HetMAML: Task-Heterogeneous Model-Agnostic Meta-Learning for Few-Shot Learning Across Modalities

Jiayi Chen, Aidong Zhang
University of Virginia
Charlottesville, VA, USA
{jc4td,aidong}@virginia.edu

ABSTRACT

Most of existing gradient-based meta-learning approaches to few-shot learning assume that all tasks have the same input feature space. However, in the real world scenarios, there are many cases that the input structures of tasks can be different, that is, different tasks may vary in the number of input modalities or data types. Existing meta-learners cannot handle the heterogeneous task distribution (HTD) as there is not only global meta-knowledge shared across tasks but also type-specific knowledge that distinguishes each type of tasks. To deal with task heterogeneity and promote fast within-task adaptions for each type of tasks, in this paper, we propose HetMAML, a task-heterogeneous model-agnostic meta-learning framework, which can capture both the type-specific and globally shared knowledge and can achieve the balance between knowledge customization and generalization. Specifically, we design a multi-channel backbone module that encodes the input of each type of tasks into the same length sequence of modality-specific embeddings. Then, we propose a task-aware iterative feature aggregation network which can automatically take into account the context of task-specific input structures and adaptively project the heterogeneous input spaces to the same lower-dimensional embedding space of concepts. Our experiments on six task-heterogeneous datasets demonstrate that HetMAML successfully leverages type-specific and globally shared meta-parameters for heterogeneous tasks and achieves fast within-task adaptions for each type of tasks.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning approaches**; **Multi-task learning**; *Supervised learning by classification.*

KEYWORDS

few-shot learning, meta-learning, task heterogeneity, multimodality

ACM Reference Format:
Jiayi Chen, Aidong Zhang. 2021. HetMAML: Task-Heterogeneous Model-Agnostic Meta-Learning for Few-Shot Learning Across Modalities. In *Proceedings of the 30th ACM Int'l Conf. on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, QLD, Australia.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3459637.3482262

CIKM '21, November 1–5, 2021, Virtual Event, QLD, Australia

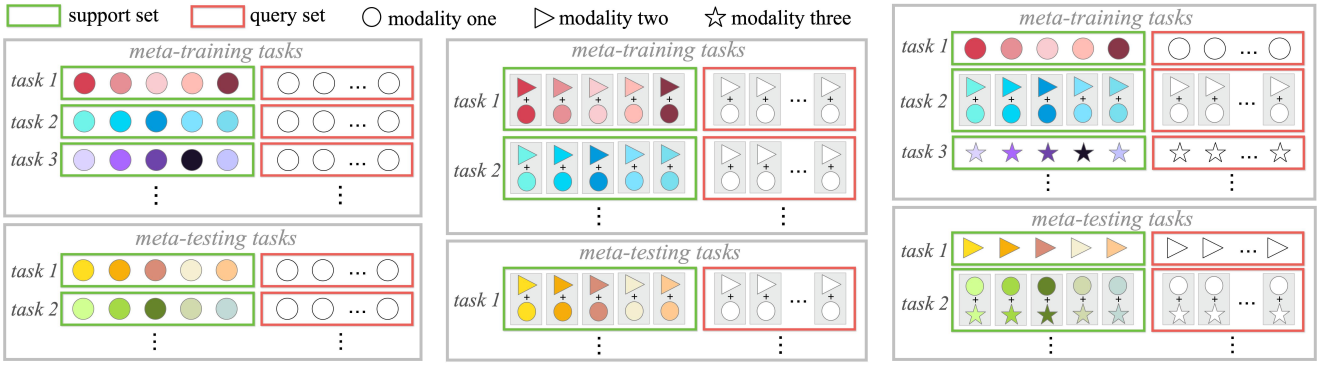
ACM ISBN 978-1-4503-8446-9/21/11…$15.00
https://doi.org/10.1145/3459637.3482262

1 INTRODUCTION

Humans can quickly learn new concepts from few examples by incorporating prior knowledge and context. Just like humans, *few-shot learning* (FSL) is the task of learning new concepts with a small number of labelled samples. *Meta-learning* is one prominent family of approaches to address FSL problems, which focuses on learning how to learn: how to generalize meta-knowledge from a distribution of tasks and utilize it to rapidly adapt to new tasks from the same task distribution. Recently, many *gradient-based meta-learning* methods have been proposed to handle FSL tasks [10, 18, 27, 38, 42, 46]. Most of these methods are model-agnostic meta-learners built upon MAML [10], aiming to find a single set of model parameters from similar tasks so that a small number of gradient updates will lead to good performance on future new tasks.

However, most existing gradient-based meta-learning methods assume the task distribution is *homogeneous*, that is, all tasks from either the same concept domain (e.g., a single dataset [10]) or a combination of different domains (e.g., multiple datasets [38]) have the same input feature space–for example, the inputs of all tasks are same-dimensional images. Such an assumption limits the generalization ability of these methods to handle more complex and *heterogeneous task distribution* (HTD): different tasks may vary in the structure of input data or the number of input modalities. For example, in few-shot emotion classification, some tasks may focus on recognizing people's emotion from the video modality, while other tasks may predict on language scripts, recorded voices, or any combination of the three modalities. Figure 1(a) shows examples of task-homogeneous FSL, which assume identical data structures in all tasks. Figure 1(b) shows an example of task-heterogeneous FSL. Compared with the homogeneous FSL in Figure 1(a), heterogeneous tasks vary in terms of data type and the structure of input. Figure 2(a) illustrates another example of task-heterogeneous FSL containing four types of tasks, where we show each *type* of tasks is associated with a specific *input space* (see the grey areas in the figure). The input space of a type of tasks refers to the feature space shared by all the training/validation samples within these tasks.

In this paper, we propose a gradient-based meta-learning approach to FSL under heterogeneous task distributions. The novel task-heterogeneous few-shot learning problem is inevitable in many real-world low-data scenarios. Considering tasks can be either unimodal or multimodal with respect to their input data sources, there are mainly two cases of FSL across modalities. First, different modalities may convey the same semantics of a concept, and it is feasible to *jointly learn multiple types of unimodal tasks* if they share knowledge in the same concept domain. For instance, the image of an animal and the caption of it (text modality) both can distinguish this animal from others. Simultaneously learning image classification



(a) Task-homogeneous few-shot learning (Left: unimodal tasks; Right: multimodal tasks) (b) Task-heterogeneous few-shot learning

Figure 1: Comparison of task-homogeneous few-shot learning and task-heterogeneous few-shot learning. Compared with task-homogeneous few-shot learning, heterogeneous tasks vary in terms of the input spaces, data types and the structure of input data, and thus will perform different within-task adaptions.

and text classification tasks that share the same concept domain could help us to build up the knowledge of animal concepts. Second, integrating multiple views of a concept or using auxiliary modalities usually achieves better performance than learning with a single modality [22, 41]. In this multimodal case, a problem neglected by existing literature is that due to the potential data missingness or data corruptness in some few-data scenarios, many tasks may not have a complete set where all modalities are always available [15]. Hence, it is uncertain that all these multimodal tasks have the same input structure, and we have to *jointly learn multiple types of multimodal tasks, where different tasks may have different combinations of modalities*. For example, in Figure 2(a) where there are four types of tasks, while type-2 tasks only have the modality-2 data available, type-2 and type-3 tasks have the auxiliary modality-1 and modality-3, respectively. For both cases, it is beneficial to learn the heterogeneous tasks across modalities due to the following reasons: 1) jointly learning multiple views of tasks could help to build up the knowledge of the concept domain; 2) we can generalize reliable semantic feature extractors due to the implicit cross-modal alignment; and 3) the mechanism of training a unified meta-learner over heterogeneous tasks is more efficient than training separate models for each type of tasks as there are global meta-parameters that can be shared across different types of tasks.

With the heterogeneous task distribution (HTD), tasks share the same concept domain but have different input spaces (as shown in Figure 2(a)). As for meta-learning, while heterogeneous tasks could globally share some common knowledge (e.g., knowledge of the concept domain and early-stage feature embedding), there is also *type-specific knowledge*, locally shared by each type of tasks, which can reflect the different concept learning mechanisms of different types of tasks. Therefore, the key challenge of dealing with task heterogeneity is how to automatically reckon with type-specific information to promote fast within-task adaptions for each type of tasks. That is, we pursue *how to customize the globally shared meta-learner* for each type of tasks.

We address two issues in this challenge. First, the meta-learner for HTD should be able to balance *generalization* (i.e., the meta-knowledge globally shared across tasks) and *customization* (i.e., the type-specific meta-knowledge able to customize the global knowledge) based on each task's specific input space. The difficulty is that, given a new task sampled from HTD whose input structure is *unclear*, the meta-learner should automatically leverage type-specific and global knowledge to achieve the *task-aware* projection: the mechanism that maps samples from the input space to the concept space should be different for different types of tasks, as illustrated in Figure 2(a) (arrow lines). Although one may convert the problem setting from heterogeneity into homogeneity using preprocessing strategies like imputation [2, 29, 36], this may introduce extra noise to the original task which has negative impacts on the performance, especially in low-data scenarios. Directly learning with different input spaces is necessary. Second, due to the inconsistency of input structures, tasks may use different model architectures and thus cannot share a single set of model parameters. Although one can train separate homogeneous meta-learners [10, 38] for each type of tasks, this strategy is not efficient as well as not effective: there are parameters shared by several types of tasks but trained repeatedly; the separate training will reduce the number of training tasks in each type which limits the ability of knowledge generalization; and, the knowledge shared between tasks is not well-explored.

To tackle the above issues, we propose Task-**Het**erogeneous **M**odel-**A**gnostic **M**eta-**L**earning (HetMAML). The key idea of HetMAML is to leverage globally shared knowledge and type-specific knowledge to promote fast within-task adaptions for each type of tasks. We propose a three-module task-aware learning framework that can effectively capture both the global and type-specific meta-parameters and achieve the knowledge customization while simultaneously preserving knowledge generalization. Our main contributions are summarized as follows:

- We study a novel task-heterogeneous few-shot learning problem involving multiple input spaces, data types, and modality combinations. We show the importance of studying this real-world problem and define our setting of task heterogeneity. As far as we know, this is the first attempt to this problem.
- We introduce HetMAML, a model-agnostic meta-learning approach to task-heterogeneous few-shot learning. We propose a task-aware iterative feature aggregation network to

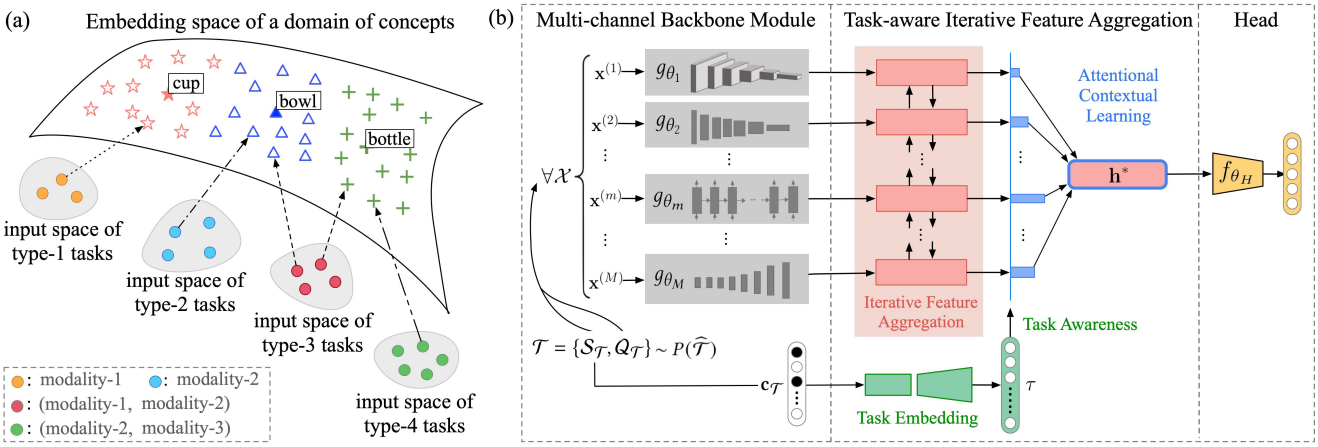


Figure 2: (a) An example of heterogeneous tasks sharing the concept domain but having different input spaces. There are a total of four types of tasks; each grey area indicates the input space of a certain type of tasks and the feature space shared by all data samples in these tasks. Colored filled circles represents unimodal or multimodal data samples; different colors within grey areas indicate that the samples have different structures, which are listed in the bottom-left box. Note that here we omit the id of samples to clearly show their structure differences. The arrow lines represent task-aware projections, meaning that data samples are mapped from different feature spaces to a unified concept space. (b) The proposed HetMAML framework.

promote the effectiveness of knowledge customization as well as adaption efficiency.

- Our experimental results on six datasets demonstrate that HetMAML can effectively capture global and type-specific meta-parameters across heterogeneous tasks and fast adapt to different types of new tasks.

2 METHODOLOGY

2.1 Problem Formulation

In this section, we will provide an explicit formulation of the few-shot learning problem under heterogeneous task distribution.

Definition 1 (*Heterogeneous Task Distribution*). Suppose we have a total of M data sources (modalities) and the data samples in each task are either unimodal or multimodal. A *heterogeneous task distribution* (HTD) $P(\widehat{\mathcal{T}})$ is defined as a mixture of M' homogeneous task distributions $P(\mathcal{T}^1), P(\mathcal{T}^2), ..., P(\mathcal{T}^{M'})$, where each $P(\mathcal{T}^r)$ is associated with a specific input space, i.e., a certain combination of the M modalities. Then there will be a total of M' types of task instances sampled from the HTD, where $2 \leq M' \leq 2^M - 1$.

Task-heterogeneous Meta-learning. We are given a dataset $\mathcal{D}$ consisting of task instances sampled from a heterogeneous task distribution $P(\widehat{\mathcal{T}})$. The dataset $\mathcal{D}$ is divided into the meta-training $\mathcal{D}^{trn}_{meta}$ and the meta-testing $\mathcal{D}^{tst}_{meta}$ sets. Each meta-training task contains N classes sampled from a set of classes C^{trn}, while the classes of each meta-testing task are sampled from a disjoint set of new classes C^{tst}. The goal of task-heterogeneous meta-learning is to find a set of meta-parameters from $\mathcal{D}^{trn}_{meta}$ which can rapidly adapt to future novel tasks $\mathcal{D}^{tst}_{meta}$.

Each task instance $\mathcal{T}_i \sim P(\widehat{\mathcal{T}})$ is composed of a within-task training set $\mathcal{S}_{\mathcal{T}_i}$ (*support set*) and testing set $\mathcal{Q}_{\mathcal{T}_i}$ (*query set*), each of which consists of a few *unimodal or multimodal* data samples. The support set of a N-way K-shot classification task contains K samples for each of N classes $\mathcal{S}_{\mathcal{T}_i} = \{(\mathcal{X}_{ink}, y_{ink})|n = 1...N, k = 1...K\}$, where $\mathcal{X}_{ink} = (\mathbf{x}^{(1)}_{ink}, \mathbf{x}^{(2)}_{ink}, ..., \mathbf{x}^{(M)}_{ink})$ denotes the composite input of each sample and y_{ink} is the label. The query set contains T samples from the same N classes $\mathcal{Q}_{\mathcal{T}_i} = \{(\mathcal{X}^*_{it}, y^*_{it})|t = 1...T\}$. Note that although we use the notation $\mathcal{X}$ to include a full set of M modalities for each sample, it still contains the information about which modalities are not available in this task. For example, if the mth source does not exist in task $\mathcal{T}_i$, the source website or sensor will return a message indicating its unavailability, or the mth modality of each sample $\mathcal{X}$ in a task is not informative. Then, given a task $\mathcal{T}_i$, we are able to recognize the input space of this task by computing a configuration vector $\mathbf{c}_{\mathcal{T}_i} = [c^{(1)}_{\mathcal{T}_i} \cdots c^{(M)}_{\mathcal{T}_i}]^\top$ from the raw data in the support set $\mathcal{S}_{\mathcal{T}_i}$ (described in Section 2.2.1).

2.2 Task-Heterogeneous Model-Agnostic Meta-Learning Framework

A traditional task neural network can be viewed as a two-module architecture, consisting of a *backbone* module (the earlier layers for feature extraction) and a *head* module (the later layers for decision making such as classification). Most existing model-agnostic meta-learners for homogeneous tasks are built upon such architecture.

Due to the existence of task heterogeneity, we design a three-module framework, HetMAML, as illustrated in Figure 2(b), consisting of three modules: a multi-channel backbone, a single-channel head module, and an additional intermediate module–the task-aware feature aggregation network (TFAN), which can *adaptively* transform the composite information produced by the backbone into a single hidden representation fed to the head.

2.2.1 Multi-Channel Backbone Module. A challenge in this problem is that the input feature space of samples is type-specific; each type of tasks has a specific combination of modalities and hence the

feature extraction schemes between different tasks can be different. For example, while image classification tasks use convolutional neural networks (CNNs) to extract features, other tasks that involve sequential data may use recurrent neural networks (RNNs). Despite the feature embedding being type-specific, each modality's feature extractor can be shared by several types of tasks. In order to enable simultaneous learning over heterogeneous tasks, our backbone module should be able to effectively model both the shared and type-specific feature extractors. To achieve this goal, we propose a *multi-channel backbone module*, where each channel is a modality-specific feature extractor. Basically, if two tasks both have a modality in their input spaces, they can share the same feature extractor's meta-parameters for embedding this modality.

Given a task $\mathcal{T}$, it is easy to identify the input space of this task based on the data samples in the support set $\mathcal{S}_\mathcal{T}$. We can obtain a task-specific vector $\mathbf{c}_\mathcal{T} = [c_\mathcal{T}^{(1)}, c_\mathcal{T}^{(2)}, \ldots, c_\mathcal{T}^{(M)}]^\top$ to represent the input space, where $c_\mathcal{T}^{(m)} = 1$ denotes that the mth modality is available in this task ($c_\mathcal{T}^{(m)} = 0$ denotes missing). Considering the missing modalities are not informative, $c_\mathcal{T}^{(m)}$ is set to 0 if and only if the mth modalities of all the samples in $\mathcal{S}_\mathcal{T}$ are similar such that $\text{Var}^{(m)} = \sum_{n,k} ||\mathbf{x}_{nk}{}^{(m)} - \bar{\mathbf{x}}^{(m)}||^2/NK < \epsilon$, where $\bar{\mathbf{x}}^{(m)} = \sum_{n,k} \mathbf{x}_{nk}{}^{(m)}/NK$ and ϵ is a threshold.

Then, suppose $\Theta_B = \{\theta_1, \theta_2, ..., \theta_M\}$ is the parameter set of the multi-channel backbone module, and $g_{\theta_m}(\cdot;\theta_m)$ is the feature extracting network for modality-m with parameter θ_m. Based on the pre-computed $\mathbf{c}_\mathcal{T}$ which indicates the input space of the task, for any sample $\mathcal{X} = (\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(M)})$ in the task $\mathcal{T}$, each modality can be embedded as $\mathbf{z}^{(m)} = g_{\theta_m}(\mathbf{x}^{(m)};\theta_m)$ if $c_\mathcal{T}^{(m)} = 1$; or, $\mathbf{z}^{(m)} = \mathbf{0}$ if $c_\mathcal{T}^{(m)} = 0$. Note that there will be no gradient update for those channels with $c_\mathcal{T}^{(m)} = 0$.

The multi-channel backbone module will produce M embedded vectors: $\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, ..., \mathbf{z}^{(M)}$. Each embedding vector $\mathbf{z}^{(m)} \in \mathbb{R}^{F_1}$ represents a piece of *modality-specific information*. Note that despite the architecture differences between channels, all modalities are encoded onto the same F_1-dimensional semantic space, reducing the impact of input statistical divergence of different types of tasks.

2.2.2 Task-Aware Feature Aggregation for Heterogeneous Tasks. The set of encoded modality-specific information $\{\mathbf{z}^{(m)}\}_{m=1}^M$ produced by the multi-channel backbone is not ready for decision making. One may consider simply concatenating all modality-specific vectors and directly feeding it to the head neural networks. However, this strategy is not effective due to the following two reasons. First, a concatenated vector is large and sparse–its dimension is $M \cdot F_1$ and many modalities that are absent in the task are represented by zero. This will result in a large feature mapping network at the beginning of the head module, which is not feasible in few-shot learning setting. Second, when there are multimodal types of tasks, the multiple input modalities can be highly-interacted [43], hence it is necessary to consider their potential interactions before decision making. These reasons motivate us to consider to effectively combine and condense all the modality-specific features. Thus, we insert an intermediate *feature aggregation* module Θ_A between the backbone module and head module, which can transform the set of modality-specific information produced by the backbone into a single lower-dimensional and non-sparse vector that represents the *task-specific information*.

In large-scale multimodal learning [16, 43], aggregating multiple features is a general topic. However, there are two challenges when multiple features are aggregated in a few-data scenario with an irregular input structure configuration. In other words, the multi-feature aggregation in a task-heterogeneous few-shot setting should satisfy two prerequisites: 1) **Efficiency of feature aggregation**. According to [15], multiple modalities contain complementary information, and effective aggregation should be able to combine them while leveraging their potential interactions. Although techniques such as Tensor Fusion Network (TFN) [43] have achieved effective interaction modeling, they basically require large-scale training data due to extensive parameters. We found that, in few-shot learning, such a large model tends to lead to over-fitting and requires more training episodes during adaption. Also, simple methods such as the concatenation are faster and smaller in model size but less powerful in exploring multimodal interactions. Correspondingly, in a heterogeneous few-data scenario, we should balance the *adaption efficiency* and the *exploration of multimodal interactions*. 2) **Task awareness of feature aggregation**. Homogeneous few-shot learning focuses on learning the transferable knowledge generally shared across tasks (i.e., *generalization*). In contrast, in our heterogeneous few-shot setting, each task has a specific combination of modalities so that how to perform multimodal interactions can be very different between tasks. For example, while image classification tasks only analyze image data, a bimodal task containing both image and text should consider image-text interactions. The feature aggregation network shared by all types of tasks should be aware of each task's specific input structure so that it can adaptively perform multimodal interactions. In other words, in addition to generalization, we also should learn how to customize the globally shared meta-learner (i.e., *customization*). That is, under a heterogeneous task distribution, the feature aggregation network should balance *generalization* and *customization*.

Motivated by the two prerequisites, we propose Task-aware Feature Aggregation Network (TFAN), which contains two components: 1) Iterative Feature Aggregation based on the bidirectional recurrent neural network and 2) Attentional Contextual Learning, to achieve the task-awareness of feature aggregation.

Iterative Feature Aggregation Network. As mentioned above, to achieve an effective feature aggregation in low-data scenarios, we pursue a balance between the exploration of multimodal interactions and the adaption speed. We found that the existing large-scale multimodal fusion technique [43, 44] as well as the straightforward feature combinations such as concatenation cannot effectively achieve this goal. Alternatively, we propose to employ the framework of bidirectional recurrent neural network (BRNN) [28] to iteratively combine the information of multiple modalities channel by channel while training parameters for processing only one channel at each step.

Suppose the set of multi-channel encoded modality-specific information $\{\mathbf{z}^{(m)}\}_{m=1}^M$ constructs a sequence with M components $Z = (\mathbf{z}^{(1)}, \mathbf{z}^{(2)} \cdots \mathbf{z}^{(M)})$, where we consider each piece of modality-specific information $\mathbf{z}^{(m)}$ as a *token* as in text embedding. Using the bidirectional long-short term memory (BiLSTM) [12] network as

an example of BRNN, in text embedding, BiLSTM learns a representation for each token by combining with the other words as context. Similarly, given the sequence Z, each modality-specific embedding $\mathbf{z}^{(m)}$ can be combined with the others through the bidirectional modality-by-modality *iterative* calculation process:

$$\begin{aligned}\overrightarrow{\mathbf{h}}^{(m)} &= \overrightarrow{\text{LSTM}}(\mathbf{z}^{(m)}, \overrightarrow{\mathbf{h}}^{(m-1)})\\ \overleftarrow{\mathbf{h}}^{(m)} &= \overleftarrow{\text{LSTM}}(\mathbf{z}^{(m)}, \overleftarrow{\mathbf{h}}^{(m+1)})\\ \mathbf{h}^{(m)} &= \overrightarrow{\mathbf{h}}^{(m)} \oplus \overleftarrow{\mathbf{h}}^{(m)} \in \mathbb{R}^{F_2},\end{aligned} \tag{1}$$

where $\oplus$ denotes concatenation and F_2 is the dimension of hidden state. The iterative feature aggregation network will produce M hidden states: $\mathcal{H} = \{\mathbf{h}^{(m)}\}_{m=1}^{M}$. Each hidden state $\mathbf{h}^{(m)}$ is considered as a view of *aggregated multimodal information* combining all modality-specific information in Z.

The idea of iterative aggregation is to encode multimodal features modality by modality while exploring their interactions step by step. According to the forward pass function of an LSTM's unit

$$\begin{aligned}v_f^m &= \text{sigmoid}(W_f\mathbf{z}^{(m)} + U_f\mathbf{h}^{(m-1)} + b_f)\\ v_i^m &= \text{sigmoid}(W_i\mathbf{z}^{(m)} + U_i\mathbf{h}^{(m-1)} + b_i)\\ v_o^m &= \text{sigmoid}(W_o\mathbf{z}^{(m)} + U_o\mathbf{h}^{(m-1)} + b_o)\\ \widetilde{v_c}^m &= \tanh(W_c\mathbf{z}^{(m)} + U_c\mathbf{h}^{(m-1)} + b_c)\\ v_c^m &= v_f^m \circ v_c^{m-1} + v_i^m \circ \widetilde{v_c}^m\\ \mathbf{h}^{(m)} &= v_o^m \circ \tanh(v_c^m),\end{aligned} \tag{2}$$

where $\{W_f, U_f, b_f\}$, $\{W_i, U_i, b_i\}$, $\{W_o, U_o, b_o\}$, and $\{W_c, U_c, b_c\}$ are the LSTM parameters of the forget gate, the input gate, the output gate, and the memory cell, respectively, at each iterative step, the memory cell's update mechanism enables the feature interaction between each new coming feature $\mathbf{z}^{(m)}$ and previous aggregated information $\mathbf{h}^{(m-1)}$. Note that Eq. (2) is for each direction.

As for adaption efficiency, the number of trainable parameters of the iterative feature aggregation is $O((F_1+F_2)F_2)$, where F_1 and F_2 are dimensions of the embedding vector of each modality and the hidden state of the recurrent network, respectively. Compared with the highly-interacted feature aggregation methods designed for large-scale data, such as the multimodal outer-product containing $O((F_1)^M F_2)$ parameters [43], our model reduces the computation complexity of fusion and thus requires less training episodes and adapts faster. Also, compared with early fusion methods such as [16, 19, 25] containing $O(MF_1F_2)$ parameters, iterative aggregation is more powerful in exploring multimodal interactions while the computation complexity is unrelated with M. In addition, as we also face the task-heterogeneity challenge, iterative aggregation works better in the task-heterogeneous few-shot learning environment.

Attentional Contextual Learning for Task-aware Feature Aggregation. Under a heterogeneous task distribution, we pursue the balance between generalization (globally shared knowledge) and customization (task-specific knowledge). In other words, the feature aggregation network should be aware of each task's specific input structure, and learn how to explore task-specific knowledge to effectively customize the globally shared feature integration process. We propose the attentional contextual learning to achieve such balance, the task-awareness of feature aggregation.

First, since our definition of task heterogeneity is the configuration or availability of a set of input modalities, the sequence $Z = (\mathbf{z}^{(1)}, \mathbf{z}^{(2)} \cdots \mathbf{z}^{(M)})$ can reflect the specific knowledge for the given task's input structure–each modality has its own position in this sequence and the absent modalities can be considered as a special token. In other words, we can view the input structure of tasks as the "context" of the sequence Z, which can distinguish different types of tasks. In text embedding, BRNN models the sequential data by leveraging context dependencies in the input sequence (*context-aware*) [45]. Similarly, when the BRNN-based iterative feature aggregation encodes the sequence of modalities, it can take into account the input structure (context) of the given task, which reflects task-specific knowledge.

Besides, to further customize the iterative feature aggregation network using task-specific knowledge, we propose an external meta-network $g_\phi(\cdot;\phi)$, which learns a task embedding vector τ for each given task to represent the task-specific knowledge as

$$\tau = g_\phi(\mathbf{c}_\mathcal{T}; \phi) \in \mathbb{R}^{F_3}, \tag{3}$$

where F_3 is the dimension of task embedding. We let the task embedding τ represent the type-specific knowledge that reflects each task's specific input space and distinguishes different tasks; hence, τ is embedded from the $\mathbf{c}_\mathcal{T}$ calculated in Section 2.2.1. Then, we utilize τ as extra knowledge to customize the globally shared meta-parameters in the feature aggregation network. Specifically, we employ a modified attention mechanism [1] to facilitate the customization. The final task-specific information is represented by

$$\mathbf{h}^* = \sum_{m=1}^{M} A_m \mathbf{h}^{(m)} \in \mathbb{R}^{F_2}, \tag{4}$$

where $\{A_m\}_{m=1}^{M}$ are attention coefficients indicating the role of each modality with respect to the type-specific knowledge τ:

$$A_m = \frac{\exp(\mathbf{v}^T \tanh(\mathbf{W}_h[\mathbf{h}^{(m)} \oplus \tau]))}{\sum_l \exp(\mathbf{v}^T \tanh(\mathbf{W}_h[\mathbf{h}^{(l)} \oplus \tau]))}, \tag{5}$$

where $\oplus$ denotes concatenation operation, and $\mathbf{v} \in \mathbb{R}^{F_3}$ and $\mathbf{W}_h \in \mathbb{R}^{F_3 \times (F_2+F_3)}$ are learnable parameters of the attention module. The attention mechanism incorporates the type-specific knowledge τ into the feature aggregation, so that it can *adaptively* perform the within-task adaption for different types of tasks.

Overall, our task-aware feature aggregation network (TFAN) transforms the multi-channel modality-specific information into task-specific information. Network parameters in this module include the iterative feature aggregation θ_{iter}, the task-embedding ϕ, and the attention module $\{\mathbf{v}, \mathbf{W}_h\}$.

2.2.3 Single-Channel Head Module. In the previous stage, TFAN adaptively projects samples from different input spaces onto a unified concept space–the generated task-specific representations of each sample in different type of tasks are in the same concept space (see arrow lines in Figure 2(a)). As for final decision making, we use a single-channel head module $f_{\theta_H}(\cdot;\theta_H)$. For few-shot classification, the architecture of the head module is a multi-layer classifier followed by a softmax layer, i.e., $\text{softmax}(f_{\theta_H}(\mathbf{h}^*;\theta_H))$.

2.3 Training Procedure

HetMAML is built upon the model-agnostic meta-learning (MAML) framework [10, 26], which solves a *bilevel optimization* problem to find an *initialization* Θ_0 for a neural network as the meta-parameters.

This training procedure assumes that the single meta-initialization Θ_0 is an appropriate generalization of prior knowledge for all tasks. It requires tasks to be homogeneous in terms of the size of labelled data, the classes number, the input structure, and so on.

Unlike homogeneous MAMLs, our task-heterogeneous model leverages extra task-specific knowledge to perform slightly different adaption between different types of tasks, that is, finding a balance between generalization and customization. Therefore, we divide meta-parameters into two disjoint sets: *internal parameters* and *external parameters*.

2.3.1 Internal and External Parameters. We define *internal parameters* (IP) Θ as the meta-parameters, which take the gradient updates during the inner-loop optimization, to perform the adaption to each specific task. The internal parameters of HetMAML include the meta-initialization of the iterative feature aggregation network and the head module, i.e., $\Theta = \{\theta_{iter}, \theta_H\}$. The task-aware iterative feature aggregation projects input features onto the concept space, by considering the input context of each task. Hence θ_{iter} may vary among different tasks and need to adapt to each task. The head module (classifier) θ_H learns decision boundaries within the concept space. Since different tasks have different concepts, the parameters for dividing the concept space is specific for each task.

External parameters (EP) Φ are defined as the meta-parameters, which pass through the neural network but does not need to adapt to each task during inner-loop optimization. The external parameters of HetMAML include the parameters of multi-channel backbone module, the task embedding network as well as the attention mechanism used for customization, i.e., $\Phi = \{\theta_B, \phi, \mathbf{v}, \mathbf{W}_h\}$. The backbone θ_B can be shared across tasks as it performs early-stage feature embedding of each data source. Meta-parameters in the task embedding network ϕ aim to capture the knowledge of how to distinguish different types of tasks, based on the structure property of each given task. ϕ, $\mathbf{v}$, and $\mathbf{W}_h$ embeds each task's input structure and utilize it to customize the internal parameters during adaption.

2.3.2 Bilevel Optimization. The overall training procedure of HetMAML is described in Algorithm 1. Following [10, 26], the *outer loop* updates the meta-initialization of internal parameters Θ_0 and the external parameters Φ to enable fast adaptation over a batch of task instances. The inner loop takes the outer-loop Θ_0 and Φ, and, separately for each task, performs a few gradient updates of internal parameters over the labelled examples in the support set, while freezing external parameters.

Formally, let Θ_i' signify Θ for task $\mathcal{T}_i$ during the inner-loop optimization, and let the initial $\Theta_i' = \Theta_0$. In the inner-loop adaption, during each gradient update, we compute

$$\Theta_i' \leftarrow \Theta_i' - \alpha \nabla_{\Theta_i'} \mathcal{L}_{\mathcal{T}_i}(f(\mathcal{X}; \Theta_i', \Phi), y; \mathcal{S}_{\mathcal{T}_i}), \tag{6}$$

where $f(\cdot)$ is the forward function of HetMAML network, and $\mathcal{L}_{\mathcal{T}_i}(\cdot; \mathcal{S}_{\mathcal{T}_i})$ is the loss on the support set of task $\mathcal{T}_i$.

Separately for each task, after a fixed number of inner-loop updates, we obtain the adapted parameter $\Theta_i'(\Theta_0)$, which is dependent on meta-initialization Θ_0. Then, the outer-loop optimization updates Θ_0 and Φ over a batch of task instances:

$$\Theta_0 \leftarrow \Theta_0 - \beta \nabla_{\Theta_0} \sum_{\mathcal{T}_i \sim p(\widehat{\mathcal{T}})} \mathcal{L}_{\mathcal{T}_i}(f(\mathcal{X}^*; \Theta_i'(\Theta_0), \Phi), y^*; \mathcal{Q}_{\mathcal{T}_i}) \tag{7}$$

Algorithm 1 Training Procedure of HetMAML

Requires: Heterogeneous task distribution $P(\widehat{\mathcal{T}})$
Requires: Learning rates α, β
Randomly initialize internal parameters Θ.
Randomly initialize external parameters Φ.
//outer-loop optimization
while not done **do**
  Sample batches of heterogeneous tasks $\mathcal{T}_i \sim P(\widehat{\mathcal{T}})$
  // inner-loop optimization
  for all i **do**
    Obtain data $\{\mathcal{S}_{\mathcal{T}_i}, \mathcal{Q}_{\mathcal{T}_i}\}$ for each task $\mathcal{T}_i$.
    Obtain task structure $\mathbf{c}_{\mathcal{T}_i}$ from each $\mathcal{S}_{\mathcal{T}_i}$ as Section 2.2.1.
    Obtain task embedding $\tau_i = g(\mathbf{c}_{\mathcal{T}_i}; \phi)$ using external ϕ.
    Compute adapted internal parameters with a fixed number of steps w.r.t. the NK examples from $\mathcal{S}_{\mathcal{T}_i}$ as in Eq.(6).
    Evaluate $\mathcal{L}_i(f(\mathcal{X}^*; \Theta_i', \Phi), y^*; \mathcal{Q}_{\mathcal{T}_i})$ w.r.t. T samples of $\mathcal{Q}_{\mathcal{T}_i}$.
  end for
  Update initialization of internal parameters Θ_0 as Eq.(7).
  Update external parameters Φ as Eq.(8).
end while
return: Θ_0 and ϕ

$$\Phi \leftarrow \Phi - \beta \nabla_{\Phi} \sum_{\mathcal{T}_i \sim p(\widehat{\mathcal{T}})} \mathcal{L}_{\mathcal{T}_i}(f(\mathcal{X}^*; \Theta_i'(\Theta_0), \Phi), y^*; \mathcal{Q}_{\mathcal{T}_i}), \tag{8}$$

where $\mathcal{L}_{\mathcal{T}_i}(\cdot; \mathcal{Q}_{\mathcal{T}_i})$ is the loss on the query set of task $\mathcal{T}_i$.

3 EXPERIMENTS

In this section, we compare HetMAML with state-of-the-art baselines on six task-heterogeneous few-shot datasets.

3.1 Datasets

Since we define a new task-heterogeneous few-shot learning problem, we constructed our datasets from existing multimodal datasets.

3.1.1 Selection of the Source Multimodal Datasets. One prerequisite for the source multimodal dataset is that the total number of classes $|C|$ should be large. First, we have to split C into two disjoint sets of class set, C^{trn} and C^{tst}, for constructing $\mathcal{D}^{trn}_{meta}$ and $\mathcal{D}^{tst}_{meta}$, respectively. Then, the class set C_i of each task instance is sampled from C. To construct a meta dataset containing a proper number of task instances (for training meta-learner), where the N classes in different task instances do not overlap too much, a large number of total classes is necessary. Another requirement is that, in the source multimodal dataset, the number of subjects (multimodal samples) for each class should also be large to make sure each data sample will not repeat frequently over tasks. Since some multimodal datasets with three or more modalities such as emotion classification datasets have a small number of classes and subjects, we did not consider using them as the source.

We chose the following two datasets as our *source* multimodal datasets: 1) Caltech-UCSD-Birds 200-2011 (CUB-200) [39] dataset contains 11,788 images for 200 bird species, including black footed albatross, bobolink, fish crow, and so on. Each image is annotated with a vocabulary of 312 attributes (e.g., crown color, belly color, eye color, etc.), which we use as the text modality. We used the

	$\lvert C^{trn}\rvert/\lvert C^{tst}\rvert$	M'	Task Input Structures
*het*ModelNet-1	30/10	2	Type 1: X1 Type 2: X2
*het*ModelNet-2	30/10	2	Type 1: X1 Type 2: (X1,X2)
*het*ModelNet-3	30/10	2	Type 1: X2 Type 2: (X1,X2)
*het*ModelNet-4	30/10	3	Type 1: X1 Type 2: X2 Type 3: (X1,X2)
*het*CUB200-1	150/50	2	Type 1: image Type 2: text
*het*CUB200-2	150/50	2	Type 1: image Type 2: (image, text)

Table 1: Statistics of the six task-heterogeneous few-shot datasets (X1: MVCNN modality, X2: GVCNN modality).

two data sources from the CUB-200 dataset: the *image modality* $\mathbf{x}^{image} \in \mathbb{R}^{3\times256\times256}$, and the *text modality* $\mathbf{x}^{text} \in \mathbb{R}^{312}$. 2) ModelNet40 [40] is a 3D object detection dataset containing 12,311 3D CAD shapes covering 40 common categories, including airplane, bathtub, bookshelf, bottle, bowl, cone, cup, and so on. Following [8], we use it as a bimodal dataset where the two modalities are the two views of shape representation extracted from two pre-trained networks: Multi-view Convolutional Neural Network (MVCNN) [33] and Group-View Convolutional Neural Network (GVCNN) [9]. Specifically, the two data sources from the ModelNet40 are: *modality-X1* (MVCNN representation) $\mathbf{x}^{mvcnn} \in \mathbb{R}^{4096}$ and *modality-X2* (GVCNN representation) $\mathbf{x}^{gvcnn} \in \mathbb{R}^{2048}$. For simplicity, we will use "X1" and "X2" to represent the two modalities of ModelNet40.

3.1.2 Construction of Task-Heterogeneous Datasets. From source datasets, we constructed six task-heterogeneous few-shot datasets in total, which were built up as follows. First, we split the class set into two subsets, i.e., $C = \{C^{trn}, C^{tst}\}$, such that $|C^{trn}| = \text{int}(3/4|C|)$ and $C^{tst} = C \setminus C^{trn}$. Then, we generated task instances for each of the meta-train and meta-test datasets: $|\mathcal{D}^{trn}_{meta}| = N_{trn}$ tasks for the meta-train dataset and $|\mathcal{D}^{tst}_{meta}| = N_{tst}$ tasks for the meta-test dataset. Specifically, for constructing each task $\mathcal{T}$ in $\mathcal{D}^{trn}_{meta}$, we randomly selected N classes from C_{trn} and then selected K samples from each class to form the support set; then, we selected *other* K_q labelled samples for each class to form the query set. Finally, we split each of $\mathcal{D}^{trn}_{meta}$ and $\mathcal{D}^{tst}_{meta}$ into M' groups; each group contains one type of tasks–we deleted certain modalities in tasks following the configuration in Table 1. Statistics of the six constructed task-heterogeneous datasets are described below.

***het*CUB200-1 and 2**. Using the multimodal samples in CUB200, we constructed $N_{trn} = 4000$ and $N_{tst} = 1000$ task instances with $K_q = 12$. Each data sample in CUB200 is a pair of (image, text) modalities. We considered the text modality as either the auxiliary modality for image or a view of sample that provides useful information under missing-image conditions. Therefore, we built two datasets; both has two types of tasks–the first type is unimodal tasks that only process images. In *het*CUB200-1, tasks have either image or text modalities. In *het*CUB200-2, the task heterogeneity was created by deleting the text modality from a half of tasks.

***het*ModelNet-1 (2, 3, and 4)**. Using the multimodal samples in ModelNet40, we constructed $N_{trn} = 4000$ and $N_{tst} = 1000$ task instances with $K_q = 12$. We considered different conditions of modality combinations–the modality-X1, the modality-X2, and a pair of modalities (X1,X2). Hence we could construct *four* datasets from ModelNet40, whose input structures are listed in Table 1.

3.2 Baselines

We compared HetMAML with three families of meta-learning baselines. The first is the methods designed for unimodal task distribution: **MAML** [10], **SNAIL** [18], and **LEO** [27]. The second is **MUMOMAML** [38], a method designed for multimodal task distribution. These baselines are limited to homogeneous tasks with the same input structure, and cannot be directly applied to our task-heterogeneous setting. We pre-processed the original heterogeneous inputs such that the processed inputs have the same feature space. The input dimension is fixed to that of the modality which has the largest input feature space. Besides, similar to [38], we used Multi-MAML baselines which consist of multiple MAML models trained separately on each type of tasks. We tested two versions of Multi-MAMLs: **Multi-MAML (TF)** and **Multi-MAML (BF)**. For fair comparisons, we let each MAML also have an intermediate module: while Multi-MAML (TF) uses a Tensor Fusion Network [43] for feature aggregation, Multi-MAML (BF) uses a BRNN network (i.e., BiLSTM) as in HetMAML.

3.3 Results and Discussions

All the experiments in this paper were conducted on a single-core GPU using Pytorch 3. In all experiments, we let $F_1 = 128$, $F_2 = 64$, and $F_3 = 64$. Image backbones used 4 convolutional building blocks with 32 channels. The threshold for calculating $\mathbf{c}$ was set to $\epsilon = 10^{-1}$. As for meta training, the gradient update step of inner-loop adaption was set to 10, and we fixed $\alpha = 10^{-2}$ and $\beta = 10^{-4}$ in all experiments. For the BRNN in TFAN module, we chose to use a BiLSTM network to perform the iterative feature aggregation.

We evaluated HetMAML and baselines based on their performance on the few-shot classification accuracy, and, to evaluate the efficiency of our model, we also report the meta-training time and the size of trainable meta-parameters.

3.3.1 Task-Heterogeneous Few-Shot Classification. Tables 2 and 3 report classification accuracy on each dataset. One can observe that HetMAML outperforms baselines on each dataset in terms of the classification accuracy within a few gradient updates. This demonstrates that HetMAML can successfully handle these heterogeneous task distribution cases and can fast adapt to all types of new tasks. The results on *het*ModelNet-1 where all tasks are unimodal show that HetMAML can achieve the best performance in the cross-modal case of task heterogeneity. On the other datasets which have multimodal tasks, or especially with larger classes or less shots, HetMAML and Multi-MAML (BF) outperforms other baselines, showing that the proposed BRNN-based iterative feature aggregation (IFA) can achieve a proper balance between the efficiency of adaption and effectiveness of multimodal integration so that is suitable for few-shot problems. Multi-MAML (TF) which

Method	***het*ModelNet-1**			***het*ModelNet-2**			***het*ModelNet-3**			***het*ModelNet-4**		
	5-way 1-shot	5-way 5-shot	10-way 1-shot	5-way 1-shot	5-way 5-shot	10-way 1-shot	5-way 1-shot	5-way 5-shot	10-way 1-shot	5-way 1-shot	5-way 5-shot	10-way 1-shot
MAML	80.6	91.2	60.5	80.4	90.6	46.5	89.6	98.7	70.1	84.2	94.7	67.7
SNAIL	82.3	92.6	63.2	81.8	92.4	50.3	91.5	98.8	74.1	85.8	95.1	70.5
LEO	83.1	94.8	68.5	84.2	95.7	55.2	94.8	99.2	80.2	91.2	96.3	74.7
MuMoMAML	82.5	93.3	64.4	84.9	92.6	53.4	93.4	99.3	78.9	89.7	95.6	69.2
Multi-MAML(TF)	75.7	94.2	50.5	68.2	88.6	32.3	77.4	93.3	54.3	74.3	91.4	47.1
Multi-MAML(BF)	76.1	94.3	50.7	80.8	94.3	62.5	92.7	98.9	81.3	83.9	94.5	66.5
HetMAML (ours)	**84.5**	**94.7**	**72.8**	**85.7**	**95.2**	**71.9**	**95.0**	**99.3**	**90.1**	**92.3**	**95.3**	**78.3**

Table 2: Few-shot classification accuracy (%) on the meta-test splits of *het*ModelNet-1 (2, 3, and 4) datasets.

Method	***het*CUB200-1**		***het*CUB200-2**	
	5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot
MAML	43.9	60.2	52.7	69.5
Multi-MAML(TF)	41.3	63.1	39.1	51.2
Multi-MAML(BF)	41.4	62.8	50.3	68.4
HetMAML (ours)	**48.7**	**64.3**	**54.2**	**70.7**

Table 3: Few-shot classification accuracy (%) on the meta-test splits of *het*CUB200-1 and *het*CUB200-2 dataset.

Method	MT-Time	Model Size
Multi-MAML (TF)	26.2 min	2,762,905
Multi-MAML (BF)	34.5 min	1,736,783
HetMAML (ours)	**17.5 min**	**908,485**

Table 4: Comparison of model size and the average meta-training time (MT-Time) on *het*ModelNet-4.

optimizes a large multimodal fusion network failed in few-shot learning scenarios as it requires a much larger parameter set due to its high-dimensional tensor. Instead, HetMAML encodes multi-modalities in a recurrent manner with a relatively small size of parameters. We also observe that the other five homogeneous baselines failed in all task-heterogeneity settings because they cannot customize the globally shared model architecture for tasks having different input structures. In contrast, HetMAML can better balance customization and generalization since the attentional contextual learning (ACL) can automatically take into account the context of tasks' input structures.

In addition, HetMAML outperforms Multi-MAML (BF), especially in 5/10-way 1-shot settings. It is because HetMAML simultaneously trains all types of tasks and thus can access more data from other types of tasks during training. In contrast, each MAML in Multi-MAMLs only uses the smaller number of tasks of a specific type for training these networks. Jointly training heterogeneous tasks could capture more reliable meta-knowledge rather than separately training multiple MAMLs.

3.3.2 Adaption Speed Comparison. We also compare adaption speed in Figure 3, where we display the 10-way 1-shot classification accuracy (meta-test) with respect to the number of gradient updates, separately for each type of tasks on each of the four *het*ModelNet datasets. In this figure, we compare HetMAML with Multi-MAML baselines on two datasets. Other baselines are not compared here because they view different types of tasks as the same type. Solid lines denote the average results of all types of tasks; dash lines denote the results of tasks that only have modality X1 as the input (type-X1); dotted lines denote the results of tasks that only have modality X2 as the input (type-X2); and dash-dotted lines denote tasks that have a pair of modalities as the input (type-(X1,X2)).

Overall, while the baselines need 8-9 or more gradient updates to adapt to new tasks, HetMAML can adapt to new tasks with just 2-3 updates, which demonstrates the fast adaption ability of HetMAML. We can observe that although HetMAML trains all types of tasks simultaneously, it can successfully fast adapt to each type of new tasks, suggesting that the proposed model is aware of the task-specific input structure. In addition, HetMAML outperforms baselines, especially on unimodal tasks, because the training of each types of tasks can benefit from more data from other types. Furthermore, these results are achieved with less training time and parameter size, showing that HetMAML is a more efficient framework than Multi-MAML baselines. The adaption speed of the TFN-based baseline on type-(X1,X2) tasks is very slow (see the blue dash-dotted lines), suggesting that the large-scale multimodal fusion strategy is not suitable to few-shot multimodal tasks. According to the left-2 and left-3 charts, Type-(X1,X2) adaption in Multi-MAML (BF) is faster than the type-X1 or type-X2 adaption, proving the efficiency of iterative aggregation on few-shot multimodal fusion.

3.3.3 Efficiency and Scalability. Table 4 reports the comparison on the model sizes and training time between HetMAML and Multi-MAML. Compared with separate training baselines, our HetMAML uses one-third or a half of trainable meta-parameters to achieve better adaption performance, showing that HetMAML can efficiently and effectively generalize both type-specific and shared meta-parameters across heterogeneous tasks. Moreover, HetMAML can be more scalable than baselines in terms of the number of data sources, in spite that our datasets include only two modalties because existing multimodal datasets having more modalities do not

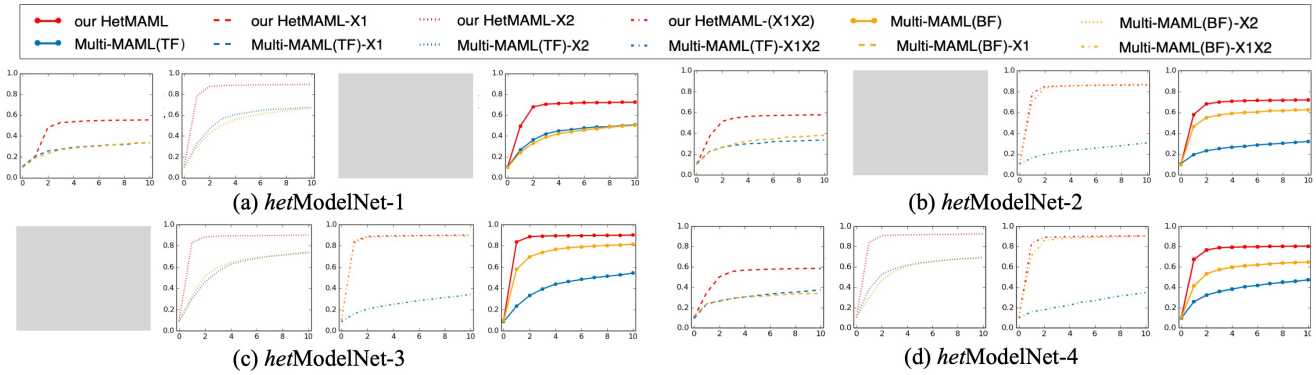


Figure 3: Adaption speed comparison with 10-way 1-shot setting on *het*ModelNet-1 (2, 3, and 4). For each dataset, the left three sub-figures indicate the results on each type of tasks, the right sub-figure indicates the average result of all testing tasks.

have enough samples to construct meta datasets. For HetMAML, the increase of $\triangle M$ modalities will result in $\triangle M$ feature extractors $\{\theta_m | \forall m \in \triangle M\}$ added to the multi-channel backbone, while the other two modules' parameters will remain the same.

4 RELATED WORKS

In this section, we briefly review related works in meta-learning, cross-modal domain adaption, multimodal multi-task learning, and multimodal integration.

Meta-Learning. Meta-learning approaches to few-shot learning problem mainly include gradient-based methods [10, 18, 42, 46] and metric-based methods [14, 20, 30, 34, 37]. While *metric-based* meta-learning learns a distance-based prediction rule over the embeddings, *gradient-based* methods employ a bilevel learning framework that adapts the embedding model parameters given a task's training examples. We follows gradient-based paradigm as the bilevel framework has the potential mechanisms for feature selection and could be more robust to input variations rather than metric-based frameworks. Recent FSL [7, 21, 22, 41] extended to multimodal few-shot scenarios. Yet most of these methods assume homogeneous multimodal features, where input samples over tasks consist of the same set of modalities. In contrast, we focus on the case that some modalities may not be available in some tasks and thus different tasks may have different combinations of modalities. Recently, task-adaptive meta-learning [13, 27, 35, 38] considered task distribution across domains, but different tasks still share the same input space. Different from these methods that assume task homogeneity, we study a novel FSL approach under heterogeneous task distributions.

Cross-modal Domain Adaption. Domain Adaptation (DA) is useful in the situations where the source and target domains have the same classes but the input distribution of the target task is shifted with respect to the source task. Cross-modal DA deals with the special case that the inputs of the source and target differ in modalities [24], which is similar to the cross-modal case of our task heterogeneity. However, cross-modal DA focuses on the adaption between two domains (source and target). In contrast, this paper learns meta-knowledge for unlimited unseen tasks. There is no meta-objective in DA that optimizes "how to learn" across tasks.

Multimodal Multi-Task Learning. Multi-Task Learning (MTL) aims to jointly learn several related tasks such that each task can benefit from parameters or representation sharing across different tasks. Multimodal MTL [5, 23] deals with using MTL methods to solve multimodal integration problem. Our HetMAML also attempts to design a parameter-sharing strategy so that all types of tasks can be learned with a unified framework. However, we aim to obtain meta-knowledge across unlimited tasks leveraging these shared parameters, whereas MTL intends to solve a fixed number of known tasks through a single-level optimization without a meta-objective.

Multimodal Integration. Deep multimodal integration studies how to align or integrate different modalities (e.g., visual, language, acoustic, or others) into a joint embedding. Existing methods include early fusion [11, 19, 25, 31], late fusion that models the dynamics of multimodal interactions [4, 17, 43], and multimodal sequential learning [3, 32, 44]. A majority of existing multimodal fusion models rely on large-scale training data so that their performance may drop dramatically in the few-data scenario. Also, most works build frameworks designed for single-type tasks, which are not very robust to heterogeneous few-shot tasks. Although one can impute unavailable modalities using imputation strategies [2, 6, 29, 36] to convert the problem into homogeneity, this may introduce extra noise to the original few-data task, and also, the multimodal integration function may still vary between different types of tasks. In contrast, HetMAML directly learns with different input spaces and can be aware of the input structures of multimodal tasks to promote knowledge customization.

5 CONCLUSIONS

In this paper, we introduced HetMAML, a novel task-heterogeneous meta-agnostic meta-learning approach to few-shot learning under heterogeneous task distribution. To effectively capture both the type-specific and globally shared meta-parameters, we designed a three-module framework, including a multi-channel backbone to extract modality-specific embeddings, a task-aware feature aggregation network to adaptively transform these multimodal embeddings into a task-specific representation by leveraging the task' type-specific knowledge, and a single-channel head module for decision making. Experiment results show that HetMAML is able to effectively and efficiently capture meta-parameters across heterogeneous tasks, balance customization and generalization, and successfully fast adapt to all types of new tasks.

REFERENCES

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1409.0473

[2] Lei Cai, Zhengyang Wang, Hongyang Gao, Dinggang Shen, and Shuiwang Ji. 2018. Deep adversarial learning for multi-modality missing data completion. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM.

[3] Stanislas Chambon, Mathieu N Galtier, Pierrick J Arnal, Gilles Wainrib, and Alexandre Gramfort. 2018. A deep learning architecture for temporal sleep stage classification using multivariate and multimodal time series. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 26, 4 (2018), 758–769.

[4] Jiayi Chen and Aidong Zhang. 2020. HGMF: Heterogeneous Graph-based Fusion for Multimodal Data with Incompleteness. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1295–1305.

[5] Shizhe Chen, Qin Jin, Jinming Zhao, and Shuai Wang. 2017. Multimodal multi-task learning for dimensional and continuous emotion recognition. In *Proceedings of the 7th Annual Workshop on Audio/Visual Emotion Challenge*. 19–26.

[6] Changde Du, Changying Du, Hao Wang, Jinpeng Li, Wei-Long Zheng, Bao-Liang Lu, and Huiguang He. 2018. Semi-supervised deep generative modelling of incomplete multi-modality emotional data. In *Proceedings of the 26th ACM international conference on Multimedia*. 108–116.

[7] Ryan Eloff, Herman A Engelbrecht, and Herman Kamper. 2019. Multimodal one-shot learning of speech and images. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 8623–8627.

[8] Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. 2019. Hypergraph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 3558–3565.

[9] Yifan Feng, Zizhao Zhang, Xibin Zhao, Rongrong Ji, and Yue Gao. 2018. GVCNN: Group-view convolutional neural networks for 3D shape recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 264–272.

[10] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 1126–1135.

[11] Hatice Gunes and Massimo Piccardi. 2005. Affect recognition from face and body: early fusion vs. late fusion. In *2005 IEEE international conference on systems, man and cybernetics*, Vol. 4. IEEE, 3437–3443.

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

[13] Bingyi Kang and Jiashi Feng. 2018. Transferable Meta Learning Across Domains. In *UAI*. 177–187.

[14] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. 2015. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, Vol. 2. Lille.

[15] Dana Lahat, Tülay Adali, and Christian Jutten. 2015. Multimodal Data Fusion: An Overview of Methods, Challenges, and Prospects. *Proc. IEEE* 103, 9 (2015), 1449–1477. https://doi.org/10.1109/JPROC.2015.2460697

[16] Mingxia Liu, Yue Gao, Pew-Thian Yap, and Dinggang Shen. 2017. Multi-Hypergraph Learning for Incomplete Multimodality Data. *IEEE journal of biomedical and health informatics* 22, 4 (2017), 1197–1208.

[17] Zhun Liu, Ying Shen, Varun Bharadhwaj Lakshminarasimhan, Paul Pu Liang, AmirAli Bagher Zadeh, and Louis-Philippe Morency. 2018. Efficient Low-rank Multimodal Fusion With Modality-Specific Factors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2247–2256.

[18] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. 2018. A Simple Neural Attentive Meta-Learner. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net. https://openreview.net/forum?id=B1DmUzWAW

[19] Louis-Philippe Morency, Rada Mihalcea, and Payal Doshi. 2011. Towards multimodal sentiment analysis: Harvesting opinions from the web. In *Proceedings of the 13th international conference on multimodal interfaces*. 169–176.

[20] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. 2018. Tadam: Task dependent adaptive metric for improved few-shot learning. In *Advances in Neural Information Processing Systems*. 721–731.

[21] Frederik Pahde, Oleksiy Ostapenko, Patrick Jä Hnichen, Tassilo Klein, and Moin Nabi. 2019. Self-paced adversarial training for multimodal few-shot learning. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 218–226.

[22] Frederik Pahde, Mihai Puscas, Tassilo Klein, and Moin Nabi. 2020. Multimodal Prototypical Networks for Few-shot Learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2644–2653.

[23] Srinivas Parthasarathy and Carlos Busso. 2017. Jointly Predicting Arousal, Valence and Dominance with Multi-Task Learning.. In *Interspeech*. 1103–1107.

[24] Jose Costa Pereira and Nuno Vasconcelos. 2014. Cross-modal domain adaptation for text-based regularization of image semantics in image retrieval systems. *Computer Vision and Image Understanding* 124 (2014), 123–135.

[25] Verónica Pérez-Rosas, Rada Mihalcea, and Louis-Philippe Morency. 2013. Utterance-level multimodal sentiment analysis. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 973–982.

[26] Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. 2019. Rapid learning or feature reuse? towards understanding the effectiveness of maml. *arXiv preprint arXiv:1909.09157* (2019).

[27] Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. 2019. Meta-Learning with Latent Embedding Optimization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net. https://openreview.net/forum?id=BJgklhAcK7

[28] Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing* 45, 11 (1997), 2673–2681.

[29] Chao Shang, Aaron Palmer, Jiangwen Sun, Ko-Shin Chen, Jin Lu, and Jinbo Bi. 2017. VIGAN: Missing view imputation with generative adversarial networks. In *Big Data (Big Data), 2017 IEEE International Conference on*. IEEE.

[30] Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*. 4077–4087.

[31] Cees GM Snoek, Marcel Worring, and Arnold WM Smeulders. 2005. Early versus late fusion in semantic video analysis. In *Proceedings of the 13th annual ACM international conference on Multimedia*. 399–402.

[32] Yale Song, Louis-Philippe Morency, and Randall Davis. 2012. Multimodal human behavior analysis: learning correlation and interaction across modalities. In *Proceedings of the 14th ACM international conference on Multimodal interaction*. 27–30.

[33] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. 2015. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE international conference on computer vision*. 945–953.

[34] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. 2018. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1199–1208.

[35] Qiuling Suo, Jingyuan Chou, Weida Zhong, and Aidong Zhang. 2020. Tadanet: Task-adaptive network for graph-enriched meta-learning. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1789–1799.

[36] Luan Tran, Xiaoming Liu, Jiayu Zhou, and Rong Jin. 2017. Missing modalities imputation via cascaded residual autoencoder. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1405–1414.

[37] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. 2016. Matching networks for one shot learning. *arXiv preprint arXiv:1606.04080* (2016).

[38] Risto Vuorio, Shao-Hua Sun, Hexiang Hu, and Joseph J Lim. 2019. Multimodal Model-Agnostic Meta-Learning via Task-Aware Modulation. In *Advances in Neural Information Processing Systems*. 1–12.

[39] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. 2011. *The Caltech-UCSD Birds-200-2011 Dataset.* Technical Report CNS-TR-2011-001. California Institute of Technology.

[40] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 2015. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1912–1920.

[41] Chen Xing, Negar Rostamzadeh, Boris Oreshkin, and Pedro O O Pinheiro. 2019. Adaptive cross-modal few-shot learning. *Advances in Neural Information Processing Systems* 32 (2019), 4847–4857.

[42] Jaesik Yoon, Taesup Kim, Ousmane Dia, Sungwoong Kim, Yoshua Bengio, and Sungjin Ahn. 2018. Bayesian model-agnostic meta-learning. In *Advances in Neural Information Processing Systems*. 7332–7342.

[43] Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2017. Tensor Fusion Network for Multimodal Sentiment Analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 1103–1114.

[44] Amir Zadeh, Paul Pu Liang, Navonil Mazumder, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2018. Memory fusion network for multi-view sequential learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[45] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 207–212.

[46] Luisa Zintgraf, Kyriacos Shiarli, Vitaly Kurin, Katja Hofmann, and Shimon Whiteson. 2019. Fast context adaptation via meta-learning. In *International Conference on Machine Learning*. PMLR, 7693–7702.