# Metadata-driven Task Relation Discovery for Multi-task Learning

**Zimu Zheng**[1] , **Yuqi Wang**[2] , **Quanyu Dai**[3] , **Huadi Zheng**[3] and **Dan Wang**[3]
[1]The Hong Kong Polytechnic University, Huawei Technologies Co.Ltd.
[2]The Hong Kong Polytechnic University, Fujian Nebula Big Data Application Service Co., Ltd.
[3]The Hong Kong Polytechnic University
zimu.zheng@connect.polyu.hk, wangyuqi@nebulabd.cn, dqyzm100@hotmail.com,
huadi.zheng@connect.polyu.hk, dan.wang@polyu.edu.hk

## Abstract

Task Relation Discovery (TRD), i.e., reveal the relation of tasks, has notable value: it is the key concept underlying Multi-task Learning (MTL) and provides a principled way for identifying redundancies across tasks. However, task relation is usually specifically determined by data scientist resulting in the additional human effort for TRD, while transfer based on brute-force methods or mere training samples may cause negative effects which degrade the learning performance. To avoid negative transfer in an automatic manner, our idea is to leverage commonly available context attributes in nowadays systems, i.e., the metadata. In this paper, we, for the first time, introduce metadata into TRD for MTL and propose a novel Metadata Clustering method, which jointly uses historical samples and additional metadata to automatically exploit the true relatedness. It also avoids the negative transfer by identifying reusable samples between related tasks. Experimental results on five real-world datasets demonstrate that the proposed method is effective for MTL with TRD, and particularly useful in complicated systems with diverse metadata but insufficient data samples.

## 1 Introduction

Task Relation Discovery (TRD), i.e., reveal the relation of tasks, has notable value: it is the key concept underlying Multi-task Learning (MTL) and provides a principled way for identifying redundancies across tasks, e.g., to seamlessly reuse supervision among related tasks or solve many tasks in one system without piling up the complexity [Zhang and Yang, 2017a; Zamir and others, 2018].

To automatically extract the task relation, historical training samples are commonly leveraged and heavily relied on [Liu and others, 2017; Zhang *et al.*, 2016]. Nevertheless, in many real-world applications, training samples capture merely partial relation. For example, when inferring task relation for thermal comfort prediction (i.e., whether one is comfort, cold or hot), if similar tasks are clustered merely using samples of sensing data (e.g., temperature and humidity), it can miss some intrinsic domain information (e.g., climate, building information like whether the building is equipped with air-conditioner or not). In such cases, merely using samples for TRD easily leads to incorrect task relation construction and transferring knowledge between unrelated tasks degrades the MTL performance, which is referred to as the *negative transfer* [Pan and Yang, 2010]. That is especially critical in multi-task transfer learning scenarios where data samples of targeted tasks are usually insufficient due to sensing networking, or privacy issues, i.e., the data size is too small to automatically extract reliable and valuable intrinsic information without integrating additional domain knowledge. However, in real-world applications, TRD with domain knowledge as apriori information requires non-trivial human effort, bringing difficulties to the wide use of such approaches [Evgeniou and others, 2005; Kato *et al.*, 2008; Han and others, 2014].

In this paper, we, for the first time, leverage *metadata* to handle the commonly existing negative-transfer issue in TRD literature. Metadata, i.e., the context attribute in the database of real-world systems [Balaji and others, 2016; Choi and others, 2017], is usually designed by domain experts for daily operations of the system and available before the learning and prediction process. Metadata usually contain intrinsic task information, such as the climate in the above example. Using metadata, we are able to extract task relation with domain knowledge (but at the same time) in an automatic manner and delivers a punch on the automatic TRD.

Metadata helps to avoid the negative transfer and reduce human effort. However, there are mainly three challenges to further release the potentials of metadata in MTL: (1) The first challenge lies in the representation generation. Originally, Metadata is developed by domain experts for the purpose of daily operations of modern systems, instead of designed by data scientists for the targeted prediction task. Due to the different objectives, simply leveraging all raw metadata in the database can lead to a negative transfer. (2) The second challenge lies in the different types and contents of metadata. In TRD, we need to quantify the task similarity with the introduced metadata representation. However, straightforward 1-hot encoding or Gower's coefficient methods on metadata can introduce false information when metadata are incomparable and cooperate for task relation discovery. (3) The final challenge is that directly adding metadata may not fully avoid negative transfer when additional constraints with metadata are not considered and properly integrated into the system.

In this paper, we propose a novel Metadata-driven TRD for MTL, which can automatically and effectively capture task relation through not only historical samples but also additional metadata. The contributions of the paper can be summarized as:

1. We are the first to introduce metadata into TRD. We propose the collection, and selection of metadata, which aims to automatically select useful metadata as task attributes to avoid negative transfer.
2. We leverage the common neighbor characteristic for distance measurement to tackle incomparable metadata. We discuss the interaction between metadata and data samples. We propose a two-phase clustering taking metadata as apriori constraints for regularization.
3. We evaluate the proposed method on five real-world datasets. We also discuss the advantages, limitations and application scenarios at the end of the paper.

We offer our approach as an attractive mechanism for MTL researchers and application developers who are on the lookout for automatic and effective approaches in MTL, especially for complicated systems with diverse metadata but insufficient data samples.

## 2 Related Work

Task Relation Discovery (TRD) is the key issue and the focus of recent works in transfer learning and multi-task learning [Zhang and Yang, 2017a], including mainly two types:

The first is to conduct TRD through apriori information with additional human effort. Task similarity graphs are given and leveraged to make the model parameters of similar tasks close to each other, based on domain knowledge [Evgeniou and others, 2005; Kato *et al.*, 2008; Han and others, 2014]. However, in real-world applications, such apriori information constructed according to domain knowledge may not be easy to obtain, bringing difficulties to the wide use of such approaches. Few Shot Learning (FSL) [Larochelle *et al.*, 2008; Isele *et al.*, 2016] focus on transferring knowledge to tasks with little or no training sample, by merely leveraging task descriptors for TRD in MTL. However, human effort is still needed for data scientists to determine the specific task descriptors and their values. When it comes to TRD, FSL also ignores the interaction between training samples and task attributes, due to its assumption of little or no training sample.

Second, historical samples are used in MTL for automatic TRD which reduces the human effort [Zhang *et al.*, 2016; Liu and others, 2017]. However, these methods merely rely on the training samples of the task for relation discovery, which can result in missing intrinsic information in complex systems, as reported in [de Roux and others, 2018].

Multi-task Learning (MTL) [Caruana, 1997] is developed to improve the performance of targeted tasks using information from the source tasks. In this sense, MTL is related to transfer learning [Pan and Yang, 2010], but the targeted tasks in MTL are learned simultaneously, whereas those tasks in transfer learning are learned independently. A survey is available in [Zhang and Yang, 2017b].

There are mainly two ways to transfer knowledge in MTL. For targeted tasks, 1) Instance Transfer groups and reuses samples of the other tasks, e.g., Clustered MTL (CMTL) [Liu and others, 2017; Xu and others, 2015; Wang *et al.*, 2009], while 2) Feature Representation Transfer groups and learns a common feature representation among the related tasks, e.g., Alternating Structure Optimization [Ando, 2006]. Particularly, our attention has been drawn to the study of instance transfer, i.e., CMTL, which naturally adapt to different feature engineering and learning models. But unlike traditional CMTL, we explicitly model the relations among tasks and extract meta-structure in an automatic manner. It is also possible to maintain a multi-task network.

## 3 Metadata-driven TRD Problem

In this section, we first give the notations and definitions used in the paper, especially the concept and collection of metadata. Then we formally define the problem of metadata-driven TRD for MTL.

Let $x_v^i \in \mathbb{R}$ denote the $i$th feature at index $v$, where $i \in [1, m], m, \in \mathbb{N}$; Let $y_v$ denote the predicted label at index $v$. A sample $s_v$ of task $j$ includes the corresponding feature and the predicted label, i.e., $s_v = (\boldsymbol{X}_v, y_v) \subset \mathbb{R}^m \times \mathbb{R}, v \in \boldsymbol{V}_j$.

**Definition of Task.** A task is an abstraction read from raw data. Traditionally, each task $t$ is associated with a set of training sample $\boldsymbol{l}^t = [l_v], \forall v \in \boldsymbol{V}_{t_i}$, and a learning function $f_t^{\Theta}(\cdot)$ where $\Theta$ denotes the model parameter and the inferred result $\hat{y}_v = f_t^{\Theta}(\boldsymbol{X}_v)$. We additionally associate each task with task attributes $\boldsymbol{a}_t$. Thus, a task in our paper is a three-tuple $t = \{\boldsymbol{a}_t, \boldsymbol{l}^t, f_t^{\Theta}(\cdot)\}$.

### 3.1 Metadata

TRD is well-known as the key issue in MTL, which heavily relies on the task attribute. To extract the attributes, we introduce a new data source of *metadata* which helps to better identify, organize and describe the context of varying tasks.

Metadata is the descriptor of information about one or more aspects of a dataset, i.e., data about data [Wikipedia, Accessed 2 DEC 2018]. It is used to summarize basic information about data which can make tracking and working with specific data easier in a real-world system. Many distinct types of metadata exist, including descriptive metadata, structural metadata, administrative metadata, reference metadata, and statistical metadata [Riley, Accessed 5 DEC 2017].

Table 1 shows the collected metadata of thermal comfort tasks mentioned in the Introduction. It shows the season and building where historical samples of tasks are collected. Each metadata record consists of the identity (e.g., Task 11-1), the information type (e.g., Season) and value (e.g., Summer).

| Task (#Samples) | Season | Building Type | Year | Hour |
|---|---|---|---|---|
| Task 3-1 (377) | Summer / Rainy | Centralized System | 1990 | 11 |
| Task 11-1 (43) | Summer | Centralized System | 1985 | 11 |
| Task 12-4 (74) | Summer | Natural Ventilation | 1985 | 11 |
| Task 19-1 (470) | Winter | Natural Ventilation | 1993 | 12 |
| Task 22-3 (1026) | Winter | Natural Ventilation | 1993 | 12 |

Table 1: Collected Metadata Examples of Thermal Comfort.

Note that metadata is designed to manage the dataset for various daily operations in modern systems. It is thus common that metadata is already collected and directly available in the database for a running system, serving as a hyper table used to manage other tables of samples. That is especially true for a public database with various datasets, e.g., an example of such a hyper table for the thermal comfort database of ARP-884 is in [The University of Sydney, 2018]. Thus, the first way to collect the metadata is to find and visit the hyper table for the database.

Another way is to collect metadata from those attributes in the samples. Table 2 shows the metadata in samples of thermal comfort tasks. It shows the season, duration of hours and the year when historical samples of tasks are collected. We see that the value of metadata for a specific task/ context is consistent regardless of time changes, e.g., a certain thermal comfort dataset of a task has a fixed value of the year of data collection. Accordingly, we observe a common characteristic that the metadata usually has duplicated values in a table of samples and merely a few values across all tables of samples. This characteristic can be used to detect and collect metadata, of which those samples are commonly regarded as informationless and even discarded in practice. For interested readers, more metadata extraction techniques can be found in [Liu and others, 2007].

| Task | ID | Season | Hour | Year |
|---|---|---|---|---|
| Task 3-1 | 600721 | Summer / Rainy | 11 | 1990 |
| Task 3-1 | 600722 | Summer / Rainy | 11 | 1990 |
| Task 3-1 | 600723 | Summer / Rainy | 11 | 1990 |
| Task 19-1 | 620144 | Winter | 12 | 1993 |
| Task 19-1 | 620145 | Winter | 12 | 1993 |

Table 2: Examples of Metadata in Samples of Thermal Comfort.

### 3.2 Problem Definition

The output of our TRD problem is the *Metadata-driven Task Mapping*. It is a computationally found hypergraph with an emphasis on metadata of tasks. The metadata is used to infer the edge in the hypergraph. An edge between a group of source tasks and a target task represents a feasible transfer case and its weight is the prediction of its performance. We use these edges to estimate the globally optimal transfer policy to solve tasks. The final metadata-driven task relation produces a family of such graphs, parameterized by the chosen tasks, transfer orders, and transfer functions' expressiveness.

Formally, the task mapping is the match of transfer-able source and target tasks, which is defined as $\mathbf{\Theta} = \{T : S\}$ where $T$ is the set of tasks which we want to solve (target), and $S$ is the set of tasks that can be trained (source).

Our problem of Metadata-driven TRD for MTL is inferring a task mapping $\mathbf{\Theta}$ which maximizes the collective performance on a set $T$ of tasks $t \in T$, given training samples and metadata of tasks.

It is critical to note the task mapping is meant to be a sampled set, not an exhaustive list, from a denser space of all conceivable tasks. This gives us a tractable way to sparsely model a dense space, and the hypothesis is that (subject to a proper sampling) the derived model should generalize to out-of-mapping tasks. The more-regular-sampled the space, the better the generalization. To this end, our task mapping with metadata is, in fact, a regularization on the space which facilitates a more general transfer process. We show this in the Evaluation Section.

## 4 Methodology

The task mapping is built using a three-step process. In stage I, a metadata graph for each task in S is generated and effective metadata is selected for each task. In stage II, the task affinities acquired using metadata are computed and normalized, and in stage III, we synthesize a hypergraph by clustering which can gather similar tasks, use multiple inputs task to transfer to one target, and avoid negative transfer with both metadata and training samples.

### 4.1 Stage I: Task-specific Sampling

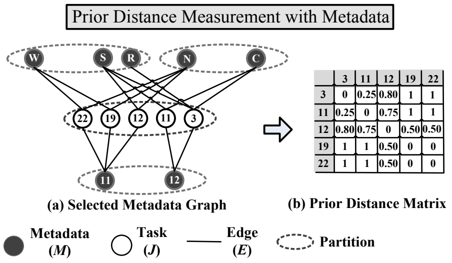
As the base of the task mapping, we establish a knowledge graph using metadata to organize tasks and task attributes, named *metadata graph*. Based on the existing work [Balaji and others, 2016] and [Xie *et al.*, 2016], we denote the metadata as a triple $\{$task, information type, information value$\}$. For example, "Task 19-1 is conducted in the Season of Winter" can be denoted as $\{$Task 19-1, Season, Winter$\}$. Formally, let the nodes $M, T \in U$ in the graph, i.e., $G(U, E)$, denote the metadata and task, respectively, where $U = M \cup T$; the edge $E$ in the graph represents the information type between nodes. Note that the edge weight $w \in [0, 1]$ on the metadata graph refers to the relatedness between the task and metadata. E.g., it is 1 if nodes are related and otherwise 0, which can be determined using the performance improvement in the task attribute selection.

Raw metadata is not necessarily related to the targeted prediction task since it is designed by domain experts to serve various daily operations instead of our targeted prediction tasks. Both the objectives and approaches are significantly different when designing task attributes and metadata. We select metadata as a task attribute if it brings improvement of collective performance for tasks.

### 4.2 Stage II: Affinity Measurement with Metadata

We want an affinity matrix of transfer abilities across tasks based on the metadata graph. We focus on discussing the distance measurement of metadata and omit that of samples in existing works due to page limitations. Between nodes of numerical and ordinal metadata, the distance can be computed using cosine similarity, with normalization to tackle the value in vastly different scales. An analytic hierarchy can be used to facilitate more effective normalization.

When it comes to nodes of categorical metadata, there are quantifying approaches including the straightforward 1-hot or 1-of-K encoding, the more-widely-used Gower's similarity coefficient [Gower, 1971] and its extensions [Legendre and Legendre, 2012; Podani, 1999]. However, these approaches basically treat categorical metadata as constant ordinal or numerical ones, which encodes a hidden ranking/ order relation across the metadata and brings noise to the final distance function. They also simply assume that different categories are independent and unrelated. Thus the above metadata processing methods are not suitable for MTL.



|  | 3 | 11 | 12 | 19 | 22 |
|---|---|---|---|---|---|
| 3 | 0 | 0.25 | 0.80 | 1 | 1 |
| 11 | 0.25 | 0 | 0.75 | 1 | 1 |
| 12 | 0.80 | 0.75 | 0 | 0.50 | 0.50 |
| 19 | 1 | 1 | 0.50 | 0 | 0 |
| 22 | 1 | 1 | 0.50 | 0 | 0 |

(a) Selected Metadata Graph

(b) Prior Distance Matrix

Figure 1: Example of Prior Distance Measurement. It infers the distance matrix between tasks from the metadata graph.

To capture the cooperation and avoid encoding a additional ranking relation, our idea is to leverage the number of common neighbors for distance measurement. The hypothesis is that a task should be more similar to another task, if it has higher numbers of shared metadata with another task. We compute Jaccard Distance [Jaccard, 1908], to compare the number of common categorical neighbors for task nodes $a, b$ and jointly consider different metadata $\boldsymbol{N}_a \cup \boldsymbol{N}_b$: $d_{inc}(a, b) = 1 - |\boldsymbol{N}_a \cap \boldsymbol{N}_b| / |\boldsymbol{N}_a \cup \boldsymbol{N}_b|$, where $\boldsymbol{N}_a, \boldsymbol{N}_b$ are categorical metadata and neighbors of $a, b$.

For all possible pair task nodes (a, b), the output is named *prior distance matrix*, which is computed using metadata, i.e., $\boldsymbol{d}_{meta} = [d_{inc}(a, b)]$. An example of the prior distance matrix is shown in Figure 1.

### 4.3 Stage III: Metadata-assisted Task Mapping

The final step is to infer task mapping based on the affinity matrix inferred from weighted metadata and training samples, e.g., $\boldsymbol{d} = w_0 \boldsymbol{d}_{meta} + w_1 \boldsymbol{d}_{samp}$, where $\boldsymbol{d}_{samp}$ denotes the distance matrix computed with samples. However, when we introduce metadata, there is also a risk of introducing more negative transfers. For example, tasks with quite different values of metadata, e.g., thermal comfort tasks with completely different types of buildings and seasons, is likely to be inreasonable to transfer their samples to each other.

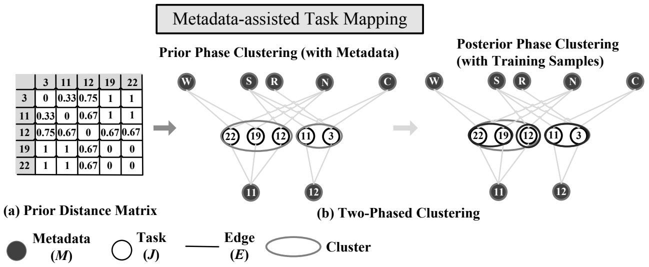
To avoid negative sample transfer between completely different tasks, we propose a two-phased clustering, i.e., prior clustering with metadata and the posterior clustering with samples. The prior clustering defines clusters using the metadata which limitting the variance of task attributes within it. The posterior clustering conducts further clustering with historical training samples based on the prior clusters, i.e.,

$$\boldsymbol{\Theta} = c_1(\ \boldsymbol{s}, \boldsymbol{d}_{samp} \mid c_0(\boldsymbol{d}_{meta})\ ),$$

where $\boldsymbol{\Theta}$ denotes the allocation of sample index for $n_c$ clusters, i.e., the matrix of $\boldsymbol{V}_j^k = \{v | v \in \text{cluster } k \text{ for task} j\}, k \in [0, n_c], j \in [1, J]$; $c_0(\cdot)$ and $c_1(\cdot)$ denote the prior and posterior clustering function. In this rest of this section, we focus on discussing the metadata clustering and omit the traditional clustering with merely samples due to page limitations.

As for the prior clustering, we propose to use Normalized Minimum Cut [Shi and Malik, 2000] for clustering given the distance matrix. The Normalized Minimum Cut can naturally adopt the case when the metadata graph is unconnected.

Consistent with our Thermal Comfort Dataset in previous sections, we also provide examples of prior-and-posterior phase clustering in Figure 2.


Metadata-assisted Task Mapping
Prior Phase Clustering (with Metadata)
Posterior Phase Clustering (with Training Samples)
W
S
R
N
C
22
19
12
11
3
11
12
Metadata (M)
Task (J)
Edge (E)
Cluster

|  | 3 | 11 | 12 | 19 | 22 |
|---|---|---|---|---|---|
| 3 | 0 | 0.33 | 0.75 | 1 | 1 |
| 11 | 0.33 | 0 | 0.67 | 1 | 1 |
| 12 | 0.75 | 0.67 | 0 | 0.67 | 0.67 |
| 19 | 1 | 1 | 0.67 | 0 | 0 |
| 22 | 1 | 1 | 0.67 | 0 | 0 |

(a) Prior Distance Matrix

(b) Two-Phased Clustering

Figure 2: Example of Metadata-assisted Task Mapping. In this step, it infers the task mapping by clustering with the distance matrix.

## 5 Evaluation

### 5.1 Experimental Setup

For the parameters, we apply grid searching [Gu and Zhou, 2009] to identify the optimal values. For all clustering algorithms, the number of nearest neighbors is set by searching the grid $\{\lceil \frac{n_s}{2 \times n_c} \rceil, \lceil \frac{n_s}{n_c} \rceil, \min(\lceil \frac{2 \times n_s}{n_c} \rceil, n_s)\}$, where $n_s$ and $n_c$ are the number of training samples and clusters. The number of clusters is set as the number of classes in each dataset. For SAMTL and our posterior-phase clustering, cosine similarity is used to compute nearest neighbors as [Zhang *et al.*, 2016].

To capture the predictive capabilities of our approach, we use three metrics: Error Rate (ER), Root-mean-square error (RMSE) and Symmetric Mean Absolute Percent Error (SMAPE) for evaluation.

$$ER = \frac{1}{J} \sum_{j=1}^{J} \frac{\sum_{n=1}^{N} |\hat{y}_n^j - y_n^j|}{\sum_{n=1}^{N} y_n^j};$$

$$SMAPE = \frac{2}{J \cdot N} \sum_{j=1}^{J} \sum_{n=1}^{N} \frac{|\hat{y}_n^j - y_n^j|}{\hat{y}_n^j + y_n^j + 1};$$

$$RMSE = \frac{1}{J} \sum_{j=1}^{J} \sqrt{\frac{\sum_{n=1}^{N} (\hat{y}_n^j - y_n^j)^2}{N}}$$

where $N$ is the number of testing data; $\hat{y}_n$ and $y_n$ are the estimation and the ground truth of the $n^{\text{th}}$ sample, respectively.

### 5.2 Datasets

The dataset is shown in Table 3. We use the following real-world datasets for our experiments on the baseline methods and the proposed method.

Hong Kong Industry Chiller Data (HK-BCOD) is used to predict the efficiency of different chillers, which are machines to generate cooling power for office buildings. The higher accuracy means more energy saving for cooling load allocation. For three office towers located in Hong Kong, chiller sensor data were collected from the building management system for four years at daily intervals.

ASHRAE RP-884 Thermal Comfort Data (ARP-884) is used to predict occupants' feeling of comfort, cold or hot. The data we use in this study is from a Public dataset of ASHRAE, which is a global professional association seeking

| Dataset | Task | Profile (Relation: Metadata) | | | #Samples |
|---|---|---|---|---|---|
| HK-BCOD | Chiller 1 | ModelType: CDHG2250 | Location: Pacific Place II | Function: Regular | 1460 |
| | Chiller 2 | ModelType: CDHG2250 | Location: Pacific Place II | Function: Regular | 1460 |
| | Chiller 3 | ModelType: CDHG2250 | Location: Pacific Place II | Function: Regular | 1460 |
| | Chiller 4 | ModelType: CVHG780 | Location: Pacific Place II | Function: Backup | 1460 |
| | Chiller 5 | ModelType: CVHG780 | Location: Pacific Place II | Function: Backup | 1460 |
| ARP884 | Task 3-1 | Season: Summer / Rainy | BuildingType: Centralized System | Hour: 11 | 377 |
| | Task 11-1 | Season: Summer | BuildingType: Centralized System | Hour: 11 | 43 |
| | Task 12-4 | Season: Summer | BuildingType: Natural Ventilation | Hour: 11 | 74 |
| | Task 19-1 | Season: Winter | BuildingType: Natural Ventilation | Hour: 12 | 470 |
| | Task 22-3 | Season: Winter | BuildingType: Natural Ventilation | Hour: 12 | 1026 |
| CHM-BST | Huamudui Road | Type: Road | Function: Street | | 665 |
| | Xiangdongnan Cross | Type: Road | Function: Street | | 665 |
| | Jinianguan Road | Type: Road | Function: Street | | 665 |
| | Xihe Bridge | Type: Road | Function: Bridge | | 665 |
| | Jinqiao Investment | Type: Building | Function: Business | | 665 |
| IBM-HWD | Tai Wai | Nation: China | City:Hong Kong | | 70080 |
| | Tsing Yi | Nation: China | City: Hong Kong | | 70080 |
| | Kowloon | Nation: China | City: Hong Kong | | 70080 |
| | Hong Kong Island | Nation: China | City: Hong Kong | | 70080 |
| HK-TSD | Road 3470-3006 | Nation: China | City: Hong Kong | | 28800 |
| | Road 4652-4633 | Nation: China | City: Hong Kong | | 28800 |
| | Road 46332-46522 | Nation: China | City: Hong Kong | | 28800 |

Table 3: Datasets.

to advance heating, ventilation, air conditioning, and refrigeration systems design and construction.

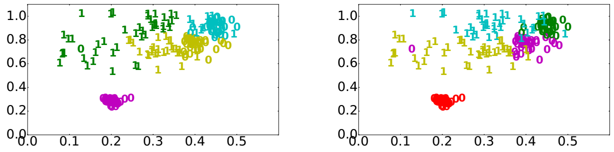
China Mobile Base Station Trace (CHM-BST) is collected to predict the number of mobile devices in Tianjin, China. It contains mobile traces collected from five base stations of *China Mobile*, a dominant carrier of the local mobile network, in Tianjin, China from 15th to 30th, August 2016.

IBM HK Weather Data (IBM-HWD) is used to predict the temperature in different locations of Hong Kong, China. We collected four-year meteorology data from the Public website of *Weather Underground* of IBM.

Hong Kong Traffic Sensing Data (HK-TSD) is used to predict the traffic speed in Hong Kong, China. We collected the four-month data in six-minute intervals from the Public government website of *data.gov.hk*.

## 5.3 Results on Clustering

To indicate the effect of metadata, Figure 3 compares the clustering result of Metadata Clustering (on both samples and metadata) with Sample Clustering (on mere training samples) in HK-BCOD, where tasks are conducted to predict the efficiency of chillers. We mark those chillers of two *modelTypes*, which are supposed to have different models, as 0 and 1. The *modelType* information is recorded in the metadata. In Sample Clustering, data samples of different *modelTypes* are mixed in the same cluster, which can incur negative transfer; whereas in Metadata Clustering, all samples in each cluster will be in the same *modelType*. That is because Metadata Clustering not only leverages additional metadata, but also treats the metadata as clustering constraints by using the Two-phase Clustering; whereas Sample Clustering heavily relies on the historical training samples for clustering, which may result in the missing of some intrinsic information. Other datasets also reveal similar results and we omit the results due to page limitations.



(a)

(b)

Figure 3: For HK-BCOD, the clustering result for two types of samples (in numbers of 0 and 1) in the clusters (in different colors) of: (a) Sample Clustering; (b) Metadata Clustering.

| **Method** | HK-BCOD | | | ARP-884 | | | CHM-BST | | |
|---|---|---|---|---|---|---|---|---|---|
| | ER | SMAPE | RMSE | ER | SMAPE | RMSE | ER | SMAPE | RMSE |
| STL | 1.13 | 2.27 | 1.07 | 0.82 | 1.00 | 3.16 | 0.18 | 0.36 | 0.42 |
| IMTL | 0.60 | 1.19 | 0.77 | 0.77 | 1.13 | 3.06 | 0.20 | 0.41 | 0.45 |
| SAMTL | 0.31 | 0.62 | 0.56 | 0.41 | 0.69 | 1.57 | 0.26 | 0.52 | 0.51 |
| mCMTL | **0.04** | **0.08** | **0.20** | **0.27** | **0.24** | **1.03** | **0.10** | **0.22** | **0.32** |

Table 4: For datasets with different metadata: the performance of different methods for HK-BCOD, ARP-884, and CHM-BST.

## 5.4 Results on MTL

With the clustered samples for training tasks, we determine the learning model to maximize the accuracy of our given tasks. Let $\boldsymbol{y}_j = [y_v], \boldsymbol{X}_j = [x_v^i], v \in \boldsymbol{V}_j$ denote label and feature vector of task $j$. Then, formally, we are to solve the following commonly-accepted MTL optimization problem [Caruana, 1997; Baxter and others, 2000; Evgeniou and Pontil, 2004]. Given a loss function $l(\cdot,\cdot)$, clustering parameter $\boldsymbol{\Theta}$, historical data of $\boldsymbol{X}$ and $\boldsymbol{y}$, we are to infer $\boldsymbol{W} = \arg\min \sum_{j \in J} 1/|\boldsymbol{V}_j| \sum_{v \in \boldsymbol{V}_j} l(f_j^{\boldsymbol{\Theta}}(\boldsymbol{X}_v, \boldsymbol{w}_v), y_v) + P(\boldsymbol{W}, \boldsymbol{\Theta})$, where $P(\boldsymbol{W}, \boldsymbol{\Theta})$ is the regularization term on parameters; Note that $\boldsymbol{\Theta}$, $\boldsymbol{X}$ and $\boldsymbol{y}$ are used to train parameters $\boldsymbol{W}$, by merely using samples of tasks in the same cluster in the training process.

To evaluate the proposed method, we employ the following state-of-the-art methods as baselines. We compare our

| **Method** | IBM-HWD | | | HK-TD | | |
|---|---|---|---|---|---|---|
| | ER | SMAPE | RMSE | ER | SMAPE | RMSE |
| STL | 0.09 | 0.08 | 1.91 | 0.11 | 0.12 | 6.67 |
| IMTL | 0.03 | 0.02 | 0.54 | 0.02 | 0.01 | 0.81 |
| SAMTL | 0.04 | 0.05 | 1.18 | 0.05 | 0.04 | 2.56 |
| mCMTL | **0.04** | **0.04** | **1.07** | **0.05** | **0.05** | **3.27** |

Table 5: For datasets with the same metadata: the performance of different methods in IBM-HWD and HK-TD.

metadata-clustered MTL (mCMTL) with those without using metadata: (1) Single Task Learning (STL), which learns a single model by pooling together data from all tasks of each dataset; (2) Independent Multi-task Learning (IMTL), which learns each task independently without sharing any sample or knowledge; (3) the Self-Adapted Multi-task Learning (SAMTL), which learns the cluster of tasks merely with historical training samples as in [Zhang *et al.*, 2016].

For all tasks, we use Support Vector Regression (SVR) model for the baseline prediction of all tasks since it is simple and allows to see the impact of different MTL methods. We chronologically order each dataset and use the first 1/2 for training and the remaining 1/4, 1/4 for evaluation and testing.

We use datasets to construct two typical cases in MTL: 1) The first case is that the metadata is not completely the same for all tasks. We use HK-BCOD, ARP-884, and CHM-BST to represent this case. 2) The second case is that the metadata are completely the same for all tasks. We use IBM-HWD, and HK-TSD to represent this case.

All our experiments are conducted in a private cloud with 16 cores of 2.6GHz CPU and 64G memory. The training time of all tasks is of seconds and the prediction time is even much less than one millisecond. Table 4 and Table 5 summarize the overall results of all the compared methods with respect to the three evaluation metrics. We have the following observations.

**The Impact of Metadata.** mCMTL generally outperforms SAMTL. This is because mCMTL considers metadata information for transfer which also captures different intrinsic characteristics of tasks, whereas SAMTL captures the related tasks merely by using the training samples, e.g., sensed from the outside environment.

**The Impact of Multi-task Setting.** mCMTL always performs better than Single Task Learning (STL) method, since mCMTL establishes different models for different tasks, whereas the single task learning assumes that tasks are of the same model and does not well capture the intrinsic characteristic and dynamic environmental difference among the tasks.

**The Impact of Transfer.** mCMTL always performs better than IMTL in HK-BCOD, ARP-884, and CHM-BST which contain less training data samples. This is because mCMTL exploits the information across the related tasks, whereas the IMTL only utilizes the information within each task.

**The Scope of mCMTL.** IMTL can outperform mCMTL and SAMTL in IBM-HWD and HK-TD. When training samples are sufficient, transfer between tasks may not be necessary and independent methods like IMTL can also work well. In addition, since tasks have the same metadata, mCMTL and SAMTL still keep transferring knowledge between different tasks which can incur a slight negative effect. mCMTL is more suitable for cases where samples are insufficient and the metadata is not completely the same.

## 6 Discussion

**Metadata Management.** The proposed method uses metadata available from many modern systems. It does not require any major human effort because we focus on metadata from one system each time, where the semantics of metadata remain consistent. We leave metadata management from varying systems as future work. Interested readers can refer to specific metadata database, e.g., Project *Brick* [BETS Research Group, Accessed 1 DEC 2017], *CANSIM* [Dunstan and Humphrey, 2005] and *METeOR* [Australian Institute of Health and Welfare, 2018], and metadata standards from ISO and ANSI, e.g., ISO/IEC 11179 [ISO/ IEC JTC 1, 2018].

**Semantics and Semantics Similarities.** Semantics was developed to find the optimal context label and representation. Applications of semantics have benefitted the constant context adaptation in the domains such as semantic web and knowledge graph, as correctly pointed out by the reviewer. Our application differs from semantic web, etc, because, at least for the time being, metadata usually do not have strong semantics, and there lacks other information, e.g., an expert dictionary, to improve semantic analysis. We have tried to use multi-source semantic analysis for our metadata. We do observe that results can be slightly improved. Nevertheless, the improvement is light also because each task has a number of metadata, and the similarity of two tasks is more influenced by the number of similar metadata. We believe that we can find some applications where the semantics of metadata can have a stronger impact.

## 7 Conclusion

In this paper, we introduced metadata into TRD for MTL and proposed Metadata Clustered Multi-Task Learning (mCMTL). It leverages not only training samples, but also additional metadata, to automatically exploit the positive relatedness between tasks, which reduces human effort for data scientists in preparing the prior task information and avoids the negative transfer by identifying the reusable training samples between such related tasks. Experiments on many real-world datasets demonstrate the superiority of the proposed algorithm over existing MTL methods. In general, this study helps in automatic relation discovery among partially related tasks and sheds new light on the development of TRD in MTL through the use of metadata as apriori information.

## Acknowledgements

This work was supported in part by CRF C5026-18G, in part by the Hong Kong ITF UIM/363, in part by Technical Innovation Department, Cloud BU, Huawei Technologies Co.Ltd. and in part by Fujian Nebula Big Data Application Service Co., Ltd. All opinions, findings, conclusions and recommendations in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

## References

[Ando, 2006] Rie Kubota Ando. Applying alternating structure optimization to word sense disambiguation. In *CoNLL*, pages 77–84, 2006.

[Australian Institute of Health and Welfare, 2018] Australian Institute of Health and Welfare. Project meteor: Metadata repository. https://meteor.aihw.gov.au/content/index.phtml/itemId/181162, 2018.

[Balaji and others, 2016] Bharathan Balaji et al. Brick: Towards a unified metadata schema for buildings. In *Proceedings of the 3rd ACM BuildSys*, pages 41–50, 2016.

[Baxter and others, 2000] Jonathan Baxter et al. A model of inductive bias learning. *J. Artif. Intell. Res.(JAIR)*, 2000.

[BETS Research Group, Accessed 1 DEC 2017] BETS Research Group. A uniform metadata schema for buildings. https://brickschema.org/, Accessed: 1 DEC 2017.

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

[Choi and others, 2017] Edward Choi et al. Gram: Graph-based attention model for healthcare representation learning. In *ACM SIGKDD*, pages 787–795, 2017.

[de Roux and others, 2018] Daniel de Roux et al. Tax fraud detection for under-reporting declarations using an unsupervised machine learning approach. In *ACM SIGKDD*, pages 215–222, 2018.

[Dunstan and Humphrey, 2005] Tim Dunstan and Charles Humphrey. Discovering microdata variables. In *Statistics Canada*, pages No. 11–522–XIE, 2005.

[Evgeniou and others, 2005] Theodoros Evgeniou et al. Learning multiple tasks with kernel methods. *JMLR*, 6(Apr):615–637, 2005.

[Evgeniou and Pontil, 2004] Theodoros Evgeniou and Massimiliano Pontil. Regularized multi–task learning. In *ACM SIGKDD*, pages 109–117, 2004.

[Gower, 1971] John C Gower. A general coefficient of similarity and some of its properties. *Biometrics*, pages 857–871, 1971.

[Gu and Zhou, 2009] Quanquan Gu and Jie Zhou. Learning the shared subspace for multi-task clustering and transductive transfer classification. In *IEEE ICDM*, 2009.

[Han and others, 2014] Lei Han et al. Encoding tree sparsity in multi-task learning: A probabilistic framework. In *AAAI*, pages 1854–1860, 2014.

[Isele *et al.*, 2016] David Isele, Mohammad Rostami, and Eric Eaton. Using task features for zero-shot knowledge transfer in lifelong learning. In *IJCAI*, 2016.

[ISO/ IEC JTC 1, 2018] ISO/ IEC JTC 1. Iso/iec 11179, information technology – metadata registries. http://metadata-standards.org/11179/, 2018.

[Jaccard, 1908] Paul Jaccard. Nouvelles researches sur la distribution florale. *Bull Soc Vaud Sci Nat*, 44, 1908.

[Kato *et al.*, 2008] Tsuyoshi Kato, Hisashi Kashima, Masashi Sugiyama, and Kiyoshi Asai. Multi-task learning via conic programming. In *NIPS*, pages 737–744, 2008.

[Larochelle *et al.*, 2008] Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. Zero-data learning of new tasks. In *AAAI*, volume 1, page 3, 2008.

[Legendre and Legendre, 2012] Pierre Legendre and Loic FJ Legendre. *Numerical ecology*, volume 24. Elsevier, 2012.

[Liu and others, 2007] Ying Liu et al. Tableseer: automatic table metadata extraction and searching in digital libraries. In *ACM/IEEE JCDL*, pages 91–100, 2007.

[Liu and others, 2017] An-An Liu et al. Hierarchical clustering multi-task learning for joint human action grouping and recognition. *IEEE TPAMI*, 39(1):102–114, 2017.

[Pan and Yang, 2010] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.

[Podani, 1999] János Podani. Extending gower's general coefficient of similarity to ordinal characters. *Taxon*, pages 331–340, 1999.

[Riley, Accessed 5 DEC 2017] Jenn Riley. Understanding metadata. http://www.niso.org/publications/press/UnderstandingMetadata.pdf, Accessed: 5 DEC 2017.

[Shi and Malik, 2000] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE TPAMI*, 22(8):888–905, 2000.

[The University of Sydney, 2018] The University of Sydney. Ashrae rp-884 adaptive model project. https://sydney.edu.au/architecture/staff/homepage/richard_de_dear/ashrae_rp-884.shtml, 2018.

[Wang *et al.*, 2009] F. Wang, X. Wang, and T. Li. Semi-supervised multi-task learning with task regularizations. In *IEEE ICDM*, pages 562–568, 2009.

[Wikipedia, Accessed 2 DEC 2018] Wikipedia. Metadata. https://en.wikipedia.org/wiki/Metadata, Accessed 2 DEC 2018.

[Xie *et al.*, 2016] Ruobing Xie, Zhiyuan Liu, and Maosong Sun. Representation learning of knowledge graphs with hierarchical types. In *IJCAI*, pages 2965–2971, 2016.

[Xu and others, 2015] Jianpeng Xu et al. FORMULA: factorized multi-task learning for task discovery in personalized medical models. In *SIAM*, pages 496–504, 2015.

[Zamir and others, 2018] Amir R Zamir et al. Taskonomy: Disentangling task transfer learning. In *IEEE CVPR*, pages 3712–3722, 2018.

[Zhang and Yang, 2017a] Yu Zhang and Qiang Yang. Learning sparse task relations in multi-task learning. In *AAAI*, pages 2914–2920, 2017.

[Zhang and Yang, 2017b] Yu Zhang and Qiang Yang. A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*, 2017.

[Zhang *et al.*, 2016] Xianchao Zhang, Xiaotong Zhang, and Han Liu. Self-adapted multi-task clustering. In *IJCAI*, pages 2357–2363, 2016.