Topological Transduction for Hybrid Few-shot Learning

Jiayi Chen
University of Virginia
Charlottesville, Virginia, USA
jc4td@virginia.edu

Aidong Zhang
University of Virginia
Charlottesville, Virginia, USA
aidong@virginia.edu

ABSTRACT

Digging informative knowledge and analyzing contents from the internet is a challenging task as web data may contain new concepts that are lack of sufficient labeled data as well as could be multimodal. Few-shot learning (FSL) has attracted significant research attention for dealing with scarcely labeled concepts. However, existing FSL algorithms have assumed a uniform task setting such that all samples in a few-shot task share a common feature space. Yet in the real web applications, it is usually the case that a task may involve multiple input feature spaces due to the heterogeneity of source data, that is, the few labeled samples in a task may be further divided and belong to different feature spaces, namely hybrid few-shot learning (hFSL). The hFSL setting results in a hybrid number of shots per class in each space and aggravates the data scarcity challenge as the number of training samples per class in each space is reduced. To alleviate these challenges, we propose the Task-adaptive Topological Transduction Network, namely TopoNet, which trains a heterogeneous graph-based transductive meta-learner that can combine information from both labeled and unlabeled data to enrich the knowledge about the task-specific data distribution and multi-space relationships. Specifically, we model the underlying data relationships of the few-shot task in a node-heterogeneous multi-relation graph, and then the meta-learner adapts to each task's multi-space relationships as well as its inter- and intra-class data relationships, through an edge-enhanced heterogeneous graph neural network. Our experiments compared with existing approaches demonstrate the effectiveness of our method.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning approaches**; *Multi-task learning*; *Classification and regression trees.*

KEYWORDS

multimodal content analysis, few-shot learning, semi-supervised learning, graph neural networks

ACM Reference Format:
Jiayi Chen and Aidong Zhang. 2022. Topological Transduction for Hybrid Few-shot Learning. In *Proceedings of the ACM Web Conference 2022 (WWW '22), April 25–29, 2022, Virtual Event, Lyon, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3485447.3512033

WWW '22, April 25–29, 2022, Virtual Event, Lyon, France

ACM ISBN 978-1-4503-9096-5/22/04…$15.00
https://doi.org/10.1145/3485447.3512033

1 INTRODUCTION

With the rapid growth of the World Wide Web, data over the internet is enormous and will continue to increase. Manually analyzing and fetching relevant information from such massive data is not only time consuming but also impossible, which makes it imperative to develop *intelligent* Web mining and Web content analysis using machine learning and deep learning techniques [2, 14, 16]. However, applying machine learning to analyze data and mine informative knowledge from the internet is a challenging task. One of challenges is that some concepts on the internet may not have enough labeled data. For example, due to the succeeding evolution of the Web, new concepts appear frequently day by day (e.g., new restaurants, new techniques or tools, and newly discovered animals) but may not have sufficient annotations. Since the success of deep learning-based web mining highly relies on large amounts of labeled data and exhaustive training, the lack of sufficient annotations makes it difficult to learn from scarcely labeled web concepts.

Few-shot learning (FSL) has recently received much attention due to its appealing ability of learning from few labeled data [8, 18, 21, 22, 22, 28, 29, 32, 43], which has potentials to improve web content analysis, especially for scarcely labeled contents without much annotations. The main purpose of FSL is to quickly learn new concepts from a handful of examples by leveraging context and prior knowledge, which simulates humans capabilities of understanding a concept. As defined in [8], a few-shot *task* refers to a training-and-testing process, aiming to learn a class distribution over the data within this task, under the supervision of a *small* set of labeled training data (support set), and then test on a set of unlabeled testing data (query set). Figure 1(a) illustrates an example of an N-way K-shot classification task, where there are N classes needed to learn from the K labeled samples per class. Approaches to FSL typically follow the *meta-learning* paradigm–given experiences on solving few-shot tasks over a set of base classes, meta-learning aims to extract domain-general information that can act as *prior knowledge* (also known as meta-knowledge) to improve learning efficiency and performance in novel concepts [12].

However, most existing FSL approaches assume a well-defined *uniform* few-shot task setting, where all the samples within a task possess identical feature space. For example, as shown in Figure 1(a), in a standard N-way K-shot single-modal classification task, all samples' modalities are the same and have the same feature space. Such assumption of existing FSL methods will limit their applicability in the web domain, where data is more complex, multimodal, and non-identically distributed. Web contents delivered to users are in different forms like videos, images, texts, audios, and so on, and usually concepts are represented by a combination of more than one modality. For instance, 'cat' can be an image, a piece of descriptions, or a video with the caption.

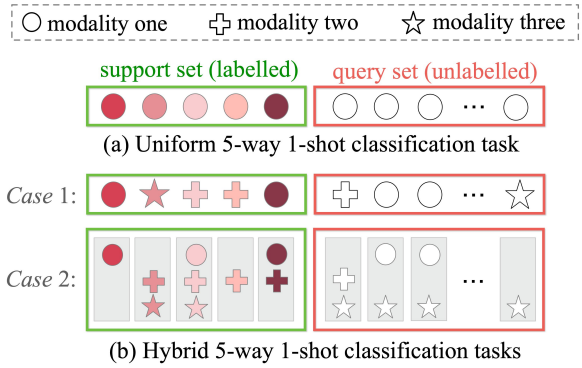

Figure 1: Comparison between uniform and hybrid FSL. Colored shapes within green rectangles denote training samples. In each task, different colors indicate the different classes (concepts). Hollow shapes within red rectangles denote unlabeled testing points. Note that in case-2, each of the grey regions denotes a data sample, where a sample may contain more than one modality.

Since web data can be heterogeneous and hybrid, the difficulty of applying FSL on web is how to deal with the heterogeneous information. To bridge the gap, this paper defines and studies a novel *hybrid few-shot learning* (hFSL) problem neglected by previous literature, which is the opposite of uniform FSL. In hFSL, we consider that a concept can be represented by different types of data or different combinations of modalities. That is, the samples in a few-shot task can be further divided and distributed in separate feature spaces. Figure 1(b) shows two examples of hybrid 5-way 1-shot classification tasks, where samples may be diverged from each other in terms of their feature spaces. The hybrid FSL could be an inevitable problem in the web domain. Web data may be frequently absent or inaccessible, thus uniform multimodal FSL usually turns into hFSL as some modalities may be missing under the web scenarios. The multimodal few-shot tasks with irregularly missing modalities can be typical hybrid few-shot tasks.

A key property of hFSL is the *heterogeneity of data* due to the existence of multiple input feature spaces, which leads to two challenges. First, compared with uniform FSL, the ***data scarcity*** problem would be escalated in hFSL. Specifically, since the few labeled samples per class may be spread in different feature spaces, in *each space*, there would be less labeled data per class (i.e., *less shots*) or even no training data available in some classes (i.e., *zero shot*). That is, the number of training samples in each space may be reduced. For example, consider the task shown in Figure 1(b) case-1, there is no modality-one training data in class-2, thus the class-2 for modality-one is a zero-shot case. Second, for a hybrid K-shot classification task, the *uneven* split of K examples per class would result in a hybrid number of labeled samples per class (i.e., ***hybrid shots***) in *each space*. Typically, a model is designed for training and testing data that has the same input space. However, the decreased and hybrid number of training examples in each space may bring difficulties to the model training. Although one may consider training an alignment function to unify the training data from heterogeneous spaces, the accuracy of such a task-specific alignment function still relies on the limited number of support examples in each input space.

To alleviate the data-scarcity and hybrid-shot problems of hFSL, we propose to formulate the hybrid few-shot task as a transductive learning task, which maximally leverages available information in the task to enrich our knowledge about the target concepts, while learning the potential relationships between heterogeneous data. Transductive inference for few-shot learning typically utilizes the query samples to improve the task-specific knowledge distillation [20, 25, 45]. Inspired by this, we propose a transductive meta-learner which can incorporate some unlabeled data containing information that is not possessed in the labeled samples. Intuitively, our key idea is to jointly learn all the samples in the task with heterogeneous spaces so that the model can obtain extra information (from unlabeled data) about the relationships between spaces and the data distribution to make better predictions. In particular, we aim to learn the task-specific relationships 1) between heterogeneous input spaces and 2) between samples within the same class (intra-class samples) or belonging to different classes (inter-class samples), where the underlying data relationships within a task are complicated and hard to be learned due to data heterogeneity.

To achieve these goals, we propose **T**ask-adaptive **Topo**logical **T**ransduction **Net**work (**TopoNet**), a graph neural network-based transductive few-shot learning framework for hFSL. Basically, we introduce a topological transductive meta-learner, which can learn the task's class distribution by simultaneously exploring relationships between concepts as well as relationships between the heterogeneous feature spaces of data. We explicitly model a graph structure to connect all the samples in a task to perform the transduction; edges expressively connect inter- and intra-class samples as well as bridge heterogeneous samples, which helps to leverage multi-space relationships and data semantic similarities. To capture both the multi-space relationships and the inter- and intra-class data relationships, we first construct a *node-heterogeneous multi-relation graph* from the original multi-space features, and then we propose the *edge-enhanced heterogeneous graph neural network* to alternatively update edge and node features layer by layer, where heterogeneous input spaces are gradually unified while leveraging the edge features to incorporate inter- and intra-class relationships. Our contributions are summarized as follows.

- We study a novel hybrid few-shot learning problem, where a task involves multiple feature spaces and contains a hybrid number of shots per class in each space. As far as we know, we are the first to consider the data heterogeneity issue under few labeled situations and aim at learning new concepts from scarcely labeled and heterogeneous web contents.
- We propose TopoNet to overcome the data-scarcity and hybrid-shot challenges in hFSL by modeling a learnable and generalizable topological structure.
- The experimental results on both uniform and hybrid few-shot tasks demonstrate that our framework is superior to existing approaches.

2 RELATED WORK

Meta-Learning for Few-shot Classification. Recent meta-learning approaches can be divided into two categories: *inductive* and *transductive* few-shot classification. Inductive few-shot learning has been

more widely studied than transductive few-shot learning. Inductive methods mainly includes metric-based and optimization-based algorithms. *Metric-based approaches* learn an embedding metric space shared by all tasks, on which data samples of different classes can distinguish with each other based on distance measurements [22, 28, 29, 32]. *Optimization-based approaches* train a meta-learner as an optimizer to fine-tune the meta-prior, thus adapt the class distribution to each specific task [8, 18, 21, 43]. Further, several works [22, 33, 41] improved the metric- or optimization-based methods in terms of task adaptability. While these approaches presume the samples in a task share a uniform input space, we assume a hybrid task setting involving a mixture of different input spaces. Some recent works studied *multimodal few-shot* learning [4, 23, 39]. Although we also use multimodal few-shot datasets, we allow for the frequent occurrence and different conditions of missing modalities in real-world multimodal few labeled data scenarios.

Transductive Inference. Transductive learning was first introduced in [30]. A family of transductive methods were built upon graph learning frameworks, such as graph propagation [35] and graph neural networks (GNN) [3, 36].

Transductive inference has been recently used to solve few-shot tasks, which has shown substantial improvements over inductive counterparts as it utilizes unlabeled query data to obtain more representative class distribution. Based on how the model incorporates unlabeled data, existing transductive approaches can be separated into implicit and explicit methods. *Implicit transductive* methods directly use the entire unlabeled feature information to enhance the classification boundaries [1, 6, 21, 25]. While implicit methods do not leverage data relationships during transduction, *explicit transductive* methods measures the underlying relationships between data to enrich class features [11, 13, 15, 20, 26, 45] Our method follows the explicit transductive paradigm in the sense that we also explore data relationships during within-task transductive adaption. However, existing transductive methods rely on a common metric space to measure data relationships. Yet this assumption does not hold in the hybrid few-shot setting with heterogeneous input spaces. This paper mainly deals with the difficulties from the division of samples (data heterogeneity), where the relationships between data could be more complicated and unclear.

Meta-learning for Graphs. Our framework utilizes Graph Neural Networks (GNNs) [36, 37] for solving hybrid few-shot tasks. Yet we focus on jointly learning the graph structure and node representations, as well as how to generalize and adapt the learnable structure over tasks. Some works [10, 46] proposed techniques for optimizing graph structures together with GNN parameters using meta-gradients, reinforcement learning, or discrete edge probabilities, but studied different problems (e.g., completing corrupted edges and adversarial attacks) on a single large-scale graph. Recent works incorporated graph structured data into meta-learning [5, 40, 42, 44]. We also formulate our problem as graph-structured semi-supervised node classification tasks plugged into meta-learning. However, these methods assume a single large-scale graph whose structure is given. In contrast, the graph structure of our task is not given, and moreover, we generalize the graph structure knowledge across unlimited graphs and adapt the graph learning procedure over different tasks.

3 PROBLEM FORMULATION

A few-shot learning task $\mathcal{T} = \{\mathcal{S}_\mathcal{T}, \mathcal{Q}_\mathcal{T}\}$ consists of a small task-level training (support) dataset $\mathcal{S}_\mathcal{T}$ and a testing (query) dataset $\mathcal{Q}_\mathcal{T}$. As for classification problems, a task $\mathcal{T}$ aims to learn a task-specific class distribution over the data within this task, supervised by the few labeled examples in $\mathcal{S}_\mathcal{T}$.

Existing FSL approaches [8, 9] mainly assume identically distributed data, called *uniform few-shot learning* (uFSL). As defined in these algorithms, a standard uniform N-way K-shot classification task $\mathcal{T} = \{\mathcal{S}_\mathcal{T}, \mathcal{Q}_\mathcal{T}\}$ contains the *support set* $\mathcal{S}_\mathcal{T} = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), ..., (\boldsymbol{x}_{N\times K}, y_{N\times K})\}$ that includes K labeled samples from each of the N classes, and the *query set* $\mathcal{Q}_\mathcal{T} = \{(\boldsymbol{x}_1^*, y_1^*), (\boldsymbol{x}_2^*, y_2^*), ..., (\boldsymbol{x}_T^*, y_T^*)\}$ that includes T different samples from the same N classes: The *uniformity* of a task refers to the consistency of input feature spaces of each sample $\forall \boldsymbol{x}_i$ in $\mathcal{T}$, that is, the input samples in a task share a common feature space. In particular, as for *single-modal few-shot learning* (SFSL) where data is collected from one type of sensor, each sample is a single modality $\boldsymbol{x}_i \in \mathbb{R}^d$, where d denotes the dimension of the shared input feature space. In multimodal applications, *uniform multimodal few-shot learning* (uMFSL) assumes the data of all modalities are available for each sample, that is, each sample consists of a set of M modalities $\boldsymbol{x}_i = (\boldsymbol{x}_{i,m} \in \mathbb{R}^{d_m} | m = 1, 2, ..., M)$. Here $\boldsymbol{x}_{i,m}$ denotes the mth-modality in the tuple $\boldsymbol{x}_i$, and d_m is the dimension of modality-m's feature space. Since the input modality set is complete, all samples $\forall \boldsymbol{x}_i$ in $\mathcal{T}$ share a composite space–a combination of M feature subspaces.

3.1 Hybrid Few-shot Learning

We define a non-uniform and more complex FSL setting, namely *hybrid few-shot learning* (hFSL). hFSL specifies a real-world case where the support/query samples of a concept in the task are separated in different feature spaces while remaining semantic-level similarities.

Definition 1 (*Hybrid Few-shot Classification*). Different from uniform few-shot classification, each input sample $\boldsymbol{x}_i$ in a hybrid task $\mathcal{T}$ is associated with an additional indicator $\mathcal{B}_i$ to specify its feature space. The *support* and *query* sets of a hybrid N-way K-shot classification task are defined as

$$\begin{aligned} \mathcal{S}_\mathcal{T} &= \{(\boldsymbol{x}_1, y_1, \mathcal{B}_1), ..., (\boldsymbol{x}_{N\times K}, y_{N\times K}, \mathcal{B}_{N\times K})\} \\ \mathcal{Q}_\mathcal{T} &= \{(\boldsymbol{x}_1^*, y_1^*, \mathcal{B}_1^*), ..., (\boldsymbol{x}_T^*, y_T^*, \mathcal{B}_T^*)\}. \end{aligned} \tag{1}$$

Assume there is a finite number (U) of input spaces over the task domain, and suppose each $u \in \{1, 2 \cdots U\}$ indicates a specific space. We are given a heuristic function $ptr(\cdot)$ to recognize the space $u_i = ptr(\mathcal{B}_i)$ of each sample $\boldsymbol{x}_i$ by $\mathcal{B}_i$.

In this paper, we particularly focus on the hFSL in multimodal domain. In contrast to uMFSL, we consider that collecting a complete set of all modalities for each sample, especially in low-data scenarios where data is more expensive, could be difficult [24]. Hence even the samples in the same task may have irregular missing modalities. Specifically, suppose the original data is collected from M modalities, an input sample of a uMFSL task consists of a set of obtainable modalities denoted by $\mathcal{B}_i$,

$$\boldsymbol{x}_i = (\boldsymbol{x}_{i,m} \in \mathbb{R}^{d_m} | m \in \mathcal{B}_i \subseteq \overline{\mathcal{B}}), \tag{2}$$

where $\overline{\mathcal{B}} = \{1, ..., M\}$ signifies a complete modality set, and the subset $\mathcal{B}_i \subseteq \overline{\mathcal{B}}$ indicates the $\boldsymbol{x}_i$'s feature space by specifying the

attainability of each modality. Therefore, each input feature space-u refers to a specific *combination* of M subspaces, with a total of $U = \binom{M}{1} + \binom{M}{2} + \cdots + \binom{M}{M} = 2^M - 1$ spaces over the task domain. Each task have spaces $U_{\mathcal{T}} \leq U$ spaces. Note that if $\mathcal{B}_i = \overline{\mathcal{B}}$ for all $\forall \mathcal{B}_i$ in $\mathcal{T}$, the hybrid task becomes uniform.

3.2 Meta-Learning

We consider a task distribution $P(\mathcal{T})$ over few-shot learning tasks. Our meta-objective is to train a *meta-learner* p_θ to adapt to $P(\mathcal{T})$, i.e., the meta-learner should be able to solve any few-shot task $\mathcal{T} \sim P(\mathcal{T})$ supervised by the few labeled samples in $\mathcal{T}$. In most of existing *inductive* FSL frameworks, the meta-learner adapts to each task $\mathcal{T}$ relying on the knowledge from the support set $p_\theta(y^*|\boldsymbol{x}^*; \mathcal{S}_\mathcal{T})$. In practice, we are given a set of *meta-training* few-shot tasks $\mathcal{D}^{meta}_{train} = \{\mathcal{T}_1, \mathcal{T}_2, \cdots \mathcal{T}_{N_{trn}}\}$ with a set of base classes C_{train}, where each meta-training task $\mathcal{T} \sim P(\mathcal{T})$ learns from a subset of N-way classes sampled from C_{train} with a few labeled samples per class. The meta-learner is trained from $\mathcal{D}^{meta}_{train}$ to be able to fast adapt to *new* tasks whose classes are held out (unseen) during meta-training.

4 METHODOLOGY

In a hybrid few-shot classification task, as defined in Eq.(1), data are heterogeneous in terms of the inconsistent feature spaces of data. That is, the limited labeled samples per class (i.e., K shots) can be further partitioned by the different feature spaces. Therefore, each space u only contains partial labeled samples for each class, which leads to two subproblems: 1) the ***data-scarcity*** problem is aggravated such that the number of training samples per class in each space is reduced to less shots or zero shot; 2) the ***hybrid-shot*** problem, where different classes have different number of training samples in each space u, as the K examples per class have been *unevenly* split.

To overcome these challenges, we propose to employ the transductive inference for task adaption. We aim to train a *transductive meta-learner* that jointly considers the knowledge about heterogeneous data in both $\mathcal{S}_\mathcal{T}$ and $\mathcal{Q}_\mathcal{T}$:

$$p_\theta(y^*|x^*, \mathcal{S}_\mathcal{T}, \mathcal{Q}_\mathcal{T} \setminus \mathcal{Y}_{\mathcal{Q}_\mathcal{T}}), \tag{3}$$

where $\mathcal{Y}_{\mathcal{Q}_\mathcal{T}}$ denotes the ground-truth labels of query samples in $\mathcal{T}$, which means the labels of query set are not required for solving each task, which is the truth in reality. An assumption underlying Eq. (3) is that we know partial testing (query) samples for solving a task. This assumption yet holds in meta-learning framework as $\mathcal{D}^{meta}_{train}$ contains the query data of each task to enable the training of meta-learner.

Intuitively, during the transductive task adaption, we incorporate unlabeled samples and jointly learn all the samples in the task with heterogeneous spaces, so that the meta-learner can obtain extra information about the task-specific data distribution and the relationships between spaces to make better predictions. In this section, we will introduce the proposed the Task-adaptive Topological Transduction Network (TopoNet), whose overview is in Figure 2.

4.1 Feature Embedding

A feature embedding network $f_e(\cdot; \theta_e)$ is used to extract features of an input $\boldsymbol{x}_i$, where θ_e indicates its parameters. Suppose there are M modalities, f_e contains M paralleled modality-specific subnetworks $f_e^1, f_e^2, ..., f_e^M$. Each *existing* (not missing) modality $\boldsymbol{x}_{i,m}$ of a sample $\boldsymbol{x}_i$ is embedded independently through the subnetwork $f_e^m(\cdot; \theta_e^m)$,

$$\boldsymbol{z}_{i,m} = f_e^m(\boldsymbol{x}_{i,m}; \theta_e^m) \in \mathbb{R}^F, \tag{4}$$

where F is the dimension of each modality's embedding. Hence, each sample $\boldsymbol{x}_i$ is embedded as a tuple containing $|\mathcal{B}_i|$ modality-specific embeddings, $\boldsymbol{z}_i = (\boldsymbol{z}_{i,m} | m \in \mathcal{B}_i)$. With the transductive inference that jointly learns support and query data in task $\mathcal{T}$, we will obtain an *embedded feature set* for all support and query samples within the task, i.e., $\mathcal{Z} = \{\boldsymbol{z}_i | \forall \boldsymbol{x}_i \in \mathcal{S}_\mathcal{T} \bigcup \mathcal{Q}_\mathcal{T}\}$. Note that for uniform multimodal FSL, f_e will generate a feature set with fixed number of embeddings per sample so that $\mathcal{Z} = \boldsymbol{Z} \in \mathbb{R}^{(NK+T) \times MF}$.

4.2 Topological Transductive Learning

To overcome the hybrid-shot and data-scarcity dilemma of hFSL, our key idea is to build connections and unify all different types of samples in a task during model training. Therefore, we consider the transductive inference (as in Eq. (3)) that can jointly learn support and query samples from multiple input spaces. In this transductive framework, we focus on solving two subproblems: 1) how to explore the relationships between multiple input spaces so that samples can be aligned in a uniform semantic space; 2) how to discover the inter- and intra-class data relationships and then utilize them to improve the representativeness of the learned class distribution.

To facilitate the exploration of data and multi-space relationships within the transductive learning framework, we propose to explicitly model a learnable graph structure to connect all the samples in a task. We consider the input set of a task is believed to have some geometric structure, and the edges (topology) of a graph structure can naturally connect different input spaces, as well as leverage the potential inter- and intra-class data relations of the task. Therefore, given a task $\mathcal{T}$, our goal is to learn its *underlying topological graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E}; \mathcal{T})$, which represents the relations among the support and query samples within the task. $\mathcal{V} = \{v_i\}_{i=1}^{NK+T}$ denotes the vertex set combining support and query samples, and $\mathcal{E} = \{e_{ij}\}_{i,j=1}^{NK+T}$ is the edge set which connects each pair of samples from different classes and different input spaces. Each node v_i is associated with a node feature $\boldsymbol{h}_i$, and each edge e_{ij} is also associated with an edge feature/weight $\boldsymbol{e}_{i,j}$ which is relevant to node relationships.

Solving an hFSL task can be viewed as learning the node and edge features of graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}; \mathcal{T})$. We formulate such graph learning task as a *semi-supervised node classification task*, supervised by the $|\mathcal{S}_\mathcal{T}|$ labeled nodes. In this section, we will first construct a multi-relation graph with its initialized node and edge features, and then, edge and node features are refined step-by-step via an edge-enhanced heterogeneous graph neural network.

4.2.1 *Graph Construction with Multi-space Nodes.*

From the multi-space feature set $\mathcal{Z}$ produced by the feature embedding network, we can construct a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}; \mathcal{T})$ with initial node features $H^{(0)}$ and initial edge features $E^{(0)}$.

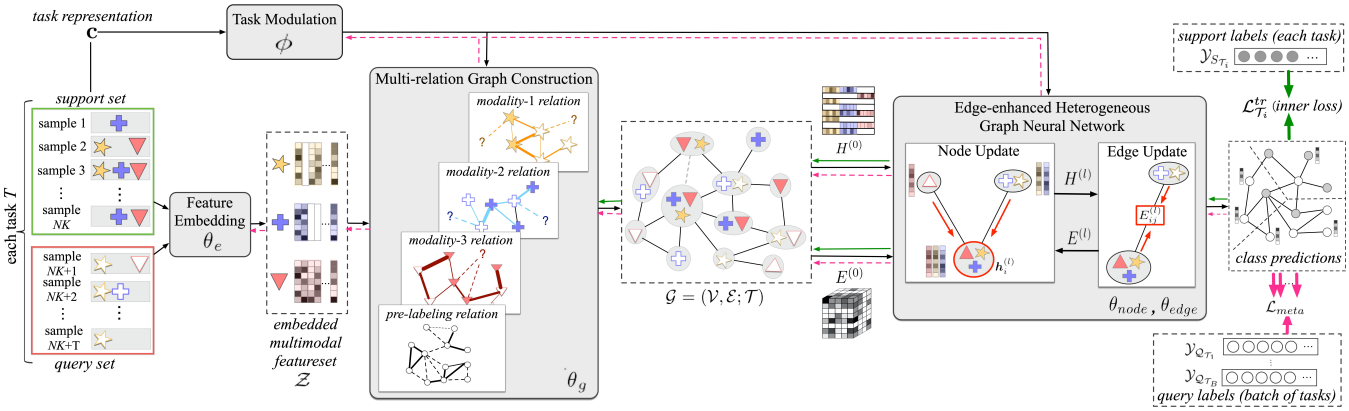

Figure 2: The proposed TopoNet framework.

The initial feature of each node v_i is the concatenation of available modalities of the sample, i.e., $\boldsymbol{h}_i^{(0)} = ||_{m \in \mathcal{B}_i} \boldsymbol{z}_{i,m}$, where $||$ denotes concatenation. The initial node feature set $H^{(0)} = \{\boldsymbol{h}_i^{(0)} \in \mathbb{R}^{|\mathcal{B}_i|F}\}_{i=1}^{NK+T}$ is ***heterogeneous*** as different nodes (samples) have different combinations of modalities. Note that if two nodes $\boldsymbol{h}_i^{(0)}$ and $\boldsymbol{h}_j^{(0)}$ with $\mathcal{B}_i = \{1, 2\}$ and $\mathcal{B}_i = \{2, 3\}$, although both are $2F$-dimensional (i.e., $|\mathcal{B}_i| = |\mathcal{B}_j| = 2$), they still belong to different feature spaces.

Edge features leverage data relationships. However, it is unfeasible to directly measure the similarity between a pair of heterogeneous nodes; also, some pairs of nodes may not contain common modalities, such as node $\mathcal{B}_i = \{1\}$ and node $\mathcal{B}_j = \{2, 3\}$, but belong to the same class and should be connected. Considering these difficulties, we initialize an ***multi-relation graph*** where each edge measures multiple views of node relationships: 1) each modality-m can provide a view of node relations by comparing the mth modality (if available); 2) the given labels of support samples can provide an additional view of class similarities. We obtain an edge-feature tensor $E^{(0)} \in \mathbb{R}^{D \times D \times (M+1)}$, where $D = NK + T$. Each (i, j, m)-entry of $E^{(0)}$ is calculated as

$$E^{(0)}_{i,j,m \le M} = \begin{cases} \sigma(f_g^m(\Delta_{i,j}^m; \theta_g^m)) & \text{if } m \in \mathcal{B}_i \bigcap \mathcal{B}_j \\ 0.5 & \text{if } m \notin \mathcal{B}_i \bigcap \mathcal{B}_j, \end{cases}$$
$$E^{(0)}_{i,j,M+1} = \begin{cases} 1 & \text{if } y_i = y_j \text{ and } v_i, v_j \in \mathcal{S}_\mathcal{T} \\ 0 & \text{if } y_i \neq y_j \text{ and } v_i, v_j \in \mathcal{S}_\mathcal{T} \\ 0.5 & \text{if } v_i \in \mathcal{Q}_\mathcal{T} \text{ or } v_j \in \mathcal{Q}_\mathcal{T}, \end{cases} \quad (5)$$

where $\Delta_{i,j}^m = |z_{i,m} - z_{j,m}|$; $f_g^m(\cdot; \theta_g^m)$ is the metric function for modality-m, a stacked Multilayer Perceptron network with parameter θ_g^m; and σ is sigmoid function. The edge feature (relationship) between a pair of nodes is an $(M+1)$-dimensional vector, constructed by measuring each view's similarity scores. Note that for some pairs of samples without common views but belong to similar classes, they should have high similarity scores in some missing views but the missing views' similarity scores cannot be calculated; we use 0.5 to account for these uncertain views.

4.2.2 ***Transductive Learning with Edge-enhanced Heterogeneous Graph Neural Network.*** An hFSL classification task has been converted into a *node-heterogeneous and multi-relation graph* $\mathcal{G} = (\mathcal{V}, \mathcal{E}; \mathcal{T})$. In the proposed topological transduction framework, solving an hFSL classification task can be formulated as the semi-supervised node classification task on $\mathcal{G}$, supervised by the training nodes $\mathcal{S}_\mathcal{T}$. Yet the difficulty here is the complexity of learning a graph with several types of nodes and multi-view node connections. Therefore, we employ the power of Graph Neural Networks (GNNs) to facilitate transductive learning on $\mathcal{G}$.

Given the initial heterogeneous node features $H^{(0)}$ and multi-relation edge features $E^{(0)}$, the edge and node features are updated iteratively layer by layer through a stacked edge-enhanced heterogeneous graph neural network (EHGNN):

$$H^{(l)} = f_{node}^l(H^{(l-1)}, E^{(l-1)}; \theta_{node}^l)$$
$$E^{(l)} = f_{edge}^l(H^{(l)}, E^{(l-1)}; \theta_{edge}^l), \quad (6)$$

where θ_{node}^l and θ_{edge}^l are the node and edge updating parameters at layer l, respectively. Basically, nodes from multiple spaces are aligned into a unified semantic space along the procedure; edge features are directly encoded in the node updating model so that multi-view similarity scores can be incorporated to improve node representativeness.

A. *Heterogeneous Space Alignment via Node Update*. At the first layer, we are give the initial heterogeneous node features $H^{(0)}$. Each node feature $\boldsymbol{h}_i^{(1)}$ is updated by aggregating its one-hop neighborhoods from each feature space

$$\boldsymbol{h}_i^{(1)} = ||_{r=1}^{M+1} \sigma\left[\left(\sum_{u \in \mathcal{U}_\mathcal{T}} \sum_{j \in \mathcal{N}(i,u)} \widehat{E}_{ijr}^{(0)} \boldsymbol{W}_{r,u}^{(1)} \boldsymbol{h}_j^{(0)}\right)\right], \quad (7)$$

where $||$ is concatenation operation, $\mathcal{U}_\mathcal{T}$ denotes a set of input spaces in $\mathcal{T}$, and $\mathcal{N}(i, u)$ denotes a set of neighboring nodes for v_i on the input-space u. $\boldsymbol{W}^{(1)} = \{\boldsymbol{W}_{r,u}^{(1)} \in \mathbb{R}^{F_1 \times F_u} | r \le M+1, u \in \mathcal{U}_\mathcal{T}\}$ are parameters of node encoders for nodes in each feature space and each view of relationship, where F_u is the dimension of feature space u, and F_1 is the dimension of node encoders' outputs. Edge features are incorporated into the neighborhood aggregation, where each view of the multi-relation edge features generates a new node feature which is then concatenated with other views' new features. To avoid increasing the scale of output features by multiplication, we normalized edge features over the neighborhood

of v_i, that is, $\widehat{E}^{(0)}_{ijr} = \frac{E^{(0)}_{ijr}}{\sum_{k\in\mathcal{N}(i)} E^{(0)}_{ikr}}$. Then, at layers $l > 1$, we simplify the aggregation process for training efficiency as node features are early homogenized in an $F_1(M+1)$-dimensional space. Given features obtained in the last layer $H^{(l-1)} \in \mathbb{R}^{D\times F_{l-1}}$ and $E^{(l-1)} \in \mathbb{R}^{D\times D}$,

$$\boldsymbol{h}_i^{(l)} = \sigma\left[\sum_{j\in\mathcal{N}(i)} \widehat{E}^{(l-1)}_{ij} \boldsymbol{W}^{(l)} \boldsymbol{h}_j^{(l-1)}\right], \tag{8}$$

where $\widehat{E}^{(l-1)}_{ij} = \frac{E^{(l-1)}_{ij}}{\sum_{k\in\mathcal{N}(i)} E^{(l-1)}_{ik}}$, and $\boldsymbol{W}^{(l)} \in \mathbb{R}^{F_l\times F_{l-1}}$ denotes the layer-l node encoder shared by each sample.

B. *Edge Update*. Edge feature update is done by measuring the relationships of current node features. The goal of edge update is to modify the previous representations for inter- and intra- class relationships, making the topological structure more relevant to the specific task. To simplify and reduce the parameter size, the dimensions of edge features after the first layer are reduced to 1. Therefore, at the first edge updating layer, the initial $(M+1)$-view edge features are compressed into a single view

$$E^{(1)}_{ij} = \frac{1}{M+1}\sum_{r=1}^{M+1} \alpha^1_{ij,r} E^{(0)}_{ijr}, \tag{9}$$

where $\alpha^1_{ij,r} = f^l_{edge,r}(\boldsymbol{h}_i^{(1)}, \boldsymbol{h}_j^{(1)}; \theta^1_{edge,r})$ is a scalar that measures the relationship between $\boldsymbol{h}_i^{(l)}$ and $\boldsymbol{h}_j^{(l)}$, which can be calculated using any metric or attention function (e.g., additive attention, dot-product, multiplicative attention) [31]. Then, at layers $l > 1$, to simplify the calculation, edge features are updated directly using the attention scores over current node features,

$$E^{(l)}_{ij} = f^l_{edge}(\boldsymbol{h}_i^{(l)}, \boldsymbol{h}_j^{(l)}; \theta^l_{edge}). \tag{10}$$

To summarize, the information aggregation through edges takes into account the current edge features, thus automatically leveraging the current learned inter- and intra-class relationships and achieving. The information exchange among support and query samples jointly models different types spaces, where each space could incorporate extra information from other spaces. This process implicitly achieves multi-space alignment so that could alleviate the hybrid-shot and data-scarcity challenges.

4.2.3 ***Generalization and Task Adaptiveness of Topology.*** Meta learning explores the transferable knowledge across tasks. In TopoNet, we aim to generalize the underlying topological structure over different hFSL tasks, including the multi-space alignment parameters and the parameters used in modeling intra- and inter-class data relationships. Despite the globally shared structural knowledge, there is also specific knowledge about underlying topological structure for each task. For example, the importance of each modality may vary between different tasks. Therefore, following [22], we build a *task modulation network* $g(\cdot;\phi)$ to condition the topological transductive learning module, which utilizes external task-level information to slightly adjust the prior knowledge for each task, may be better suited for finding correct underlying task-specific class distribution.

Algorithm 1 Training Procedure of TopoNet

Requires: Distribution of hybrid few-shot tasks $P(\mathcal{T})$
Requires: Learning rates α, β; GNN layer number L
Randomly initialize task network θ and meta-network ϕ.
while not done **do**
  Sample batches of tasks $\mathcal{T}_t \sim P(\mathcal{T})$
  for all t **do**
    Obtain data $\{\mathcal{S}_{\mathcal{T}_t}, \mathcal{Q}_{\mathcal{T}_t}\}$ for each task $\mathcal{T}_t$.
    Initialize task network $\theta'_t = \theta_0$, and replace ϕ_0 using $\phi_{0,t}$.
    Calculate embedded multi-space feature set $\mathcal{Z}_t$.
    Construct graph $\mathcal{G}_t$ and initialize $H_t^{(0)}$ and $E_t^{(0)}$.
    Update node and edge features via EHGNN; obtain $H_t^{(L)}$.
    Obtain predictions $\mathcal{Y}_{\mathcal{S}_{\mathcal{T}_t}}$ for support set, compute adapted internal parameters with a fixed number of steps w.r.t. the NK examples from $\mathcal{S}_{\mathcal{T}_t}$ as in Eq.(12).
    Evaluate $\mathcal{L}_t(f(\boldsymbol{x};\theta'_t,\phi), y^*; \mathcal{Q}_{\mathcal{T}_t})$ w.r.t. T samples of $\mathcal{Q}_{\mathcal{T}_t}$.
  end for
  Update initialization of task network θ_0 as Eq.(13).
  Update meta-network ϕ as Eq.(14).
end while
return: θ_0 and ϕ

4.3 Optimization

Task Objective. In our framework, a hybrid N-way K-shot classification task is converted into a semi-supervised N-way K-shot node classification task with heterogeneous nodes. After obtaining node features $H^{(L)} \in \mathbb{R}^{D\times F_L}$ at the last GNN layer L, we use a nonlinear classifier $p(\cdot;\theta_p)$ followed by a softmax layer to make class predictions for each node. The predictions are compared with ground-truth labels to calculate cross-entropy losses. The inner-loop optimization is supervised by the support labels, by minimizing the cross-entropy loss defined as follows:

$$\mathcal{L}_{\mathcal{T}_t} = -\sum_{y_i\in\mathcal{Y}_{\mathcal{S}_{\mathcal{T}_t}}} y_i \cdot \log(\mathrm{softmax}(p(\boldsymbol{h}_i^{(L)};\theta_p))), \tag{11}$$

where $\mathcal{Y}_{\mathcal{S}_{\mathcal{T}_t}}$ denotes the NK labels in the support set $\mathcal{S}_{\mathcal{T}_t}$. Note that the final-layer node representation $\boldsymbol{h}_i^{(L)}$ of v_i has aggregated the data information from both $\mathcal{S}_{\mathcal{T}_t} \setminus \mathcal{Y}_{\mathcal{S}_{\mathcal{T}_t}}$ and $\mathcal{Q}_{\mathcal{T}_t} \setminus \mathcal{Y}_{\mathcal{Q}_{\mathcal{T}_t}}$ through GNNs. With the supervision of the support labels, the topological structure learned by the topological learning network can be relevant to true class distribution of the specific task.

Meta-objective. We train TopoNet following the optimization-based meta-learning paradigm, such as the model-agnostic meta-learning [8], which solves a *bilevel optimization* problem to find a *prior* θ as the meta-learner's parameters. The parameters of our three-module network is $\theta = \{\theta_e, \psi, \theta_p\}$, where $\psi = \{\theta_g, \theta_{node}, \theta_{edge}\}$ is the topological transduction module. The meta-objective is to obtain a set of meta-initialization parameters θ_0, an appropriate generalization of prior knowledge for all tasks, plus the parameters of the external task-modulation meta-network ϕ [22].

Bilevel optimization. Formally, let θ'_t signify θ for task $\mathcal{T}_t$ during the inner-loop optimization, and let the initial $\theta'_t = \theta_0$. In the inner-loop adaption, during each gradient update, we compute

$$\theta'_t \leftarrow \theta'_t - \alpha\nabla_{\theta'_t}\mathcal{L}_{\mathcal{T}_t}(f(\boldsymbol{x};\theta'_t,\phi), y; \mathcal{S}_{\mathcal{T}_t}), \tag{12}$$

where $f(\cdot)$ is the forward function of TopoNet, and $\mathcal{L}_{\mathcal{T}_t}(\cdot; \mathcal{S}_{\mathcal{T}_t})$ is the loss on the support set of $\mathcal{T}_t$ as in Eq.(11).

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:

$$\theta_0 \leftarrow \theta_0 - \beta \nabla_{\theta_0} \sum_{\mathcal{T}_t \sim p(\widehat{\mathcal{T}})} \mathcal{L}_{\mathcal{T}_t}(f(\boldsymbol{x}^*; \theta_i'(\theta_0), \phi), y^*; \mathcal{Q}_{\mathcal{T}_t}) \quad (13)$$

$$\phi \leftarrow \phi - \beta \nabla_{\phi} \sum_{\mathcal{T}_t \sim p(\widehat{\mathcal{T}})} \mathcal{L}_{\mathcal{T}_t}(f(\boldsymbol{x}^*; \theta_i'(\theta_0), \phi), y^*; \mathcal{Q}_{\mathcal{T}_t}), \quad (14)$$

where $\mathcal{L}_{\mathcal{T}_t}(\cdot; \mathcal{Q}_{\mathcal{T}_t})$ is the loss on the query set of task $\mathcal{T}_t$. The overall training procedure of TopoNet is in Algorithm 1.

5 EXPERIMENTS

We evaluate TopoNet on N-way K-shot classification tasks with both uniform and hybrid few-shot settings.

5.1 Dataset

We first evaluated our model under the normal uniform FSL setting, using *five* standard few-shot classification datasets: 1) *three* datasets were used for single-modal (image) few-shot classification, including ***mini*ImageNet** [27] (having 100 classes split as $|C_{train}| = 62$, $|C_{test}| = 30$, and $|C_{val}| = 8$), **omniglot** [17] (having 1623 classes split as $|C_{train}| = 1150$, $|C_{test}| = 423$, and $|C_{val}| = 50$), and **CUB-200** [34] (containing 200 bird species split as $|C_{train}| = 100$, $|C_{test}| = 50$, $|C_{val}| = 50$); 2) *two* datasets were used to simulate the uniform multimodal few-shot scenarios, including the **CUB-200 (image+text)** [34] originated from CUB-200, where each image is annotated with a 312-dimensional text (attribute) modality, and the 3D-object recognition dataset ***mini*Model40 (view1+view2)** constructed from the ModelNet40 [38], which contains 3D CAD objects covering 40 common categories (split as $|C_{train}| = 25$, $|C_{test}| = 9$, and $|C_{val}| = 6$) and each object is marked by two views of feature representations as in [7].

In addition, to simulate the web application scenarios where concepts are scarcely labeled and data is heterogeneous, we constructed *two* hybrid few-shot classification datasets, as hFSL was never studied before and we cannot find existing datasets available. The two datasets are constructed from each of the uniform multimodal datasets: **h-CUB-200** and **h-*mini*Model40**, which contain hybrid combinations of modalities. Specifically, in order to simulate the irregular and frequent occurrence of missing modality in the real-world web applications, each uniform task in the source dataset was turned into the hybrid task by randomly deleting modalities from randomly picked samples. The deletion process is as follows. For each task, we first union the support and query set, and shuffle the instances. Then, we separate the combined set, which contains $(NK + T)$ instances, into $2^M - 1$ disjoint subsets (groups): given the hybrid ratio $0 < \rho < 1$, the first group has $(1 - \rho)(NK + T)$ samples, and the other groups has $\rho(NK + T)/(2^M - 2)$ samples. Each group except the first one is a proper subset of $\{1, ..., M\}$ indicating the modality availability, and for all the samples in the same group, we remove the absent modalities from the original multimodal data. Finally, in the first group, we picked ρ percentage of samples, and from each picked sample, we randomly deleted one of modalities.

Method	***mini*ImageNet**		**CUB200**	**omniglot**
	5-way 1-shot	5-way 5-shot	5-way 1-shot	20-way 1-shot
MAML	49.61	65.72	74.25	95.83
ProtoNet	46.14	65.77	73.99	96.00
RelationNet	51.38	67.07	76.58	97.60
GNN	52.91	68.23	73.76	97.40
TPN	59.46	75.64	75.20	-
TransductiveTuning	62.35	74.53	73.46	-
LaplacianShot	72.11	82.31	80.96	-
TopoNet-U	**72.45**	**83.22**	**81.13**	**99.62**

Table 1: Average accuracy (%) on single-modal few-shot classification datasets.

5.2 Baseline Methods

We compared TopoNet with three families of existing FSL approaches: 1) supervised learning approaches with inductive inference: **ProtoNet** [28], **RelationNet** [29], and **MAML** [11]; 2) semi-supervised learning approaches with transductive inference: **GNN** [11], **TPN** [19], **TransductiveTuning** [6] and **LaplacianShot** [45]; 3) while the previous two families are single-modal baselines, we also consider recent works on multimodal domain: **AM3** [39] and **MultiProtoNet** [23].

When testing the single-modal baselines (e.g., MAML, ProtoNet, RelationNet, LaplacianShot, etc.) under uniform multimodal low-data conditions, we concatenated all the modalities after the feature embedding network, converting multimodal data to single-modal data by linearly combining the multimodal feature embeddings. In addition, when testing the single-modal baseline methods on the hybrid multimodal dataset, we imputed the missing modalities by zeros on the input before concatenating all the original/imputed modalities. Baseline results on single-modal classification datasets are mostly retrieved from their papers.

5.3 Results

We implemented two versions of TopoNet: **TopoNet-U** for uniform tasks and **TopoNet-H** for hybrid tasks. The configurations of our model and baselines, hyperparameters, and experimental setups can be found in Appendix A.2. We fix the number of inner-loop gradient updates to 10 steps in all experiments, and the batch size for updating the meta-learner was fixed to 4 tasks each step.

5.3.1 Uniform Few-shot Classification. We evaluated our model (version TopoNet-U) in the standard single-modal scenarios in Table 1. We can observe that TopoNet-U and existing transductive methods generally outperform inductive methods as unlabeled data was incorporated into task adaptation. The performance of TopoNet-U is comparative to existing transductive methods, so that TopoNet could work for both uniform and hybrid few-shot learning. In Table 2, we evaluated our model under the uniform multimodal scenarios, where all modalities are available all the time. The M modalities are

Method	CUB-200		*mini*Model40	
	5-way 1-shot	5-way 5-shot	5-way 1-shot	10-way 1-shot
MAML	75.78	80.31	82.45	71.46
ProtoNet	67.32	73.74	74.91	62.27
RelationNet	78.22	83.26	85.84	73.62
GNN	72.98	77.75	78.35	68.54
TPN	77.23	81.67	84.28	72.39
TransductiveTuning	75.31	84.28	82.75	72.82
LaplacianShot	81.36	87.76	89.91	78.48
AM3-ProtoNet++	76.60	82.9	83.24	71.71
AM3-TADAM	77.16	82.7	84.10	72.94
MultiProtoNet	77.21	83.29	84.34	73.89
TopoNet-U (Ours)	**81.75**	**88.12**	**91.23**	**79.17**

Table 2: Average accuracy (%) on uniform multimodal few-shot classification datasets.

Method	h-CUB-200		h-*mini*Model40	
	5-way 1-shot	5-way 5-shot	5-way 1-shot	10-way 1-shot
MAML	69.45	74.26	77.83	67.54
ProtoNet	62.44	68.5	69.30	57.10
RelationNet	73.90	78.72	80.45	67.4
GNN	67.41	72.34	73.45	62.58
TPN	71.17	76.38	79.83	66.05
TransductiveTuning	69.73	68.62	76.15	68.10
LaplacianShot	78.06	82.37	84.63	74.43
AM3-ProtoNet++	72.46	76.55	78.68	67.18
AM3-TADAM	73.15	77.28	79.54	68.72
MultiProtoNet	71.34	77.44	79.71	69.44
TopoNet-H (Ours)	**80.23**	**83.11**	**86.46**	**77.15**

Table 3: Average accuracy (%) on hybrid few-shot classification datasets with hybrid ratio $\rho = 0.5$.

concatenated in both baselines and our model. Our model TopoNet-U achieved slightly better performance rather that baselines as we constructed a multi-relation graph where edge features were more complex than baselines, and then learned the data relationships through the graph neural network by incorporating multi-view edge features.

5.3.2 Hybrid Few-shot Classification. Table 3 reports the results on the created hybrid few-shot datasets with $\rho = 0.5$. These results compare our method TopoNet-H, which directly learned with the original heterogeneous data, against the baselines (designed for uniform tasks), which used zeros to impute missing modalities so that hybrid tasks were converted into uniform tasks. From uniform settings (Table 2) to hybrid counterparts, although our models were relevantly influenced by the missing modalities, we can observe that the performance of baselines drops more dramatically than our model. The reason might be that the zero imputation brought some extra noise to the baselines. In contrast, our model, which directly learn from present data from multiple feature spaces, can avoid such noise. This concludes that the imputation strategy is not recommended in few-shot situations, and that TopoNet is an useful tool for hybrid FSL rather than existing uniform algorithms. Also, the effectiveness of our methods demonstrated our heterogeneous neighborhood aggregation can comprehensively utilize other samples' information to alleviate the impact of missing information.

5.3.3 Impact of Hybrid Levels. In Table 4, from column 2 to 4, we increase the hybrid ratio of tasks over the dataset. The larger ρ implies more missing modalities and a larger number of input feature spaces. The last column shows the result with dynamic hybrid ratios, where for each task, the value of ρ was not given but randomly chosen, thus different tasks have different hybrid levels. As the hybrid ratio increases, the less change on TopoNet-H's performance rather than baselines demonstrates the effectiveness of our method to handle multiple spaces.

Method	$\rho = 0.3$	$\rho = 0.5$	$\rho = 0.7$	dynamic ρ
TPN	73.29	71.33	66.23	67.42
AM3-TADAM	75.54	73.15	67.91	66.73
MultiProtoNet	73.67	71.34	64.78	69.72
LaplacianShot	80.31	78.06	72.01	75.13
TopoNet-H‡	71.20	69.18	63.89	69.93
TopoNet-H†	80.16	68.50	69.82	77.34
TopoNet-H	**81.67**	**80.23**	**75.13**	**74.96**

Table 4: Ablation study on hybrid 5-way 1-shot h-CUB-200.

5.3.4 Ablation Study. In Table 4, we evaluate the influence of each component in our model. The TopoNet-H† model replaces the graph construction module with a non-parameter metric kernel (i.e., dot-product similarity) and removes missing-view connections. The TopoNet-H‡ deletes the GNN-based node and edge updating mechanism, and replaces it with the non-parameterised Label Propagation [19] strategy. TopoNet-H outperform TopoNet-H† and TopoNet-H‡. Also, as the hybrid ratio increased, the performance of TopoNet-H† and TopoNet-H‡ dropped more dramatically than TopoNet-H. These proved the ability of heterogeneous GNN in multi-space alignment, and the ability of the topology learning module to generalize reliable inter- and intra-class data relationships across tasks.

6 CONCLUSION

Web data may contain new concepts that are lack of sufficient supervision as well as could be multimodal, heterogeneous, and hybrid, thus may bring challenges to machine learning or deep learning-based web content analysis and web mining that relies on large-scale data. Therefore, in this paper, we studied a novel hybrid few-shot learning (hFSL) problem to employ FSL in such web scenarios. We proposed a task-adaptive topological tansduction network (TopoNet) to solve hFSL, which trained a heterogeneous graph-based transductive meta-learner to handle the special few-shot tasks with multiple input spaces. Our experimental results demonstrated that TopoNet successfully generalized the meta-knowledge about data and multi-space relationships over tasks, and could fast adapt to real tasks with different levels of hybrid settings.

REFERENCES

[1] Malik Boudiaf, Imtiaz Ziko, Jérôme Rony, Jose Dolz, Pablo Piantanida, and Ismail Ben Ayed. 2020. Information Maximization for Few-Shot Learning. *Advances in Neural Information Processing Systems* 33 (2020).

[2] Hsinchun Chen and Michael Chau. 2004. Web mining: Machine learning for web applications. *Annual review of information science and technology* 38, 1 (2004), 289–329.

[3] 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.

[4] Jiayi Chen and Aidong Zhang. 2021. HetMAML: Task-Heterogeneous Model-Agnostic Meta-Learning for Few-Shot Learning Across Modalities. *arXiv preprint arXiv:2105.07889* (2021).

[5] Mingyang Chen, Wen Zhang, Wei Zhang, Qiang Chen, and Huajun Chen. 2019. Meta relational learning for few-shot link prediction in knowledge graphs. *arXiv preprint arXiv:1909.01515* (2019).

[6] Guneet S Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. 2019. A baseline for few-shot image classification. *arXiv preprint arXiv:1909.02729* (2019).

[7] 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.

[8] 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.

[9] Chelsea Finn, Kelvin Xu, and Sergey Levine. 2018. Probabilistic model-agnostic meta-learning. In *Advances in Neural Information Processing Systems*. 9516–9527.

[10] Luca Franceschi, Mathias Niepert, Massimiliano Pontil, and Xiao He. 2019. Learning discrete structures for graph neural networks. In *International conference on machine learning*. PMLR, 1972–1982.

[11] Victor Garcia and Joan Bruna. 2017. Few-shot learning with graph neural networks. *arXiv preprint arXiv:1711.04043* (2017).

[12] Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. 2020. Meta-learning in neural networks: A survey. *arXiv preprint arXiv:2004.05439* (2020).

[13] Ruibing Hou, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. 2019. Cross attention network for few-shot classification. *arXiv preprint arXiv:1910.07677* (2019).

[14] Andrea Isoni. 2016. *Machine learning for the web*. Packt Publishing Ltd.

[15] Jongmin Kim, Taesup Kim, Sungwoong Kim, and Chang D Yoo. 2019. Edge-labeling graph neural network for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11–20.

[16] Shyam Nandan Kumar. 2015. World towards advance web mining: A review. *American Journal of Systems and Software* 3, 2 (2015), 44–61.

[17] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. 2015. Human-level concept learning through probabilistic program induction. *Science* 350, 6266 (2015), 1332–1338.

[18] Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. 2017. Meta-SGD: Learning to Learn Quickly for Few Shot Learning. *CoRR* abs/1707.09835 (2017). arXiv:1707.09835 http://arxiv.org/abs/1707.09835

[19] Lu Liu, Tianyi Zhou, Guodong Long, Jing Jiang, and Chengqi Zhang. 2019. Learning to Propagate for Graph Meta-Learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (Eds.). 1037–1048. https://proceedings.neurips.cc/paper/2019/hash/00ac8ed3b4327bdd4ebbebcb2ba10a00-Abstract.html

[20] Yanbin Liu, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sung Ju Hwang, and Yi Yang. 2019. Learning to Propagate Labels: Transductive Propagation Network for Few-Shot Learning. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net. https://openreview.net/forum?id=SyVuRiC5K7

[21] Alex Nichol, Joshua Achiam, and John Schulman. 2018. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999* (2018).

[22] Boris N Oreshkin, Pau Rodriguez, and Alexandre Lacoste. 2018. Tadam: Task dependent adaptive metric for improved few-shot learning. *arXiv preprint arXiv:1805.10123* (2018).

[23] Frederik Pahde, Mihai Puscas, Tassilo Klein, and Moin Nabi. 2021. Multimodal Prototypical Networks for Few-shot Learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2644–2653.

[24] 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.

[25] Limeng Qiao, Yemin Shi, Jia Li, Yaowei Wang, Tiejun Huang, and Yonghong Tian. 2019. Transductive episodic-wise adaptive metric for few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 3603–3612.

[26] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. 2018. Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676* (2018).

[27] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International journal of computer vision* 115, 3 (2015), 211–252.

[28] Jake Snell, Kevin Swersky, and Richard S Zemel. 2017. Prototypical networks for few-shot learning. *arXiv preprint arXiv:1703.05175* (2017).

[29] 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.

[30] Vladimir N Vapnik. 1999. An overview of statistical learning theory. *IEEE transactions on neural networks* 10, 5 (1999), 988–999.

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

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

[33] 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.

[34] 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.

[35] Fei Wang and Changshui Zhang. 2007. Label propagation through linear neighborhoods. *IEEE Transactions on Knowledge and Data Engineering* 20, 1 (2007), 55–67.

[36] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying graph convolutional networks. In *International conference on machine learning*. PMLR, 6861–6871.

[37] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems* (2020).

[38] 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.

[39] Chen Xing, Negar Rostamzadeh, Boris N Oreshkin, and Pedro O Pinheiro. 2019. Adaptive cross-modal few-shot learning. *arXiv preprint arXiv:1902.07104* (2019).

[40] Wenhan Xiong, Mo Yu, Shiyu Chang, Xiaoxiao Guo, and William Yang Wang. 2018. One-shot relational learning for knowledge graphs. *arXiv preprint arXiv:1808.09040* (2018).

[41] Huaxiu Yao, Xian Wu, Zhiqiang Tao, Yaliang Li, Bolin Ding, Ruirui Li, and Zhenhui Li. 2020. Automated relational meta-learning. *arXiv preprint arXiv:2001.00745* (2020).

[42] Huaxiu Yao, Chuxu Zhang, Ying Wei, Meng Jiang, Suhang Wang, Junzhou Huang, Nitesh Chawla, and Zhenhui Li. 2020. Graph few-shot learning via knowledge transfer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 6656–6663.

[43] 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.

[44] Fan Zhou, Chengtai Cao, Kunpeng Zhang, Goce Trajcevski, Ting Zhong, and Ji Geng. 2019. Meta-GNN: On Few-shot Node Classification in Graph Meta-learning. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, Wenwu Zhu, Dacheng Tao, Xueqi Cheng, Peng Cui, Elke A. Rundensteiner, David Carmel, Qi He, and Jeffrey Xu Yu (Eds.). ACM, 2357–2360. https://doi.org/10.1145/3357384.3358106

[45] Imtiaz Ziko, Jose Dolz, Eric Granger, and Ismail Ben Ayed. 2020. Laplacian regularized few-shot learning. In *International Conference on Machine Learning*. PMLR, 11660–11670.

[46] Daniel Zügner and Stephan Günnemann. 2019. Adversarial attacks on graph neural networks via meta learning. *arXiv preprint arXiv:1902.08412* (2019).