Yi, Roberto Martín-Martín, Silvio Savarese, and Jeannette Bohg. Multimodal sensor fusion with differentiable filters. *IROS*, 2020a.

Michelle A Lee, Yuke Zhu, Peter Zachares, Matthew Tan, Krishnan Srinivasan, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette Bohg. Making sense of vision and touch: Learning multimodal representations for contact-rich tasks. *IEEE Transactions on Robotics*, 36(3):582–596, 2020b.

Sangho Lee, Youngjae Yu, Gunhee Kim, Thomas Breuel, Jan Kautz, and Yale Song. Parameter efficient multimodal transformers for video representation learning. In *International Conference on Learning Representations*, 2020c.

Luis A Leiva, Asutosh Hota, and Antti Oulasvirta. Enrico: A dataset for topic modeling of mobile ui designs. In *22nd International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI'20 Extended Abstracts)*, 2020.

R Gary Leonard and George Doddington. Tidigits speech corpus. *Texas Instruments, Inc*, 1993.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019.

Qing Li, Boqing Gong, Yin Cui, Dan Kondratyuk, Xianzhi Du, Ming-Hsuan Yang, and Matthew Brown. Towards a unified foundation model: Jointly pre-training transformers on unpaired images and text. *arXiv preprint arXiv:2112.07074*, 2021.

Yaoyiran Li, Edoardo Maria Ponti, Ivan Vulić, and Anna Korhonen. Emergent communication pretraining for few-shot machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4716–4731, 2020.

Paul Pu Liang, Terrance Liu, Anna Cai, Michal Muszynski, Ryo Ishii, Nicholas Allen, Randy Auerbach, David Brent, Ruslan Salakhutdinov, and Louis-Philippe Morency. Learning language and multimodal privacy-preserving markers of mood from mobile data. In *ACL/IJCNLP (1)*, 2021a.

Paul Pu Liang, Yiwei Lyu, Xiang Fan, Zetian Wu, Yun Cheng, Jason Wu, Leslie Yufan Chen, Peter Wu, Michelle A Lee, Yuke Zhu, et al. Multibench: Multiscale benchmarks for multimodal representation learning. In *NeurIPS Datasets and Benchmarks Track*, 2021b.

Paul Pu Liang, Peter Wu, Liu Ziyin, Louis-Philippe Morency, and Ruslan Salakhutdinov. Cross-modal generalization: Learning in low resource modalities via meta-alignment. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 2680–2689, 2021c.

Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. Foundations and recent trends in multimodal machine learning: Principles, challenges, and open questions. *arXiv preprint arXiv:2209.03430*, 2022.

Valerii Likhosherstov, Mostafa Dehghani, Anurag Arnab, Krzysztof Marcin Choromanski, Mario Lucic, Yi Tay, and Adrian Weller. Polyvit: Co-training vision transformers on images, videos and audio, 2022. URL `https://openreview.net/forum?id=9r4_7GxTLnS`.

Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 2021.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pages 13–23, 2019.

Kevin Lu, Aditya Grover, Pieter Abbeel, and Igor Mordatch. Pretrained transformers as universal computation engines. *arXiv preprint arXiv:2103.05247*, 2021.

George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November 1995. ISSN 0001-0782.

Isabel Papadimitriou and Dan Jurafsky. Learning music helps you read: Using transfer to study linguistic structure in language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6829–6839, 2020.

Rosalind W Picard. *Affective computing*. MIT press, 2000.

Karol J Piczak. Esc: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1015–1018, 2015.

Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *arXiv preprint arXiv:2102.12092*, 2021.

Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, et al. A generalist agent. *arXiv preprint arXiv:2205.06175*, 2022.

Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

Naeha Sharif, Uzair Nadeem, Syed Afaq Ali Shah, Mohammed Bennamoun, and Wei Liu. Vision to language: Methods, metrics and datasets. In *Machine Learning Paradigms*, pages 9–62. Springer, 2020.

Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. *arXiv preprint arXiv:2112.04482*, 2021.

Richard Socher, Milind Ganjoo, Hamsa Sridhar, Osbert Bastani, Christopher D Manning, and Andrew Y Ng. Zero-shot learning through cross-modal transfer. *arXiv preprint arXiv:1301.3666*, 2013.

Trevor Standley, Amir Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? In *International Conference on Machine Learning*, pages 9120–9132. PMLR, 2020.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=SygXPaEYvH`.

Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

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

Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111, 2019.

Abiy Tasissa and Rongjie Lai. Exact reconstruction of euclidean distance geometry problem using low-rank matrix completion. *IEEE Transactions on Information Theory*, 65(5):3124–3144, 2018.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *ECCV*, 2020.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv e-prints*, pages physics–0004057, 2000.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. An empirical study of example forgetting during deep neural network learning. In *International Conference on Learning Representations*, 2018.

Christopher Tosh, Akshay Krishnamurthy, and Daniel Hsu. Contrastive learning, multi-view redundancy, and linear models. In *Algorithmic Learning Theory*, pages 1179–1206. PMLR, 2021.

Yao-Hung Hubert Tsai, Shaojie Bai, Paul Pu Liang, J Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. Multimodal transformer for unaligned multimodal language sequences. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6558–6569, 2019.

Michael Tschannen, Josip Djolonga, Paul K Rubenstein, Sylvain Gelly, and Mario Lucic. On mutual information maximization for representation learning. In *International Conference on Learning Representations*, 2019.

Endel Tulving and Michael J Watkins. On negative transfer: Effects of testing one list on the recall of another. *Journal of Verbal Learning and Verbal Behavior*, 13(2):181–193, 1974.

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

Matias Vera, Pablo Piantanida, and Leonardo Rey Vega. The role of the information bottleneck in representation learning. In *2018 IEEE International Symposium on Information Theory (ISIT)*, pages 1580–1584. IEEE, 2018.

Valentin Vielzeuf, Alexis Lechervy, Stéphane Pateux, and Frédéric Jurie. Centralnet: a multilayer approach for multimodal fusion, 2018.

Zirui Wang, Zihang Dai, Barnabás Póczos, and Jaime Carbonell. Characterizing and avoiding negative transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11293–11302, 2019.

Xuetong Wu, Jonathan H Manton, Uwe Aickelin, and Jingge Zhu. An information-theoretic analysis for transfer learning: Error bounds and applications. *arXiv preprint arXiv:2207.05377*, 2022.

Yuhuai Wu, Markus N. Rabe, Wenda Li, Jimmy Ba, Roger B. Grosse, and Christian Szegedy. LIME: learning inductive bias for primitives of mathematical reasoning. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 11251–11262. PMLR, 2021. URL `http://proceedings.mlr.press/v139/wu21c.html`.

Chen Xing, Negar Rostamzadeh, Boris Oreshkin, and Pedro O O. Pinheiro. Adaptive cross-modal few-shot learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*. 2019. URL `http://papers.nips.cc/paper/8731-adaptive-cross-modal-few-shot-learning.pdf`.

Yilun Xu, Shengjia Zhao, Jiaming Song, Russell Stewart, and Stefano Ermon. A theory of usable information under computational constraints. In *International Conference on Learning Representations*, 2019.

Shaowei Yao and Xiaojun Wan. Multimodal transformer for multimodal machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.400. URL `https://www.aclweb.org/anthology/2020.acl-main.400`.

De Jong Yeong, Gustavo Velasco-Hernandez, John Barry, Joseph Walsh, et al. Sensor and sensor fusion technology in autonomous vehicles: A review. *Sensors*, 21(6):2140, 2021.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. doi: 10.1162/tacl_a_00166. URL `https://aclanthology.org/Q14-1006`.

Amir Zadeh. CMU multimodal SDK. `https://github.com/A2Zadeh/CMU-MultimodalSDK`, 2019.

Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. Tensor fusion network for multimodal sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1103–1114, 2017.

Amir Zadeh, Paul Pu Liang, and Louis-Philippe Morency. Foundations of multimodal co-learning. *Information Fusion*, 64:188–193, 2020.

AmirAli Bagher Zadeh, Paul Pu Liang, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph. In *ACL*, 2018.

Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3712–3722, 2018.

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

Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

Appendix

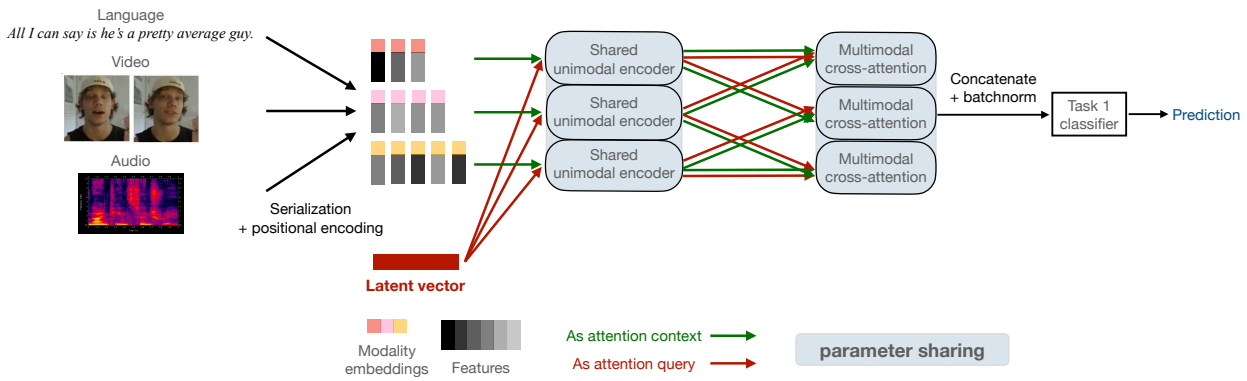

Figure 7: General architecture of HighMMT: Given arbitrary modalities, (1) the inputs are standardized into a sequence and padded, (2) modality embeddings and positional encodings are added to the serialized raw input, (3) a single shared unimodal Perceiver encoder is applied to all modalities to learn general-purpose representations regardless of the specific input modality, (4) each pair of unimodal representations is fed through a shared multimodal cross-attention layer twice (the first time with one modality as query and the other as context, and the second time vice versa) to learn general multimodal representations regardless of the input modalities and task, and finally (5) all outputs from cross-attention layers are concatenated, batch-normalized, and fed through a task-specific classification head to make a prediction. The unimodal encoders and multimodal layers are shared across tasks during multitask learning to enable statistical strength sharing, parameter efficiency, and quick generalization across diverse modalities and tasks.

A Measuring Heterogeneity via Modality Information Transfer

A.1 Modality and interaction heterogeneity matrix

We construct a modality heterogeneity matrix $M_U(i,j) = d(X_i; X_j)$ and an interaction heterogeneity matrix (technically 4D-tensor) $M_C(i,j,k,\ell) = d(X_i, X_j; X_k, X_\ell)$. As a side note, observe that these matrices are highly structured due to distances satisfying the triangle inequality, which implies that we do not need to compute all entries and instead rely on low-rank reconstruction from partial entries in practice (Drineas et al., 2006; Tasissa and Lai, 2018). For example, we can approximate the modality heterogeneity matrix $M_U = \sum_{i=1}^{h} u_i v_i^\top$ as an outer product of k individual basis vectors u_i and v_i, where h is a smaller number than the actual dimension of M_U.

A.2 Determining parameter groupings

We balance both total performance and parameter efficiency via agglomerative hierarchical clustering where modalities are nodes and heterogeneity measurements are edges. The number of clusters k is treated as a hyperparameter dependent on the parameter budget. Clustering on the modality heterogeneity matrix M_U results in a grouping of modalities based on similarity (e.g., $\mathcal{U}_1 = \{X_1, X_2, X_4\}, \mathcal{U}_2 = \{X_3\}, \mathcal{U}_3 = \{X_5\}$), and likewise for the interaction matrix M_C (e.g., $\mathcal{C}_1 = \{\{X_1, X_2\}, \{X_1, X_3\}, \{X_4, X_5\}\}, \mathcal{C}_2 = \{\{X_2, X_3\}, \mathcal{C}_3 = \{\{X_4, X_6\}, \{X_5, X_6\}\}$, and so on. How can we choose the number of clusters k? Note that if k is equal to the total number of modalities (or modality pairs) then it reduces to having separate models for each modality (and interaction), while $k = 1$ implies using a single model for all modalities and interactions. k is therefore most suitably seen as a 'parameter budget' that one would like to control for efficiency. In our experiments, we explored a range of k giving rise to a suite of models across controlled trade-offs between performance and efficiency (see Figure 5).

B HighMMT Details

At a high level, HighMMT includes the following components: (1) the inputs are standardized into a sequence and padded, (2) the perceiver input processing adds modality-specific modality embeddings and positional encodings to the serialized raw input; (3) the processed input from each modality is fed into a shared unimodal perceiver encoder; (4) each pair of unimodal perceiver output (unimodal representations) is fed through a shared crossmodal transformer layer twice (the first time with one modality as query and the other as context, and the second time vice versa); (5) finally, all outputs from multimodal layers are concatenated, batch-normalized to form a multimodal representation, and fed through a task-specific classification head to make a prediction. Figure 7 is an illustration of the high-level architecture.

B.1 Perceiver Input Processing

We follow the data processing pipeline in the GitHub implementation for multimodal perceivers: `https://github.com/fac2003/perceiver-multi-modality-pytorch`. For each modality, we must specify in advance the channel size (i.e., embedding size) and how many extra dimensions there are other than the channel/embedding dimension.

The modality embedding is just a one-hot vector denoting the index of the current modality, and the size of the vector is equal to the total number of modalities involved. This embedding layer identifies common modalities across different tasks to enable sharing of information. For example, the modality embedding of the image sequence for a video classification task will be shared with that of an input (static) image for an image and text question-answering task.

We also specify a few hyperparameters (such as num_freq_bands and max_freq) for the Fourier transformation used in the positional encoding. The positional encoding represents where this embedding is at through Fourier transformations (so if there is 1 extra dimension, then the positional encoding will encode the 1D position of each embedding; if there are 2 extra dimensions, then the positional encoding will encode the 2D position of each embedding). The positional encoding length can vary for each modality depending on the number of extra dimensions and the Fourier transformation hyperparameters.

The total embedding size of the processed output will be equal to $d_{all} = \max_{m \in M}(d_m + d_{pm} + |M|)$, where M is the set of all modalities involved, d_m is the channel size of modality m, d_{pm} is the positional encoding size of modality m, and $|M|$ is the modality encoding size (i.e., the total number of involved modalities). When processing each modality, we concatenate the input channels, the positional encoding, and the modality encoding along the channel/embedding axis before adding zero-padding along this axis to match a desired total embedding size d_{all}. As a result, all modalities will be processed to have the same embedding size d_{all}. We also flatten all non-embedding dimensions so the processed input will always have shape $n \times t_m \times d_{all}$ where n is a common batchsize, t_m is a modality-specific sequence length, and d_{all} is the common embedding dimension.

For example, during multitask learning in the large setting (4 datasets involved: UR-FUNNY, MOSEI, MIMIC, and AV-MNIST), $d_{all} = 387$ (because the image modality from UR-FUNNY has a channel size of 371, positional encoding size of 7, and modality encoding size of 9). When processing the colorless image modality from AVMNIST ($7 \times 7 \times 16$), we have a channel size of 16, positional encoding size of 26, and modality encoding size of 9, so the processed output will be 49×387 where the first 16 dimensions along the last dimension represent 16 raw input dimensions, the next 336 dimensions are padded zeroes, the next 26 dimensions are positional encodings, and the final 9 dimensions are modality encodings.

Note that during this entire processing step all procedures are programmatic and there are no trainable parameters involved.

B.2 Unimodal Perceiver Encoder

Now that we have standardized all modality inputs into a common representation, we follow the Perceiver architecture (Jaegle et al., 2021b) to perform modality and task-agnostic representation learning from each input modality. Starting with a latent array of shape $d_{LN} \times d_{LS}$ (array size configurable as a hyperparameter,

where d_{LN} is the number of latent vectors and d_{LS} is the latent dimension) with trainable initialization, for each layer, we first perform cross-attention on the latent array using the processed input array (of shape $t_m \times d_{all}$) as context. Cross-attention between the latent vector and the input modality sequence learns relationships between elements in each modality, resulting in unimodal contextualized representations. The resulting latent array then goes through a latent transformer (with self-attention and feed-forward layers). We repeat this architecture for each layer within the encoder. The main advantage of this Perceiver encoder is that it can encode the input into a common $d_{LN} \times d_{LS}$ latent array regardless of the input shape $t_m \times d_{all}$, and the total runtime is linear with respect to the size of t_m which scales to high-modality scenarios. Note that only one copy of a unimodal Transformer (Perceiver) block is used to encode all modalities simultaneously, which enables statistical strength sharing and general-purpose representation learning regardless of the specific input modality.

B.3 Crossmodal Transformer layer

To learn modality and task-agnostic multimodal representations, we use multiple layers of a general-purpose Crossmodal Transformer block (Tsai et al., 2019; Lu et al., 2019). Given 2 unimodal representations $\mathbf{z}_1$ and $\mathbf{z}_2$ of common shape $d_{LN} \times d_{LS}$ learned from unimodal Perceiver encoders, a Crossmodal Transformer (CT) block uses crossmodal self-attention by setting the input layer query $Q = \mathbf{z}_1$ and keys and values $K, V = \mathbf{z}_2$ to learn attention from modality 1 to modality 2, and a separate block to capture the attention in the opposite direction. This step enables one modality's sequence elements to discover correspondences in another. A Crossmodal Transformer block using $\mathbf{z}_1$ to attend to $\mathbf{z}_2$ (and vice-versa) results in a multimodal representation $\mathbf{z}_{\text{mm}} = [\mathbf{z}_{1\to2}, \mathbf{z}_{2\to1}] = [\text{CT}(\mathbf{z}_1, \mathbf{z}_2), \text{CT}(\mathbf{z}_2, \mathbf{z}_1)]$. For each layer, we first perform cross-attention followed by self-attention and feed-forward functions. In the end, we only take the last d_{LS}-dimensional vector out of the $d_{LN} \times d_{LS}$ final latent array as the output of this module. For tasks with more than 2 modalities, a Crossmodal Transformer block is applied for each pair of modalities before concatenating all multimodal representations. Again, only one copy of a multimodal layer is used on all tasks to learn general representations regardless of the input modalities and task.

B.4 Task-specific classifiers

Since each task may have a different number of modalities and output classes, we create a separate classification head for each task. For each classification head, it concatenates all outputs of the Crossmodal Transformer layer (so 2-modality tasks have concatenated size of $2d_{LS}$, 3-modality tasks have concatenated size of $6d_{LS}$, etc), performs batch-normalization, and feeds the normalized multimodal representation $\mathbf{z}_{\text{mm}}$ into a linear layer that maps to the logits for this task. This classification layer composes individual correspondences learned within and across modalities to form a final prediction.

B.5 Homogeneous multitask pre-training

Since each task has a different number of training batches, not all tasks will be involved in each training step. We arrange the tasks to be included in each training step such that more tasks will be trained simultaneously towards the end of an epoch. For example, if task A has 300 training batches, task B has 200 training batches, and task C has 100 training batches, then for the first 100 training steps in an epoch, only task A will be used; then for the next 100 steps both A and B will be used; and for the last 100 steps, all three tasks will be used. This approach tends to work better than including all tasks in all steps via uniform batch sampling because the task with fewer training batches tends to overfit in the latter approach.

Within each training step, we compute the losses of the batch from each task used and compute the gradient using a weighted sum of the losses. The weights are part of the hyperparameters that we can tune to ensure balanced training. Then we update the model using the computed gradients.

We compute validation performance after each epoch for each task, and aggregate validation performances across all tasks (this is necessary because different tasks are measured differently, sometimes bigger is better, sometimes smaller is better). When all tasks are accuracy-based (such as the large setting), we just weigh

them equally. Then we report test performance on the checkpoint with the highest aggregated validation performance.

The result from homogeneous multitask pre-training is a set of modality embeddings, common unimodal and crossmodal parameters $\mathbb{U}^*$ and $\mathbb{C}^*$, and individual task classifiers.

B.6 Heterogeneity-aware fine-tuning

We account for heterogeneity by grouping unimodal parameters based on modalities that we know to be similar from §2.1 (e.g., setting $\mathbb{U}_1 = \{U_1, U_2\}, \mathbb{U}_2 = \{U_3\}, \mathbb{U}_3 = \{U_4, U_5, U_6\}$), and likewise for the crossmodal parameters (e.g., $\mathbb{C}_1 = \{C_{12}, C_{13}, C_{14}\}, \mathbb{C}_2 = \{C_{23}, C_{15}\}, \mathbb{C}_3 = \{C_{24}, ...\}$). These groups of parameters are first initialized with the homogeneous model $\mathbb{U}^*$ and $\mathbb{C}^*$ before separate fine-tuning, which results in final parameters $\mathbb{U}^* \rightarrow \{\mathbb{U}_1^*, \mathbb{U}_2^*, ...\}$ and $\mathbb{C}^* \rightarrow \{\mathbb{C}_1^*, \mathbb{C}_2^*, ...\}$. The modality embeddings and task classifiers are jointly fine-tuned as well.

B.7 Transfer learning details

If we are trying to transfer from tasks $\{A, B, C\}$ to D, initially we start with a randomly initialized HighMMT model that defines modality embeddings for all modalities in $\{A, B, C, D\}$ as well as a classification head for each. Then, we pretrain the model using multitask learning on $\{A, B, C\}$ using the same procedure as before. After saving a good checkpoint as measured by aggregated validation performance on pretraining tasks $\{A, B, C\}$, we finetune the trained model on target task D. The modality and tasks in $\{A, B, C\}$ present during multitask pretraining can be very different from those encountered in D during fine-tuning.

B.8 Few-shot multitask learning details

We also investigated few-shot learning using limited labeled data in a target task. When we perform few-shot learning on task D with the help of tasks $\{A, B, C\}$, we jointly train $\{A, B, C, D\}$ together in the same multitask manner as before, but since we don't care about the performance of our model on auxiliary tasks $\{A, B, C\}$, we assign a higher weight to the losses on task D and keep track of the best validation performance on D when selecting checkpoints.

C Experimental Setup

In this section, we provide additional details on the experimental setup to analyze the multitask, transfer, and generalization capabilities of HighMMT.

C.1 Setup

We use a large collection of multimodal datasets provided in the standardized and public MultiBench benchmark (Liang et al., 2021b). This benchmark spans 15 real-world datasets, 10 modalities, 20 prediction tasks, and 6 research areas. Each of these datasets requires a model to learn basic representations of features in each modality and aggregate complementary information across multiple modalities to make a prediction.

Affective computing involves understanding our natural display of multimodal signals spanning language (spoken words), visual (facial expressions, gestures), and acoustic (prosody, speech tone) in order to predict human affective states (emotions, sentiment, and personalities) (Picard, 2000). We test on 2 datasets involving fusing *language*, *video*, and *audio* time-series data to predict sentiment and emotions (MOSEI (Zadeh et al., 2018)) as well as humor (UR-FUNNY (Hasan et al., 2019)).

Healthcare: Medical decision-making often involves integrating multiple sensory readings from instruments such as lab tests, imaging reports, and patient-doctor conversations (Amisha et al., 2019). We experiment with the large-scale MIMIC dataset (Johnson et al., 2016) which records ICU patient data including *time-series* data measured every hour and other demographic variables in the form of *tabular numerical* data. These are used to predict the disease ICD-9 code and mortality rate.

Figure 8: The 17 concepts shared across image and audio datasets that were used to define positive retrieval groups in CIFAR-ESC. Note that we only show the images - the audio spectrograms make up the second modality in each concept.

Robotics: Modern robot systems are equipped with multiple sensors in order to capture complementary signals useful for holistic decision-making. We test on the large-scale MujoCo PUSH (Lee et al., 2020a) and V&T (Vision&Touch) (Lee et al., 2020b) datasets which record the manipulation of simulated and real robotic arms equipped with *visual* (RGB and depth), *force*, and *proprioception* sensors. In PUSH, the goal is to predict the pose of the object being pushed by the robot end-effector. In V&T, the goal is to predict action-conditional learning objectives that capture forward dynamics (contact prediction and robot end-effector pose).

Human Computer Interaction (HCI) studies the design of computer technology and interactive interfaces between humans and computers (Dix et al., 2000). We use the ENRICO dataset (Deka et al., 2017; Leiva et al., 2020) of Android app screens (consisting of an *image* as well as a *set* of apps and their locations) categorized by their design motifs and collected for data-driven design applications such as design search, user interface (UI) layout generation, UI code generation, and user interaction modeling.

Multimedia: A significant body of research in multimodal learning has been fueled by the large availability of multimedia data (language, image, video, and audio) on the internet. We experiment on 2 large-scale multimedia datasets with varying sizes and levels of difficulty: (1) AV-MNIST (Vielzeuf et al., 2018) is assembled from *images* of handwritten digits (LeCun et al., 1998) and *audio* samples of spoken digits (Leonard and Doddington, 1993), and (2) CIFAR-ESC (Liang et al., 2021c) is an image-audio retrieval dataset. To construct CIFAR-ESC, we follow Liang et al. (2021c) and combine 100 classes from CIFAR-100 and 10 classes from CIFAR-10 (Krizhevsky et al., 2009) to form 110 image classes, as well as 50 audio classes from ESC-50 (Piczak, 2015). To bridge these two modalities with partially related label spaces, we define 17 shared classes across the 2 datasets for weak concept alignment. These clusters are obtained by mapping similar classes between the datasets using similarities from WordNet (Miller, 1995) and text cooccurrence, and we show the resulting 17 clustered concepts we used for weak alignment in Figure 8. For the retrieval task, we first split all images and audio into a 3/1/1 train/valid/test split. Within each split, we paired each image with one randomly selected audio clip from the same shared class and label that pair as positive, and with one randomly selected audio clip from a different shared class and label that pair as negative. We evaluate retrieval performance via binary classification accuracy of an image and audio clip classified into either positive or negative pairs. The final retrieval dataset consists of 38K training pairs, 13K validation pairs and 13K test pairs.

Multitask setup: We trained 3 multitask models across combinations of the aforementioned datasets. Each multitask setup is designed to include tasks with different modality inputs and prediction objectives.

1. **Small:** PUSH, V&T: 2 tasks in the same research area (robotics) but with different modality inputs: {image, force, proprioception, control} and {image, force, proprioception, depth} respectively. Furthermore, each robot's sensor readings come from different robot-dependent sensors.

2. **Medium:** ENRICO, PUSH, AV-MNIST across 3 domains (multimedia, HCI, and robotics) with different modalities: {image, set}, {image, force, proprioception, control}, and {image, audio}.

Table 6: We investigate 3 multitask training setups to evaluate the performance of HighMMT. Each multitask setup is designed to include tasks with different modality inputs and prediction objectives. The total size of datasets involved in our experiments exceeds $370,000$ and covers diverse modalities such as images, video, audio, text, time-series, various robotics sensors, sets, and tables, as well as multiple research areas and prediction tasks from affective computing, healthcare, multimedia, robotics, and HCI.

Setting	Datasets	Modalities	Size	Prediction task	Research Area
Small	Push	{image, force, proprioception, control}	37,990	object pose	Robotics
	V&T	{image, force, proprioception, depth}	147,000	contact, robot pose	Robotics
Medium	ENRICO	{image, set}	1,460	design interface	HCI
	Push	{image, force, proprioception, control}	37,990	object pose	Robotics
	AV-MNIST	{image, audio}	70,000	digit	Multimedia
Large	UR-FUNNY	{text, video, audio}	16,514	humor	Affective Computing
	MOSEI	{text, video, audio}	22,777	sentiment, emotions	Affective Computing
	MIMIC	{time-series, table}	36,212	mortality, ICD-9 codes	Healthcare
	AV-MNIST	{image, audio}	70,000	digit	Multimedia

3. **Large:** UR-FUNNY, MOSEI, MIMIC, and AV-MNIST, across 3 domains (affective computing, healthcare, and multimedia), again with different modalities: {text, video, audio} for the first 2 tasks with different format of preprocessed embeddings of video and audio, {time-series, table}, and {image, audio}.

We summarize these experimental settings in Table 6. Overall, the total size of datasets involved in our experiments exceeds $370,000$ and covers diverse modalities such as time-series, various robotics sensors, sets, and tables, as well as multiple research areas and prediction tasks from affective computing, healthcare, multimedia, robotics, and HCI.

C.2 Hyperparameters and training details

We list hyperparameters used throughout our models in Table 7, Table 8, and Table 9 for small, medium, and large multitask settings respectively. Code is also included in the supplementary material for reproducibility.

Table 7: Table of hyperparameters for multitask prediction on the **small** setting involving PUSH, V&T: 2 tasks in the same research area (robotics) but with different modality inputs: {image, force, proprioception, control} and {image, force, proprioception, depth} respectively, and readings come from different robot-dependent sensors.

Part of Model	Hyperparameter	Values	
		PUSH	V&T
Unimodal Perceiver Encoder	Depth	1	
	Num Latents	20	
	Latent Dim	64	
	Cross Attention Heads	1	
	Latent Self-Attention Heads	8	
	Cross Head Dim	64	
	Latent Head Dim	64	
	Num Latent Blocks Per Layer	1	
Multimodal Cross-Attention Layer	Depth	1	
	Num Latents	20	
	Latent Dim	64	
	Cross Attention Heads	1	
	Latent Self-Attention Heads	8	
	Cross Head Dim	64	
	Latent Head Dim	64	
	Num Latent Blocks Per Layer	1	
Classification Heads (BatchNorm+Linear)	Input/output dimensions	756/32	1280/1
Training	Optimizer	Adam	
	Learning rate	0.0005	
	Weight decay	0.0	
	Training loss weights	100.0	1.0
	Batchsize	18	64
	Evaluation weights	100.0	1.0
	Original MultiBench Input Dimensions	Gripper Pos: 16x3 Gripper Sensors: 16x7 Image: 16x32x32 Control: 16x7	Image: 128x128x3 Force: 6x32 Proprio: 8 Depth: 128x128 Action: 4
	Perceiver Input Channel Size	Gripper Pos: 3 Gripper Sensors: 7 Image: 1 Control: 7	Image: 3 Force: 32 Proprio: 8 Depth: 1 Action: 4
	Perceiver Input Extra Axis	Gripper Pos: 1 Gripper Sensors: 1 Image: 3 Control: 1	Image: 2 Force: 1 Proprio: 1 Depth: 2 Action: 1
	Perceiver Input Num_freq_bands	Gripper Pos: 6 Gripper Sensors: 6 Image: 6 Control: 6	Image: 6 Force: 6 Proprio: 6 Depth: 6 Action: 6
	Perceiver Input Max_freq	Gripper Pos: 1 Gripper Sensors: 1 Image: 1 Control: 1	Image: 1 Force: 1 Proprio: 1 Depth: 1 Action: 1
	Shared Modality Encoding	N/A	

Table 8: Table of hyperparameters for multitask prediction on the **medium** setting involving AV-MNIST, ENRICO and PUSH: 3 tasks across 3 domains (multimedia, HCI, and affective computing), again with vastly different modality sets: {image, audio}, {image, set}, and {image, force, proprioception, control} for each task.

Part of Model	Hyperparameter	Values		
		AV-MNIST	ENRICO	PUSH
Unimodal Perceiver Encoder	Depth	1		
	Num Latents	12		
	Latent Dim	64		
	Cross Attention Heads	1		
	Latent Self-Attention Heads	8		
	Cross Head Dim	64		
	Latent Head Dim	64		
	Num Latent Blocks Per Layer	1		
Multimodal Cross-Attention Layer	Depth	1		
	Num Latents	12		
	Latent Dim	64		
	Cross Attention Heads	1		
	Latent Self-Attention Heads	8		
	Cross Head Dim	64		
	Latent Head Dim	64		
	Num Latent Blocks Per Layer	1		
Classification Heads (BatchNorm+Linear)	Input/output dimensions	128/10	128/20	768/2
Training	Optimizer	Adam		
	Learning rate	0.001		
	Weight decay	0.0		
	Training loss weights	0.8	1.0	1.1
	Batchsize	32	32	32
	Evaluation weights	1	1	1
	Original MultiBench Input Dimensions	Colorless Image: 28x28 Audio Spectogram: 112x112	Image: 256x128x3 Set: 256x128x3	Gripper Pos: 16x3 Gripper Sensors: 16x7 Image: 16x32x32 Control: 16x7
	Perceiver Input Channel Size	Colorless Image: 16 (cut into 4x4 squares) Audio Spectogram: 256 (cut into 16x16 squares)	Image: 3 Set: 3	Gripper Pos: 3 Gripper Sensors: 7 Image: 1 Control: 7
	Perceiver Input Extra Axis	Colorless Image: 2 Audio Spectogram: 2	Image: 2 Set: 2	Gripper Pos: 1 Gripper Sensors: 1 Image: 3 Control: 1
	Perceiver Input Num_freq_bands	Colorless Image: 6 Audio Spectogram: 6	Image: 6 Set: 6	Gripper Pos: 6 Gripper Sensors: 6 Image: 6 Control: 6
	Perceiver Input Max_freq	Colorless Image: 1 Audio Spectogram: 1	Image: 1 Set: 1	Gripper Pos: 1 Gripper Sensors: 1 Image: 1 Control: 1
	Shared Modality Encoding	N/A		

Table 9: Table of hyperparameters for multitask prediction on the **large** setting involving MIMIC, AV-MNIST, MOSEI and UR-FUNNY: 4 tasks across 3 domains (healthcare, multimedia, and affective computing), again with vastly different modality sets: {time-series, table}, {image, audio}, and {text, video, audio} for the final 2 tasks with different format of preprocessed embeddings of video and audio.

Part of Model	Hyperparameter	Values			
		MIMIC	AV-MNIST	MOSEI	UR-FUNNY
Unimodal Perceiver Encoder	Depth	1			
	Num Latents	20			
	Latent Dim	64			
	Cross Attention Heads	1			
	Latent Self-Attention Heads	6			
	Cross Head Dim	64			
	Latent Head Dim	64			
	Num Latent Blocks Per Layer	1			
Multimodal Cross-Attention Layer	Depth	1			
	Num Latents	20			
	Latent Dim	64			
	Cross Attention Heads	4			
	Latent Self-Attention Heads	6			
	Cross Head Dim	64			
	Latent Head Dim	64			
	Num Latent Blocks Per Layer	1			
Classification Heads (BatchNorm+Linear)	Input/output dimensions	128/2	128/10	384/2	384/2
Training	Optimizer	Adam			
	Learning rate	0.0008			
	Weight decay	0.001			
	Training loss weights	1.2	0.9	1.1	1.5
	Batchsize	20	40	32	32
	Evaluation weights	1	1	1	1
	Original MultiBench Input Dimensions	Static: 5 Timeseries: 24x12	Colorless Image: 28x28 Audio Spectogram: 112x112	Image: 50x35 Audio: 50x74 Text: 50x300	Image: 20x371 Audio: 20x81 Text: 50x300
	Perceiver Input Channel Size	Static: 1 Timeseries: 1	Colorless Image: 16 (cut into 4x4 squares) Audio Spectogram: 256 (cut into 16x16 squares)	Image: 35 Audio: 74 Text: 300	Image: 371 Audio: 81 Text: 300
	Perceiver Input Extra Axis	Static: 1 Timeseries: 2	Colorless Image: 2 Audio Spectogram: 2	Image: 1 Audio: 1 Text: 1	Image: 1 Audio: 1 Text: 1
	Perceiver Input Num_freq_bands	Static: 6 Timeseries: 6	Colorless Image: 6 Audio Spectogram: 6	Image: 3 Audio: 3 Text: 3	Image: 3 Audio: 3 Text: 3
	Perceiver Input Max_freq	Static: 1 Timeseries: 1	Colorless Image: 1 Audio Spectogram: 1	Image: 1 Audio: 1 Text: 1	Image: 1 Audio: 1 Text: 1
	Shared Modality Encoding	The text modality from MOSEI and UR-FUNNY are shared.			

Table 10: We train multitask HighMMT on 1/2/3 datasets and find that it generalizes to new modalities and tasks on the 4th dataset, with improved performance over single-task training on the 4th dataset. 0 source tasks implies transferring randomly initialized parameters, which is equivalent to single-task training on the target task. Cross-modal transfer improves with the number of pretraining tasks and works best on the smallest target tasks (UR-FUNNY).

Source tasks	Target task UR-FUNNY
0 (no transfer)	63.3
MOSEI	64.1
MOSEI + AV-MNIST	65.5
MOSEI + MIMIC + AV-MNIST	**65.7**

Source tasks	Target task MOSEI
0 (no transfer)	79.4
AV-MNIST	79.4
AV-MNIST + MIMIC	80.0
UR-FUNNY + MIMIC + AV-MNIST	**80.5**

Source tasks	Target task MIMIC
0 (no transfer)	67.7
MOSEI	68.3
AV-MNIST + MOSEI	68.5
UR-FUNNY + MOSEI + AV-MNIST	**68.5**

Source tasks	Target task AV-MNIST
0 (no transfer)	70.4
MOSEI	70.4
MIMIC + MOSEI	70.5
UR-FUNNY + MOSEI + MIMIC	**70.5**

D Additional Results

In this section, we detail additional experimental results that support the multitask, transfer, and generalization capabilities of HighMMT.

D.1 Generalization to new modalities and tasks

HighMMT also offers opportunities to study whether we can *transfer* knowledge between completely different tasks and modalities. On the large setting, we first pretrained a model on 0/1/2/3 of the four tasks before fine-tuning on the fourth task only. We show these full results in Table 10. On all four target tasks, our proposed multitask pretraining and fine-tuning paradigm improves performance over single target-task training. Therefore, weights learned from other multimodal tasks indeed generalize well to new modalities and tasks. We further analyze this transfer learning phenomenon by studying the following research questions:

Effect of pretraining datasets. When we vary the number of pretraining datasets, we observe a consistent improvement on fine-tuned target task performance across all datasets. This effect is particularly pronounced on the UR-FUNNY target task, which shows the biggest improvement using pretrained parameters from 0 to 3 multitask datasets. This implies that HighMMT learns more generalizable multimodal features as more tasks are involved in multitask training.

Effect of target dataset size. We observed an inverse correlation between target task size and performance improvement: the smallest dataset, UR-FUNNY, benefited the most (+2.4%) from transfer learning. This implies that this multimodal pretraining-fine-tuning paradigm is useful for improving performance for low-resource target modalities and tasks.

Effect of transfer modalities. We compare transfer learning performance across different levels of partial observability. While one would expect transfer to the MIMIC dataset to be the hardest due to its modality

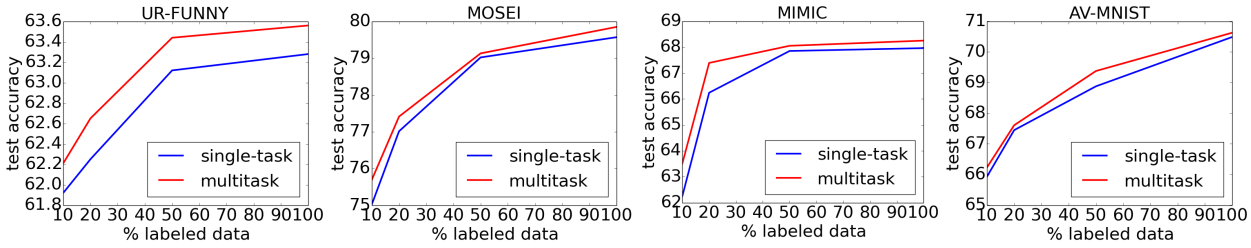

Figure 9: **Few-shot results** on new modalities and tasks. Multimodal multitask training using HighMMT learns more generalizable representations which improves performance across all ranges of data. The x-axis shows the percentage of labeled data used during training.

set {time-series, table} being completely disjoint from the remaining 3 datasets, we still observe a +0.8% gain as compared to single-task training. Therefore, HighMMT can generalize to new modalities and tasks. Unsurprisingly, for datasets with more overlap in modality sets (e.g., UR-FUNNY with complete overlap in {text, video, audio} as compared to the other 3 datasets used for pretraining, we find larger improvements using transfer learning over single-task models (+2.4%).

Comparisons to unimodal transfer. Recent work has explored the possibility of transferring Transformer representations trained in one modality to another. Lu et al. (2021) found that a frozen pretrained Transformer on text surprisingly transfers to a variety of sequence classification tasks of different modalities spanning numerical computation, vision, and protein fold prediction. This observation had been previously observed in transfer learning from language to vision (Kiela et al., 2019), referential communication games to real-world NLP tasks (Li et al., 2020), computational primitives to transfer to mathematics tasks (Wu et al., 2021), and between code, different languages, and music (Papadimitriou and Jurafsky, 2020). Our transfer experiments also corroborate these findings in a multimodal setting with promising results on new modalities and tasks, especially involving real-world, smaller, and noisier datasets such as those involving human videos (MOSEI and UR-FUNNY), medical data (MIMIC), or real and simulated robots (Push and V&T).

D.2 Few-shot learning

HighMMT offers opportunities for statistical strength sharing across tasks. We test this hypothesis in the few-shot learning scenario, by evaluating whether multitask information sharing can improve performance on low-resource target tasks. We compare a single-task HighMMT trained only on a percentage p of labeled training data in the target task with multitask HighMMT trained on the same percentage p (during multitask training we prioritize performance of the target task over others). By varying $p \in [0.1, 1.0]$, we plot the performance under few-shot settings in Figure 9. We find that multitask training is consistently better across all ranges of data, which supports the fact that more generalizable representations across modalities and tasks are learned in HighMMT. The main takeaway is that if it is too difficult to collect data in a target domain, collecting data from a different domain and using a shared multimodal model is an alternative approach for improving performance.

D.3 Multitask fusion and retrieval

To assess task generalization, we train multitask models over fusion in AV-MNIST and retrieval in CIFAR-ESC. While fusion emphasizes information integration from complementary data sources, retrieval focuses on aligning corresponding elements expressed through different views of the data (Baltrušaitis et al., 2018). Table 11 shows the full results of this experiment: even across vastly different multimodal prediction tasks, we find that multitask training (60.5% retrieval accuracy) improves upon single-task training (58.8% accuracy), while performance on the AV-MNIST fusion tasks is similar for both single-task and multitask learning. Not only have the unimodal encoders simultaneously processed different modalities, the multimodal attention layer has also learned to capture correspondences useful for both fusion and retrieval, while halving the total number of parameters required as compared to task-specific modeling.

Table 11: Multitask HighMMT also enables training a single model for both multimodal fusion and retrieval tasks.

Model	AV-MNIST ↑	CIFAR-ESC ↑	Params (M) ↓
HighMMT	70.4	58.8	1.04
HighMMT multitask	70.4	**60.5**	**0.52**

Table 12: We conduct in-depth **ablation studies** on the architecture design, parameter sharing settings, and fine-tuning strategies in HighMMT, and find strong evidence for (1) having separate unimodal and interaction architecture layers, (2) determining parameter sharing via feature transfer rather than having parameters fully separate, fully shared, or computed via feature difference across modalities and tasks, and (3) homogeneous pre-training before heterogeneity-aware fine-tuning into parameter groups rather than directly training for heterogeneity.

	Model	UR-FUNNY ↑	MOSEI ↑	MIMIC ↑	AV-MNIST ↑	Ave ↑
	HighMMT	**66.3**	**80.2**	**68.5**	**71.3**	**71.6**
Architecture ablations	- w/o embeddings	62.5	78.4	67.9	69.5	69.6
	- w/o unimodal	57.6	61.8	63.0	59.1	60.4
	- w/o crossmodal	61.3	80.3	67.7	69.4	69.7
Param sharing ablations	- share none	63.4	79.7	68.5	69.0	70.2
	- share unimodal	63.5	79.5	65.3	70.0	69.6
	- share crossmodal	64.6	79.9	65.4	69.3	69.9
	- share all	63.0	79.9	67.8	70.4	70.3
	- random difference	62.4	79.5	67.6	70.4	70.4
	- feature difference	63.0	79.7	68.1	70.5	70.3
Training ablations	- direct training	61.2	78.5	64.8	71.1	69.9

D.4 Understanding HighMMT

In this subsection, we analyze why this general model achieves strong results in multitask, transfer, and few-shot settings. Based on prior work in multitask learning (Caruana, 1997; Ruder, 2017; Zhang and Yang, 2021), we set up two possible hypotheses: (1) improved generalization and (2) improved regularization. We supplement the results in the main paper with additional visualizations and comparisons in this subsection.

D.4.1 Hypothesis 1: Improved generalization

Investigating parameter sharing. In which components of the HighMMT model is parameter sharing important?

We further study the importance of parameter sharing in HighMMT. From the ablation studies in Table 12, using separate parameters for either unimodal or multimodal layers results in worse performance. The full model with completely separate unimodal and multimodal layers is reminiscent of typical single-task multimodal transformers (Tsai et al., 2019; Lu et al., 2019; Hendricks et al., 2021) trained separately for each task. We show that HighMMT maintains competitive performance (with slightly better performance on several datasets) due to statistical strength sharing, while also reducing parameters by 6× due to sharing of unimodal encoders and multimodal layers across tasks.

Furthermore, we surprisingly find that removing the modality-specific embedding layer results in only slightly worse performance (70.3 to 69.6 average score). This implies that the shared unimodal encoder has learned generalizable feature extractors that can encode heterogeneous modalities even without a modality identifier.

Visualization of attention patterns. How do the shared unimodal encoders attend to modality-specific tokens?

Given that parameter sharing seems to be useful for performance and efficiency, we aim to better visualize the nature of information sharing in the attention layers of unimodal encoders. We perform inference on a trained multitask HighMMT model on the test data of the large multitask setting, and average the attention patterns across test datapoints for each dataset. Following Lu et al. (2021), the average attention pattern provides information on general inductive biases captured by the unimodal encoders and enables us to make holistic conclusions rather than comparing attention maps on individual datapoints.

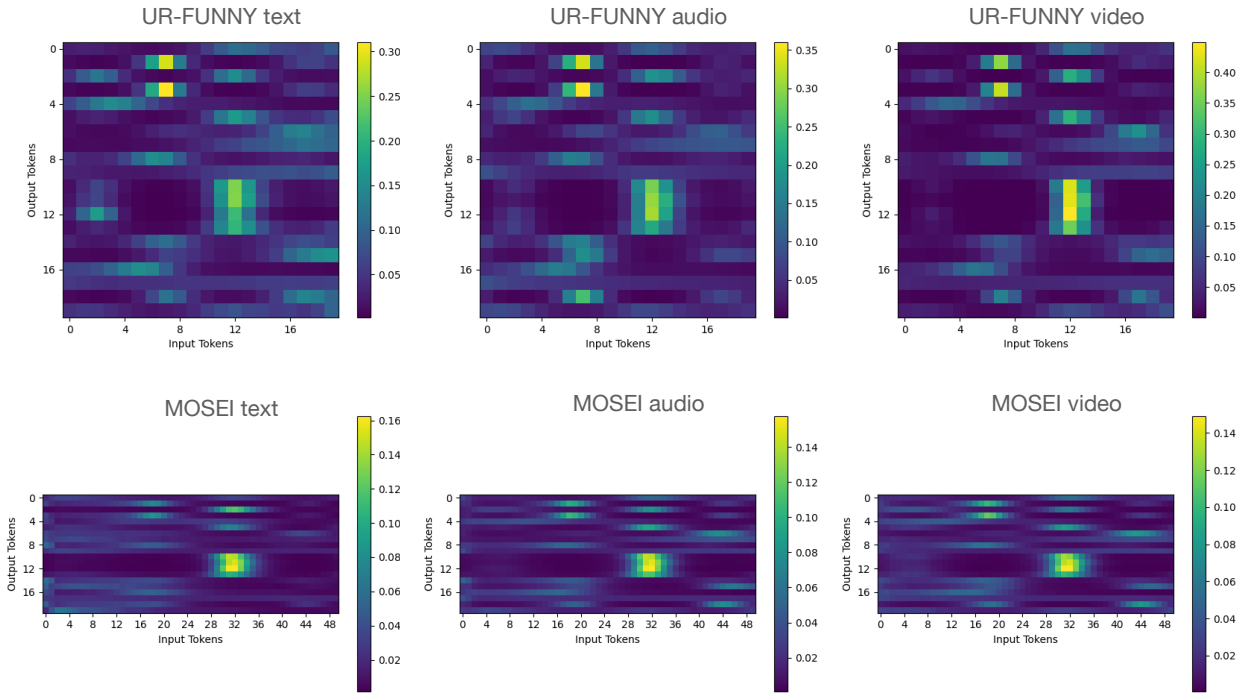

Figure 10: Visualizations of attention patterns learned by the unimodal encoders across different modalities and datasets. We find that there are some common attention patters across tasks, which implies that the shared unimodal encoders have learned common information across tasks.

From Figure 10, we actually find that a common attention pattern emerges across modalities and tasks. First looking across modalities in the same dataset, we find that the model captures common temporal patterns at the same time steps, which makes sense since the 3 modalities are time-aligned in a video. The attention patterns are quite similar which implies that the same attention strategy can often work well across different modalities and tasks. This could be an explanation of why our model is able to perform multiple tasks simultaneously using shared parameters in attention layers.

It is also interesting to see how the model automatically learns to "divide up work" amongst its 20 latent tokens (numbered 0-19): the latent tokens $10-13$ typically all focus on the region about two-thirds after the start of the input sequence, while latent tokens 1 and 3 always focuses on the region about one-third from the start. Certain tokens (3 and 18) seem to learn oscillating attention patterns, and certain pairs of tokens learn complementary attention patterns (e.g., 4, 5, and 6 attend one after the other). There are also some latent tokens that more evenly attend to the whole input sequence, such as latent tokens 9 and 17, which can be seen as "summary" tokens. This shows that the perceiver-based encoder is able to divide up its limited latent space well to capture important information both in specific time-steps and contextual information across all the time-steps, thus creating a holistic representation of the input using a much smaller set of latent variables.

D.4.2 Hypothesis 2: Improved regularization

In parallel to improved generalization, another line of research has focused on the regularization effects of multitask learning. Baxter (1997) showed that multitask parameter sharing reduces the risk of overfitting on the original task by forcing the model to learn across multiple tasks. We study the following regularization effects:

Training dynamics. In Figure 11, we traced the train and valid accuracies across 60 training epochs (with multitask training in the large setting). The training process of HighMMT converges at about the same rate between single-task and multitask learning, but the multitask model overfits less (a smaller gap between

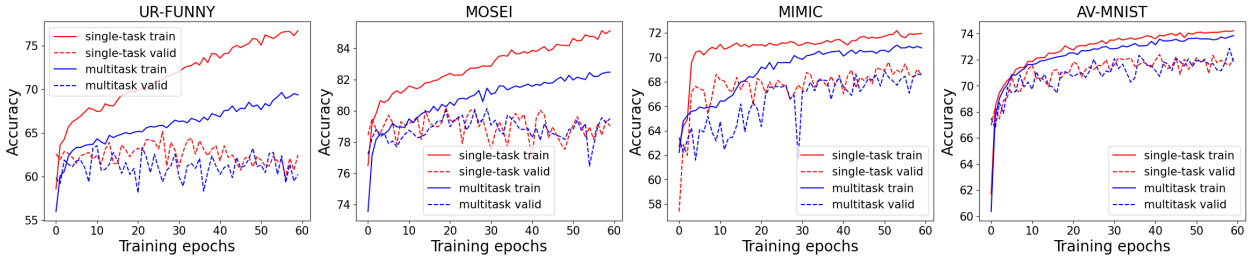

Figure 11: Multitask models converge as fast but overfit less (a smaller gap between train and valid accuracies) vs single-task models, which implies that multitask training helps to regularize the joint parameters and reduces overfitting on the target task.

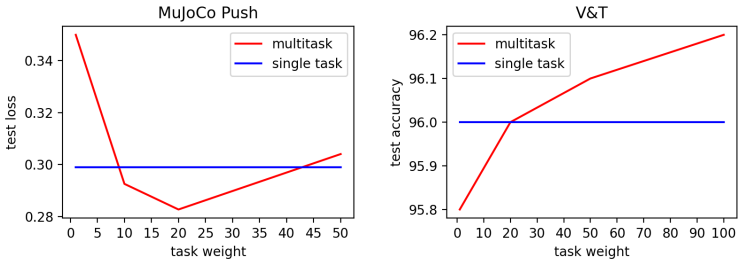

Figure 12: Multitask performance can sometimes be sensitive to task weights especially when prediction objectives are of different scales (i.e., MSE for PUSH vs accuracy for V&T), in a manner similar to how carefully-tuned regularization terms help in training models.

training and valid accuracies). This implies that multitask training helps to regularize the joint parameters and alleviates their overfitting on the target task.

Task weights. We found that optimizing a simple weighted sum of loss functions over all tasks was sufficient to obtain strong multitask performance. Instead of assigning uniform weights to each task, sometimes we found it helpful to set the weight higher for more challenging datasets during HIGHMMT multitask training. We show some examples of this phenomenon in Figure 12, where multitask performance can sometimes be sensitive to weights especially when prediction objectives are of different scales (i.e., MSE vs accuracy). This supports the regularization argument where carefully tuned weighted auxiliary objectives encouraging the model to also fit other auxiliary tasks can help improve performance on a target task. However, doing so would not achieve the best performance on auxiliary tasks.

D.5 Summary of main take-away messages

In conclusion, we designed a general multimodal multitask model for high-modality (a large set of diverse modalities) and partially-observable (each task only defined on a small subset of modalities) scenarios. Our approach relies on training for *multitask* and *transfer* learning: multitask learning with shared unimodal and multimodal layers enables stable parameter counts (addressing scalability) and cross-modal transfer learning enables information sharing across modalities and tasks (addressing partial observability). Through an extensive set of experiments and analysis, we summarize our main take-away messages as follows:

1. **Standardized multitask modeling.** We train a single multitask HIGHMMT model for numerous high-modality and partially-observable multimodal tasks (across 10 modalities, 15 prediction tasks, and 5 research areas), achieving strong performance while reducing total parameter counts. We believe that standardized modeling leads to a smaller set of architectural decisions, enables transfer to understudied modalities and tasks, and present a unified platform for subsequent theoretical and empirical analysis.

2. **Cross-modal transfer to new modalities and tasks.** Multitask HighMMT enables cross-modal information transfer by pretraining on source multimodal tasks before transferring to completely new target modalities and tasks. Involving more tasks during pretraining improves performance, and gains are more apparent when fine-tuning on low-resource target tasks. This finding can supplement current pretrain-finetune paradigms typically performed on the same modality (e.g., text-only or image-only), and encourage research in more general multimodal pretraining over high-modality settings before fine-tuning on only a partial subset of all observed modalities.

3. **Tradeoff between performance and efficiency.** Multitask HighMMT improves the tradeoff between performance and efficiency over task-specific state-of-the-art models especially in low-resource scenarios (less training data and partially-observable modalities). Coupled with the relatively fewer architectural decisions and generalization to understudied modalities and tasks, we believe that multitask HighMMT and similar architectures should be a starting point for future research.

4. **Few-shot multitask learning.** Multitask information sharing can improve performance on low-resource target tasks with limited labeled training data. Therefore, if it is too difficult to collect data in a target domain, collecting data from a different domain and using a shared multimodal model is an alternative approach for improving performance.

5. **Information sharing.** Finally, our analysis reveals surprising insights regarding the nature of information sharing in multimodal and multitask models, which may be of independent interest. Specifically, there are both generalization and regularization effects at play in our implementation of multimodal multitask learning:

    - On the generalization side, information sharing is present across modalities and tasks, but at different levels across shared unimodal and multimodal layers. Information sharing enables strong multitask performance even under partial-observability and generalization to new modalities and tasks via transfer learning.
    - While using modality-specific embeddings achieves the best performance, there is only a minor drop when removing them, which implies that shared unimodal encoders can learn generalizable feature extractors even without a modality identifier.
    - On the regularization side, well-tuned regularization weights yield training dynamics that display less overfitting on target tasks as compared to single-task learning.