Towards Enabling Learnware to Handle Heterogeneous Feature Spaces

Peng Tan, Zhi-Hao Tan, Yuan Jiang, Zhi-Hua Zhou

Learnware Paradigm

Learnware paradigm [Zhou, 2016] aims to change the paradigm of traditional machine learning where a well-performed model is trained starting from scratch, which is hard for most ordinary people.

Motivation

Current effort
- [Wu et al., 2021] first realized a homogeneous prototype learnware market via reduced kernel mean embedding (RKME) specification.
- sketch the entire dataset via weighted samples in the RKHS
- capture major distribution information with minor samples.

Complicated real-world scenarios
- The feature spaces of existed models and future tasks are usually different.
- However, current effort is only suitable when all models and future tasks share the same feature space.

Key challenges
- How to identify useful models for the current user task when models are from different feature space and the raw user data can't be leaked to the market?
- How does the user reuse helpful models with heterogeneous feature spaces?

Experiments

Real-world tasks
- Our method outperforms 4 contenders in 22 out of 23 cases.

Our Methods

Overview
- The more powerful specification which can manage models from different feature space.
- The procedure of the construction and usage of the heterogeneous learnware market.

Procedure

Establishing stage (the market assign specifications to heterogeneous models)
- (Subspace learning) the market maps the original data of models to a subspace and generate the projection tool for future tasks.
- (Specification assignment) the market generates RKME specification on the mapped data.

Deploying stage (the market help user search and reuse heterogeneous models)
- (User data mapping) the user generates the task requirement.
- map the task data to the subspace using the projection tool provided by the market.
- generate RKME on the mapped data as the task requirement.
- (Learnware recommendation) the market searches for useful learnwares.
- estimate the relevance of each model given the new task based on the specifications and the task requirement, then recommend learnwares with high relevance.
- (Model reuse) the user reuses models via dynamic classifier selection.

Key components

Subspace learning
- Input: \(X^i\) the concentration of all task data \(X^i\) in the \(i\)-th feature space \(X\) and part of the auxiliary data in \(X\).
- Output: \(V^i\) the mapped task data, which can be split as \(V^i_R\) for the next RKME specification procedure.

User data mapping
- Input: \(X^u = [X^u_1, \ldots, X^u_{n^u}]\) the user’s unlabeled task data across different feature spaces.
- Output: \(V^u\) the mapped user data, which is used to generate RKME-based task requirement.