The 38th Annual AAAI Conference on Artificial Intelligence
FEBRUARY 20-27, 2024 | VANCOUVER, CANADA

Towards Making Learnware Specification and Market Evolvable

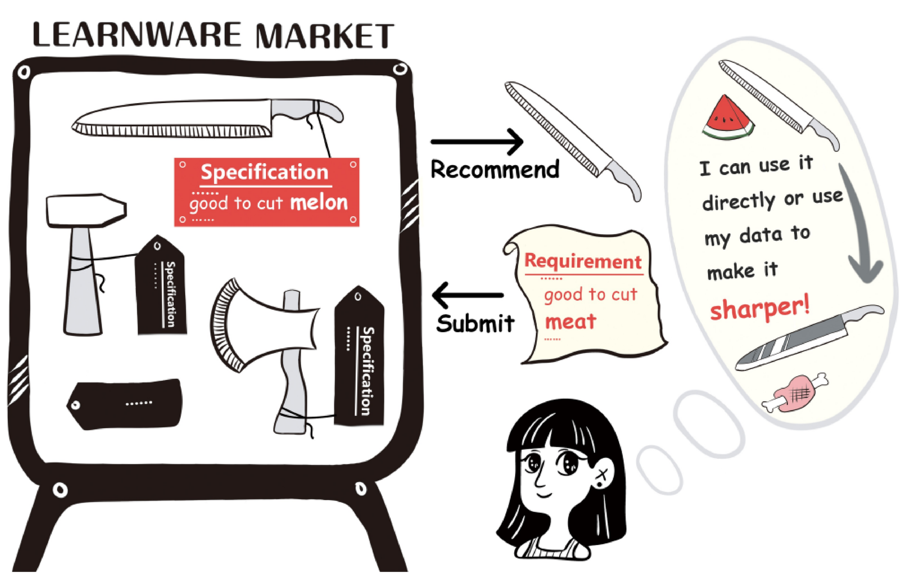
Jian-Dong Liu, Zhi-Hao Tan, Zhi-Hua Zhou

{liujd, tanzh, zhouzh}@lamda.nju.edu.cn

Learning And Mining from DatA
http://www.lamda.nju.edu.cn

1. Background: Learnware Paradigm

Learnware paradigm [Zhou, 2016; Zhou and Tan, 2024]



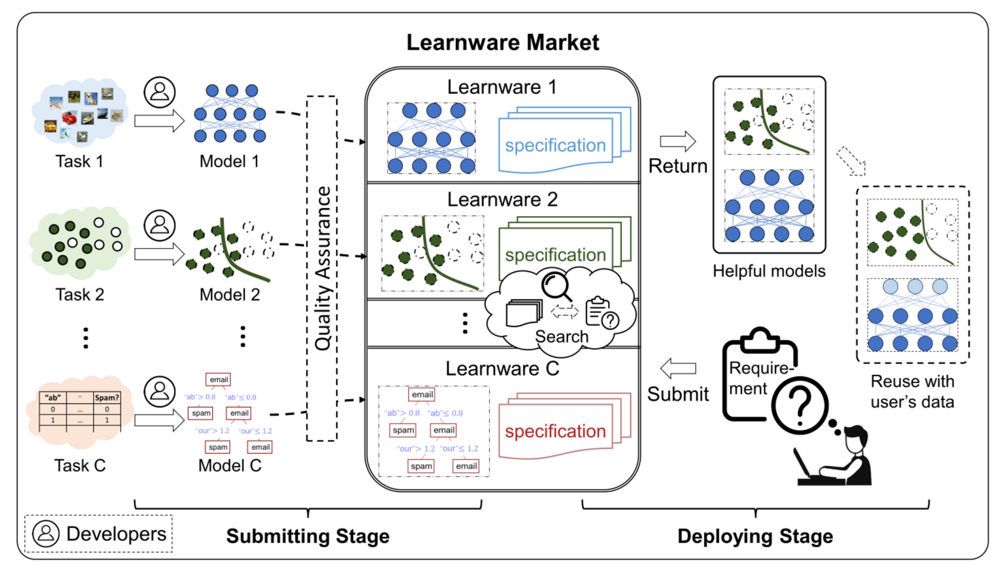
- Construct a *learnware market* containing numerous well-performed models and organize them to solve future user tasks by *identifying and reusing helpful learnware(s)* without building models from scratch.

Learnware components

- Learnware = well-performed model + *specification*
  - *Specification* describes the *specialty and utility* of the model.

Procedure of learnware paradigm

- *Submitting stage*: The learnware market manages submitted models by *specifications*.
- *Deploying stage*: The market helps the user *identify and reuse* helpful learnware(s).



Reduced Kernel Mean Embedding (RKME) specification [Zhou and Tan, 2024]

$$\min_{\beta,z}\left\|\frac{1}{m}\sum_{i=1}^{m}k(\boldsymbol{x}_i,\cdot)-\sum_{j=1}^{n}\beta_j k(\boldsymbol{z}_j,\cdot)\right\|^2_{\mathcal{H}_k}$$

KME of original data — RKME specification

- The RKME specification sketches the dataset via weighted samples in RKHS and *captures major distribution information without leaking the original data*.

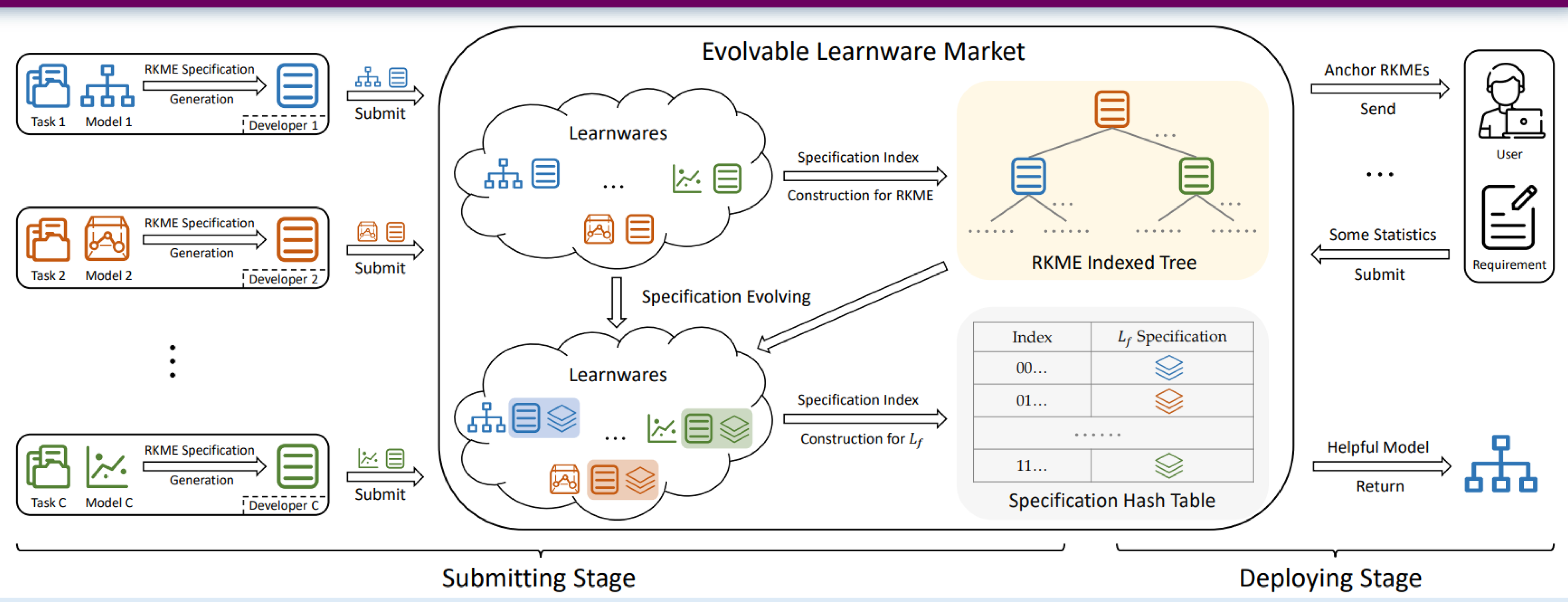
2. Contribution

Two specific key issues

- How to characterize model abilities beyond models' original tasks for *accurate learnware identification*?
- How to avoid examining the entire market for *efficient learnware identification*?

Evolvable Learnware Specification with Index (ELSI)

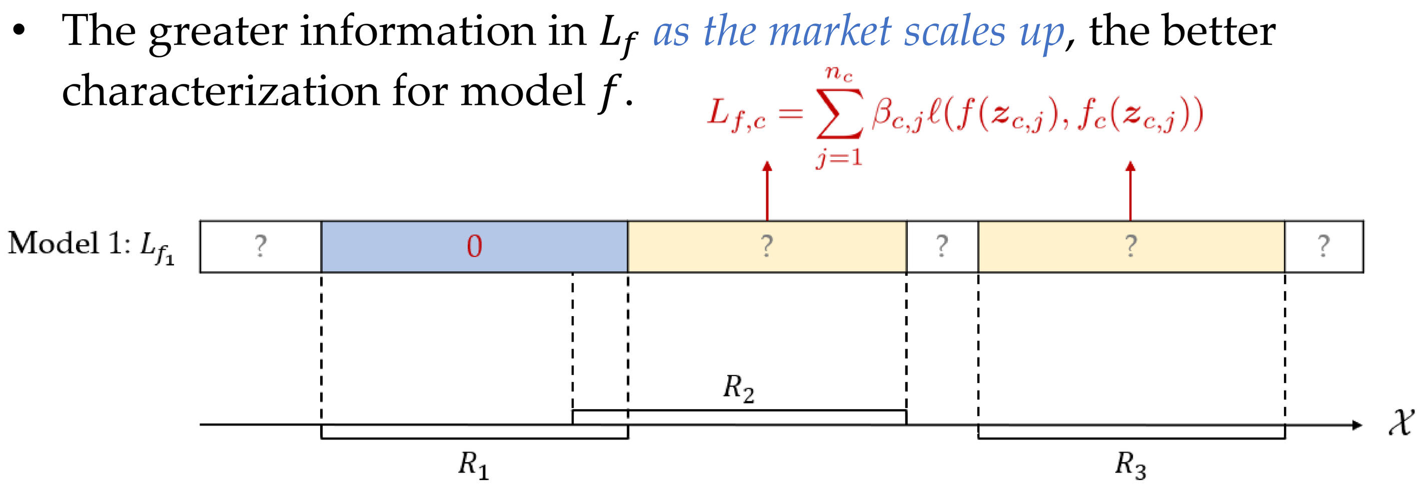
- Evolvable specification: *Accurate* learnware characterization and identification as the market continuously grows.
- Specification index: Organize specifications to ensure *efficient* operations related to both learnwares and specifications.



3. Evolvable Learnware Specification

Evolvable learnware specification (RKME, L_f)

- Loss vector $L_f \in \mathbb{R}^C$, $L_{f,c}$ denotes the loss of the model f on the c-th RKME $R_c = \{(\beta_{c,j}, \boldsymbol{z}_{c,j})\}_{j=1}^{n_c}$
- The greater information in L_f *as the market scales up*, the better characterization for model f.

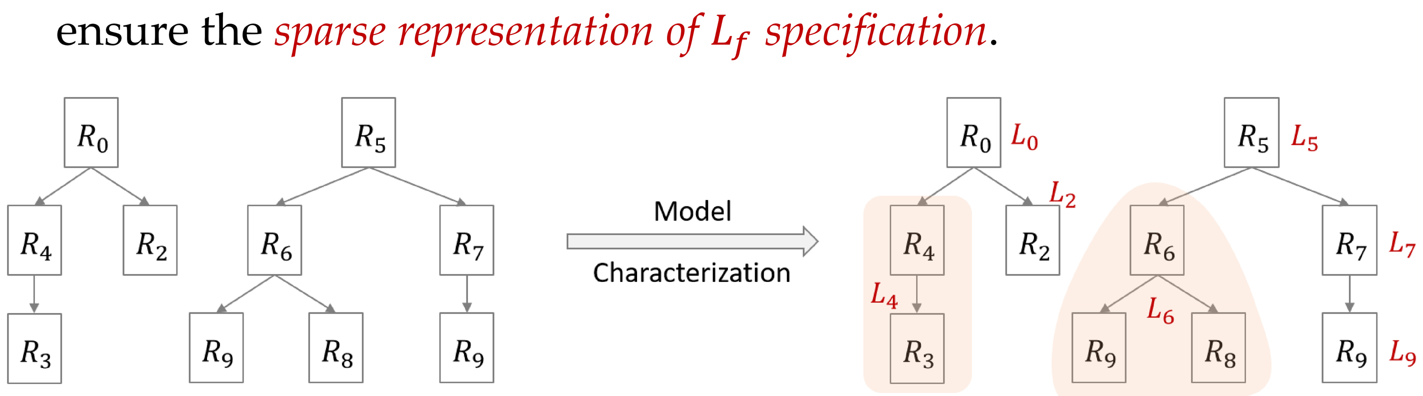


Challenge for calculating L_f specification

- Inefficiency arises from the *increasing high dimensions* of L_f specification as the market continuously grows up.

Solution: RKME specification index

- Structurally organize RKMEs via divisive hierarchical clustering to ensure the *sparse representation of L_f specification*.



4. Learnware Identification

The objective for learnware identification

- User data: $\{\boldsymbol{x}_{u,i}\}_{i=1}^{m_u}$ sampled from $\mathcal{D}_u$ with the ground-truth function h_u.

$$f_u = \underset{f\in\{f_c\}_{c=1}^{C}}{\arg\min}\,\mathcal{L}_{\mathcal{D}_u}(f,h_u) = \underset{f\in\{f_c\}_{c=1}^{C}}{\arg\min}\,\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_u}\left[\ell\left(f(\boldsymbol{x}),h_u(\boldsymbol{x})\right)\right]$$

Two challenges

- Challenge-1: Learnware performance estimation on user task
- Challenge-2: Avoid traversing the market

Solution for learnware performance estimation on user task

Assume ℓ obeys the triangle inequality and for all $c\in[C], m_c = m, n_c = n$. Let $\ell_{f,f'}: \boldsymbol{x}\mapsto\ell(f(\boldsymbol{x}), f'(\boldsymbol{x}))\in\mathcal{H}_k$, and suppose $\|\ell_{f,f'}\|_{\mathcal{H}_k}\le U, \forall f,f'\in\mathcal{F}$. Then, with probability at least $1-\delta(\delta\in(0,1))$, for all $\boldsymbol{w}\in\Delta^C$ and $f\in\mathcal{F}$, the following holds:

Approximate user task interactively by RKMEs

$$\mathcal{L}_{\mathcal{D}_u}(f,h_u)\le\boldsymbol{w}^\top L_f + U\left\|\widehat{\mu}_{\mathcal{D}_u}-\sum_{c=1}^{C}w_c\widetilde{\mu}_{\mathcal{D}_c}\right\|_{\mathcal{H}_k} + O\left(m^{-\frac{1}{2}}+n^{-\frac{1}{2}}+{m_u}^{-\frac{1}{2}}\right)+\text{constant},$$

where *Estimate learnware performance*

$$\widehat{\mu}_{\mathcal{D}_u}=\frac{1}{m_u}\sum\nolimits_{i=1}^{m_u}k(\boldsymbol{x}_{u,i},\cdot)\text{ and }\widetilde{\mu}_{\mathcal{D}_c}=\sum\nolimits_{j=1}^{n_c}\beta_{c,j}k(\boldsymbol{z}_{c,j},\cdot).$$

- Minimize the second term $\boldsymbol{w}_u = \underset{\boldsymbol{w}\in\Delta^M}{\arg\min}\left\|\widehat{\mu}_{\mathcal{D}_u}-\sum\nolimits_{c=1}^{C}w_c\widetilde{\mu}_{\mathcal{D}_c}\right\|_{\mathcal{H}_k}$
- Objective convertation:

$$f_u = \underset{f\in\{f_c\}_{c=1}^{C}}{\arg\min}\,\mathcal{L}_{\mathcal{D}_u}(f,h_u) \Longrightarrow f_u = \underset{f\in\{f_c\}_{c=1}^{C}}{\arg\min}\,\boldsymbol{w}_u^\top L_f$$

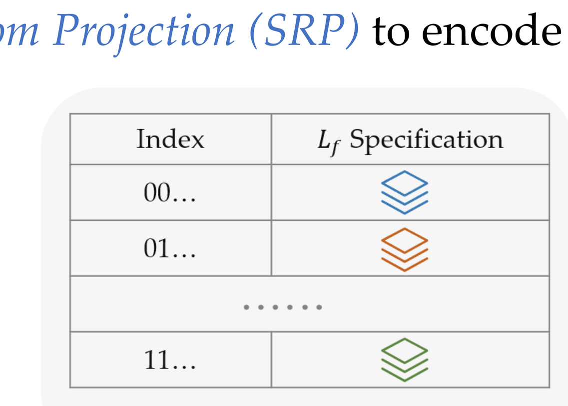
Solution for efficient learnware identification

- Using existing hash methods to converting *inner product* into *cosine similarity*.

$$f_u = \underset{f\in\{f_c\}_{c=1}^{C}}{\arg\min}\,\boldsymbol{w}_u^\top L_f = \underset{f\in\{f_c\}_{c=1}^{C}}{\arg\max}\,\frac{p(\boldsymbol{w}_u)^\top q(L_f)}{\|p(\boldsymbol{w}_u)\|_2\|q(L_f)\|_2}$$

- L_f specification index: Employing *Signed Random Projection (SRP)* to encode L_f specifications as indexes.

$$\mathbb{P}\left[h_{\boldsymbol{a}}^{srp}(p(\boldsymbol{w}_u))h_{\boldsymbol{a}}^{srp}(q(L_f))\right] = 1-\frac{1}{\pi}\cos^{-1}\left(\frac{p(\boldsymbol{w}_u)^\top q(L_f)}{\|p(\boldsymbol{w}_u)\|_2\|q(L_f)\|_2}\right)$$



5. Experiments

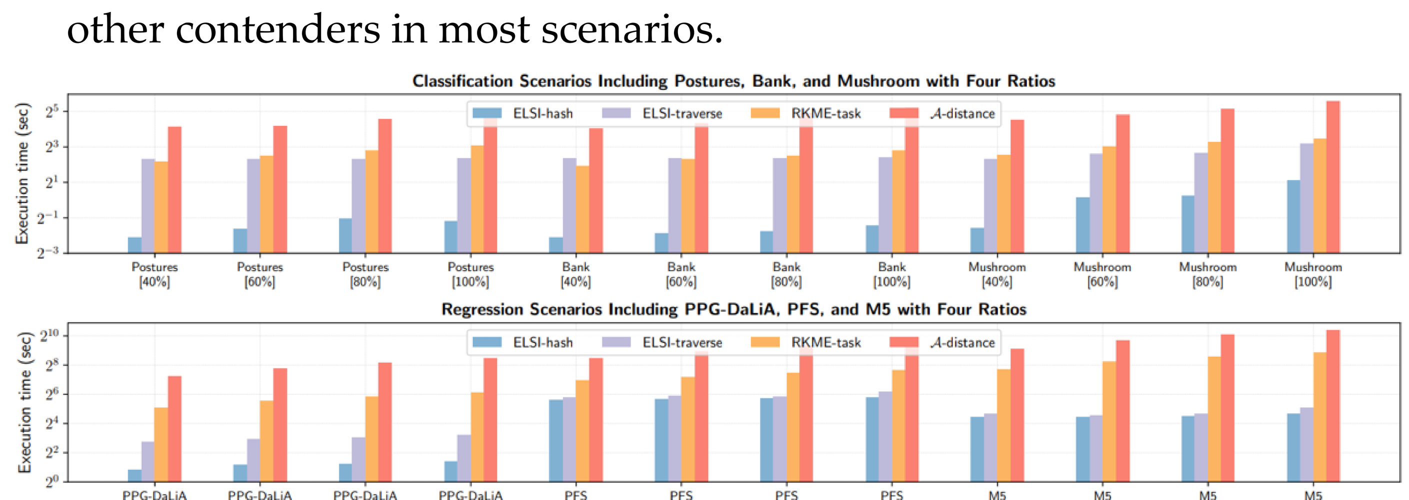
Learnware identification performance

- *ELSI-traverse* achieves the best performance, and *ELSI-hash*, the efficient version, closely matches it and still outperforms all other contenders.

Scenario (ℓ)	Ratio (%)	Random	$\mathcal{A}$-distance	RKME-task	ELSI-hash	ELSI-traverse
Postures (Error Rate)	40	52.20 ± 0.87	42.34 ± 1.57	42.44 ± 1.82	∘ **33.93 ± 2.15**	∘ **33.93 ± 2.15**
	60	51.72 ± 0.58	36.62 ± 1.17	36.34 ± 1.77	∘ **27.45 ± 1.93**	∘ **27.45 ± 1.93**
	80	51.56 ± 0.60	31.40 ± 0.63	31.16 ± 0.81	∘ 22.15 ± 1.48	∘ **22.03 ± 1.27**
	100	51.59	23.43	23.43	**11.08**	11.27
Bank (Error Rate)	40	15.53 ± 1.04	15.98 ± 1.70	15.00 ± 0.58	∘ 12.41 ± 0.18	∘ **12.38 ± 0.19**
	60	15.20 ± 0.59	15.25 ± 0.90	14.32 ± 0.49	∘ 11.75 ± 0.35	∘ **11.74 ± 0.35**
	80	15.06 ± 0.21	14.93 ± 0.34	14.26 ± 0.51	∘ 12.19 ± 0.35	∘ **12.17 ± 0.35**
	100	14.83	14.64	14.13	**12.11**	12.31
Mushroom (Error Rate)	40	44.09 ± 0.68	30.64 ± 2.47	30.47 ± 2.22	∘ 22.27 ± 2.65	∘ **22.22 ± 2.68**
	60	43.94 ± 0.58	26.10 ± 2.15	24.93 ± 2.02	∘ **20.38 ± 1.86**	∘ 21.20 ± 1.80
	80	43.67 ± 0.46	21.18 ± 1.67	19.76 ± 0.84	∘ 15.74 ± 2.18	∘ **15.67 ± 2.04**
	100	43.66	16.90	16.29	**6.23**	6.29
PPG-DaLiA (RMSE)	40	37.01 ± 1.19	30.74 ± 1.25	29.51 ± 0.91	∘ 17.68 ± 0.44	∘ **17.36 ± 0.52**
	60	36.42 ± 1.21	27.48 ± 0.79	26.30 ± 0.59	∘ 16.21 ± 0.88	∘ **15.17 ± 0.83**
	80	36.38 ± 0.45	23.89 ± 0.62	23.28 ± 0.36	∘ 14.65 ± 0.54	∘ **12.70 ± 0.35**
	100	36.43	20.62	20.62	13.88	**11.11**
PFS (RMSE)	40	2.46 ± 0.12	2.16 ± 0.10	2.11 ± 0.15	2.03 ± 0.15	**2.00 ± 0.13**
	60	2.52 ± 0.14	2.17 ± 0.10	2.18 ± 0.09	∘ **1.98 ± 0.07**	∘ 1.99 ± 0.06
	80	2.57 ± 0.06	2.22 ± 0.08	2.18 ± 0.09	∘ 2.04 ± 0.15	∘ **1.99 ± 0.10**
	100	2.58	2.21	2.21	2.03	**1.97**
M5 (RMSE)	40	3.28 ± 0.35	4.17 ± 1.78	2.33 ± 0.07	∘ 2.26 ± 0.09	∘ **2.22 ± 0.05**
	60	3.35 ± 0.28	4.80 ± 1.53	2.28 ± 0.06	∘ 2.22 ± 0.05	∘ **2.19 ± 0.04**
	80	3.28 ± 0.17	4.28 ± 1.30	2.23 ± 0.05	2.21 ± 0.06	∘ **2.16 ± 0.05**
	100	3.36	5.25	2.19	**2.14**	**2.14**

Learnware identification efficiency

- *ELSI-hash* achieves the highest efficiency and *ELSI-traverse* outperforms other contenders in most scenarios.



6. Conclusion

- We make the first attempt to establish evolvable learnware specifications, aiming for *increasingly accurate characterization of model abilities beyond their original training tasks* as the market continuously grows, thereby constantly facilitating the evolution and enhancement of the overall market capability.
- Through organizing learnwares and constructing specification indexes, we propose an approach called *Evolvable Learnware Specification with Index (ELSI)*, which could achieve evolvable learnware specifications and corresponding efficient learnware identification for users without leaking raw data. As the key components of our approach, specification indexes are established based on the RKME indexed tree and the specification hash table.
- Extensive experimental results on a *learnware market encompassing thousands of models and covering six real-world scenarios* validate the effectiveness and efficiency of our approach.