

T2S: Tokenized Skill Scaling for Lifelong Imitation Learning

Hongquan Zhang^{1,2,†}, Jingyu Gong^{1,3,4,†}, Zhizhong Zhang^{1,4}, Xin Tan^{1,3}, YuanXie^{1,2,3,*}

Abstract—The main challenge in lifelong imitation learning lies in the balance between mitigating catastrophic forgetting of previous skills while maintaining sufficient capacity for acquiring new ones. However, current approaches typically address these aspects in isolation, overlooking their internal correlation in lifelong skill acquisition. We address this limitation with a unified framework named Tokenized Skill Scaling (T2S). Specifically, by tokenizing the model parameters, the linear parameter mapping of the traditional transformer is transformed into cross-attention between input and learnable tokens, thereby enhancing model scalability through the easy extension of new tokens. Additionally, we introduce language-guided skill scaling to transfer knowledge across tasks efficiently and avoid linearly growing parameters. Extensive experiments across diverse tasks demonstrate that T2S: 1) effectively prevents catastrophic forgetting (achieving an average NBT of 1.0% across the three LIBERO task suites), 2) excels in new skill scaling with minimal increases in trainable parameters (needing only 8.0% trainable tokens in an average of lifelong tasks), and 3) enables efficient knowledge transfer between tasks (achieving an average FWT of 77.7% across the three LIBERO task suites), offering a promising solution for lifelong imitation learning.

I. INTRODUCTION

Imitation learning enables robots to acquire complex task skills by observing and mimicking expert demonstrations, facilitating the deployment of robust strategies across various sequential decision-making tasks [1], [2], [3], [4], [5], [6]. This approach is particularly well-suited for scenarios where designing reward functions is challenging [7], [8], [9], as well as for complex tasks where trial-and-error learning is too costly or poses safety risks [10], [11].

However, most existing research lacks the ability to support incremental skill acquisition and cannot facilitate lifelong skill adaptation or the open-ended expansion of skill scaling [12]. This drawback conflicts with the dynamic and consequent demands of real-world task flows, *i.e.*, when robots need to expand their skills incrementally, traditional frameworks often face the dual challenges of catastrophic forgetting and skill capacity saturation [13]. The former arises from the parameter-sharing mechanism in neural networks, where the acquisition of new skills inevitably interfere with learned knowledge, akin to the memory conflict phenomenon in connectionist networks as

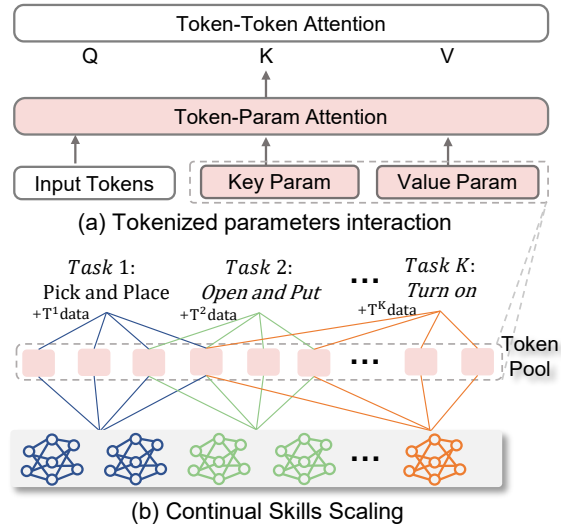


Fig. 1. (a): We convert the traditional linear mapping of parameters into tokenized interactions to improve the scalability of the model through token extensions. (b): Through token sharing between tasks, we achieve efficient utilization of parameters while skills are continuously scaling.

described by McCloskey et al. [14] in cognitive science. The latter is due to the loss of plasticity in learning from extended training on new data in a single neural network [15]. Consequently, constructing lifelong imitation learning frameworks capable of simultaneously maintaining knowledge retention and architectural scalability has become a central challenge for promoting robots’ adaptation to open environments.

LOTUS [16] employs hierarchical skill learning to acquire new skills continuously and integrates them into a dynamic skills library, effectively improve scalability but still struggle with catastrophic forgetting. M2Distill [17] addresses catastrophic forgetting by transferring multi-modal knowledge from the teacher model to the student model using knowledge distillation techniques. However, this approach lacks scalability and suffers from reduced plasticity in lifelong learning, thereby limiting the model’s ability to acquire new skills over time. Despite recent advancements in lifelong imitation learning, existing methods predominantly focus on either skill capacity saturation or catastrophic forgetting in isolation, overlooking the internal correlation of these challenges in lifelong skill acquisition.

To address the limitations of current methods, we propose a **Tokenized Skill Scaling (T2S)** approach. Specifically, we transform the traditional linear parameter mapping in the Transformer architecture with a cross-attention mechanism based on token-parameter interactions, as illustrated in Fig.

¹School of Computer Science and Technology, East China Normal University, Shanghai 200062, China.

²Shanghai Innovation Institute, Shanghai 200062, China.

³Chongqing Key Laboratory of Precision Optics, Chongqing Institute of East China Normal University, Chongqing, China.

⁴Shanghai Key Laboratory of Computer Software Evaluating and Testing, Shanghai Development Center of Computer, Shanghai, China.

*Corresponding author (e-mails: yxie@cs.ecnu.edu.cn), [†]Equal Contribution.

1(a). This full parameter tokenization enables the acquisition of new skills by simply extending additional tokens, thereby addressing the loss of model plasticity commonly encountered in lifelong learning. However, naïvely expanding new tokens for each task results in a linear increase in model parameters, leading to significant storage overhead. To mitigate this, we introduce a token pool, from which relevant tokens are selected based on task descriptions, as depicted in Fig. 1(b). Tokens shared across tasks can be seen as tokenized atomic skills. When new tasks are introduced, task descriptions retrieve a subset of atomic skills from the token pool and combine them with newly added tokens to acquire task-specific skills. With the progressive acquisition of skills, the number of atomic skills in the token pool increases, thereby providing more shared knowledge for learning subsequent skills. This mechanism promotes inter-task skill sharing, allowing for the efficient acquisition of new skills with only a modest increase in tokens, while effectively controlling the growth of model parameters. In summary, our main contributions are as follows:

- We introduce a scalable lifelong imitation learning framework that enhances model scalability by tokenizing all learnable parameters.
- We propose a language-guided token activation and selection strategy that effectively transfers shared knowledge across tasks while mitigating the linear growth of model parameters.
- Our proposed method achieves state-of-the-art performance in mitigating catastrophic forgetting.

II. RELATED WORK

A. Lifelong Imitation Learning

Lifelong learning has made significant progress in various fields, including computer vision [18], natural language processing [19], multi-modal large language models [20], and robotics [21], [22]. This work focuses on lifelong imitation learning, which enables robots to continuously acquire new skills through human demonstrations without forgetting previously learned skills. In recent years, numerous researchers have investigated this field. ER [23] mitigates catastrophic forgetting by retaining a portion of the demonstration trajectory for each task. EWC[24] prevents forgetting by calculating the importance of model parameters and constraining updates to parameters critical to previous tasks while learning new ones. BUDS [25] employs a bottom-up approach to autonomously identify and organize skills from unsegmented, long-duration demonstration data, allowing robots to effectively manage complex and prolonged manipulation tasks. LOTUS [16] uses open-vocabulary visual models for skill discovery and meta-controllers for skill integration, enabling robots to learn and adapt to new tasks continuously. M2Distill [17] addresses catastrophic forgetting by utilizing multimodal knowledge distillation to maintain consistency in the potential space across visual, verbal, and motor distributions during skill learning. In contrast, T2S enhances model scalability by

tokenizing parameters and employing task descriptions to guide inter-task token share, allowing robots to learn new skills throughout their lifetime without being limited by linearly growing parameters and catastrophic forgetting.

B. Parameter Scaling of Transformer

In recent years, with the rapid advancement of large language models such as GPT [26], DeepSeek [27], and LLaMA [28], researchers have increasingly focused on efficiently scaling models from small to large sizes, driven by the substantial overhead of intensive training. Gong et al. [29] propose a stacking algorithm that transfers knowledge from a shallow model to a deeper one by progressively applying stacking to accelerate BERT [30] training. LiGO [31] tackles the challenge of efficiently scaling pre-trained transformers by learning parameter growth through a factorized approach, decomposing the linear transformation into Kronecker-factorized width- and depth-growth operators. bert2BERT [32] facilitates knowledge transfer from small models via function-preserving parameter initialization and accelerates training using a two-stage strategy. Tokenformer [33] enhances architectural flexibility by leveraging the attention mechanism not only for computations among input tokens but also for interactions between tokens and model parameters. In this work, inspired by Tokenformer’s high extensibility, we adopt its parameter tokenization approach as the foundation for lifelong imitation learning.

III. METHOD

Our method builds a behavior cloning network based on the token-Parameter attention (Pattention) layer, augmented by a tailored training and deployment strategy for lifelong learning in robotics. In subsequent sections, we begin with a problem formulation for lifelong imitation learning. We then present the core architecture of the Pattention-based behavior cloning model, emphasizing its scalability and flexibility. Finally, we introduce Language-Guided Skill Scaling—a novel token-sharing and expansion mechanism developed to fully harness the model’s potential in lifelong imitation learning.

A. Problem Formulation

Lifelong Robot Learning constitutes a fundamental aspect of embodied intelligence, distinguished by its capability to incrementally acquire and refine skills through continuous engagement with a sequence of learning tasks $\{T^1, \dots, T^K\}$. This foundational robot learning paradigm can be formulated using a finite-horizon Markov Decision Process (MDP), denoted as $M = (S, A, \mathcal{T}, H, \mu_0, R)$. Here, S represents the state space, A denotes the action space, $\mathcal{T} : S \times A \rightarrow S$ specifies the transition function, H is the length of each task episode, μ_0 characterizes the initial state distribution, and $R : S \times A \rightarrow \mathbb{R}$ is the reward function. Due to the sparsity of R commonly encountered in robotic learning scenarios, a binary goal predicate $g : S \rightarrow \{0, 1\}$ is employed as a replacement to explicitly indicate the achievement of task objectives. Within the lifelong learning

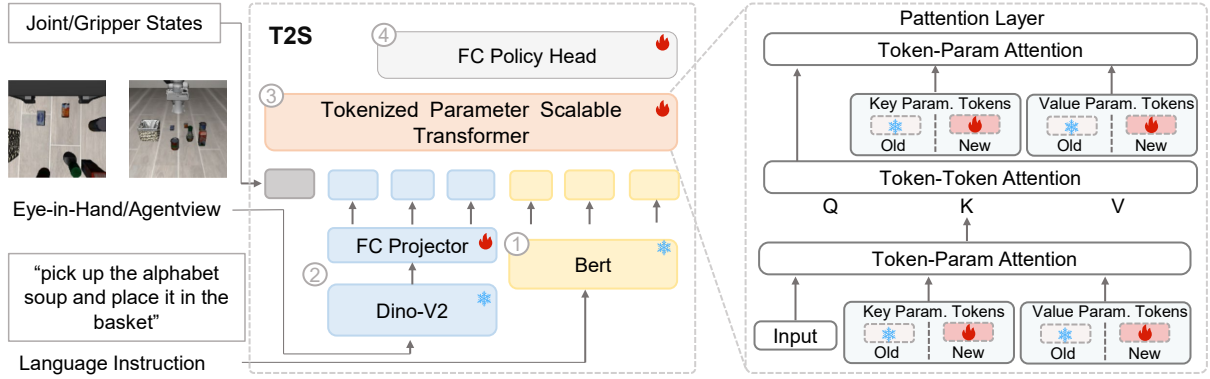


Fig. 2. An overview of T2S: On the left is the full pipeline of our model architecture, and on the right is the Attention layer in the Tokenized Parameter Scalable Transformer.

context, the robot is tasked with developing a unified policy π capable of sequentially adapting to the specific demands of each task T^k , which is uniquely characterized by its initial state distribution μ_0^k and a task-specific goal predicate g^k . In this paradigm, we assume S , A , \mathcal{T} , and H are consistent across all tasks. After sequentially experiencing tasks up to task k , the robot aims to optimize its policy to maximize the expected return, formulated as:

$$\max_{\pi} J(\pi) = \frac{1}{K} \sum_{p=1}^K \mathbb{E}_{s_t^p, a_t^p \sim \pi(\cdot; T^p), \mu_0^p} \left[\sum_{t=1}^H g^p(s_t^p) \right]. \quad (1)$$

Lifelong Imitation Learning. Given the inherent challenges associated with sparse-reward reinforcement learning, we adopt a practical scenario in which a user provides a small set of demonstrations for each task within a sequence. Specifically, we consider a lifelong imitation learning scenario, in which each task T^k is associated with N expert demonstrations $D_k = \{\tau_i^k\}_{i=1}^N$ and corresponding natural language task instruction l^k . Each demonstration trajectory τ_i^k consists of observation-action pairs, formally defined as $\tau_i^k = \{(o_t, a_t)\}_{t=0}^{L^k}$, where $L^k \leq H$. The observation o_t comprises the robot’s sensory inputs, including perceptual observations and proprioceptive data from the robot’s joints and gripper. In practice, due to partial observability in MDPs [34], observations o_t alone do not fully satisfy the Markov property. Thus, consistent with prior work, we define the state s_t as the historical sequence of observations up to time t , formally represented as: $s_t \equiv o_{\leq t} \triangleq \{(o_t, a_t)\}_{t=0}^{L^k}$. This formulation aligns with the lifelong imitation learning setting described by LIBERO [35], whose objective remains consistent with the one introduced in Eq. 1. However, during training, behavioral cloning [36] is employed as a surrogate objective, defined as follows:

$$\min_{\pi} J(\pi) = \frac{1}{K} \sum_{k=1}^K \mathbb{E}_{(o_t, a_t) \sim D_k} \left[\sum_{t=1}^{L^k} -\log \pi(a_t | o_{\leq t}; T^k) \right]. \quad (2)$$

B. Attention-Based Behavior Cloning

The architecture of T2S consists of four main components, as illustrated in Fig. 2: 1) A frozen Large Language Model

(LLM) that maps the task instruction into language tokens, 2) a frozen visual encoder that transforms visual inputs into a set of tokens and subsequently projects them into the same dimensional space as the language tokens, 3) a token-scalable transformer that continuously encodes these tokens into action tokens, and 4) a policy head that decodes the action tokens into robot-executable actions. During training, the model is trained end-to-end using a mean squared error objective on the demonstration data. We take a Tokenized Parameter Scalable Transformer (TPST) as the basic block and build the network in LIBERO manner [35]. This design provides a fundamental advantage for lifelong imitation learning.

In this work, we build the TPST upon the Tokenformer [33]. Tokenformer operates through token-parameter attention (Pattention) layers, wherein a set of learnable tokens are treated as model parameters. These tokens interact with input tokens through cross-attention mechanisms, allowing the model to capture the relationships between the input and parameter tokens. For each TPST block, the input and output tokens are marked as $\mathcal{I} \in \mathbb{R}^{T \times d_1}$ and $\mathcal{O} \in \mathbb{R}^{T \times d_2}$ respectively. Here, T denotes the sequence length, and d_1 and d_2 are the input and output dimensions. Two sets of n learnable parameter tokens are introduced: $K_P \in \mathbb{R}^{n \times d_1}$ for the keys and $V_P \in \mathbb{R}^{n \times d_2}$ for the values. The output \mathcal{O} from the scaled dot-product attention mechanism is computed as follows:

$$\text{Attention}(X, K_P, V_P) = \Theta(X \cdot K_P^\top) \cdot V_P, \quad (3)$$

where Θ denotes the softmax operation. We adopt Pattention in place of the traditional Multi-Head Attention (MHA) and Feed-Forward Networks (FFN) in the transformer due to the flexible design of the Pattention layer, which facilitates the faster incremental development of larger models while reusing parameters from smaller, pre-trained counterparts. However, despite its advantages in incremental training, Pattention is insufficient for lifelong imitation learning, arising from its linear growing tokens and inability to maintain knowledge from previously acquired skills when scaling to new tasks, kindly refer to the answer to question 3 (Sec. IV-D) for experimental proofs.

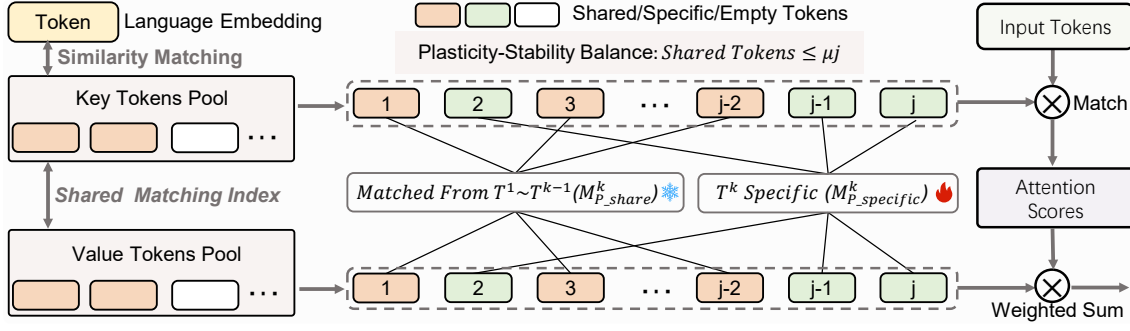


Fig. 3. The visual explanation of Language-Guided Skill Scaling of task $k(k > 1)$.

C. Language-Guided Skill Scaling

A straightforward solution to address the drawbacks of Pattention in lifelong imitation learning would be to learn independent sets of tokens for each task and identify them via task IDs. However, this approach leads to a linear increase in model parameters as the number of tasks grows, resulting in significant storage overhead. To overcome this limitation, we propose a language-guided parameter token activation and expansion approach (shown in Fig. 3). Such a sharing mechanism enables continuous scaling of new skills with much less extra storage (validated in Sec. IV-D). Specifically, within each Pattention layer, we construct token pools $KP \in \mathbb{R}^{n \times d}$ and $VP \in \mathbb{R}^{n \times d}$ separately for keys and values, where n denotes the number of tokens and d represents the embedding dimension. For each task, the amount of tokens is $j = n/K$. These token pools are managed by a unified global mask $M_G \in \{0, 1\}^n$, indicating whether each token is utilized. A language embedding e^k is further leveraged to identify and select the most relevant tokens as follows:

$$M_P^k = \text{Top-}\mathcal{K}(\langle e^k, KP \rangle, j), \quad (4)$$

where \langle, \rangle denotes the cosine similarity and $M_P^k \in \{0, 1\}^n$ is the mask of selected tokens for task k . Particularly, the M_P^k is shared between KP and VP . For the first task, all the selected tokens are trained as task-specific tokens, while in the following lifelong tasks, the selected tokens consist of two components: 1) task-shared tokens $M_{P_share}^k = M_P^k \wedge M_G$, which are reused from previous tasks, and 2) task-specific tokens $M_{P_specific}^k = M_P^k \wedge \neg(M_P^k \wedge M_G)$, which are newly initialized. To ensure that the new policy contains a sufficient number of parameter tokens, we introduce a hyper-parameter μ to regulate the number of shared tokens. Specifically, the number of shared tokens is constrained to be at most $\lfloor \mu \times j \rfloor$, where $\mu \in [0, 1]$. Finally, the selected key and value tokens interact with the input as defined in Eq. 3.

The entire training and evaluation procedure is presented in Algorithm 1; we omit the backpropagation process for simplicity. During the training stage (lines 6–23), all selected tokens are trained for the first task since no prior reference knowledge is available. This step serves as the initialization for the entire lifelong learning process, providing prior knowledge for subsequent tasks. In later tasks, shared knowledge can be extracted from previously learned tokens. As

lifelong learning progresses, the token pool gradually accumulates knowledge, enabling subsequent tasks to achieve their objectives by training only a small number of task-specific tokens. During the evaluation stage (lines 25–28), all previously learned tasks are evaluated, and the mask M_P^i is applied to select corresponding tokens from KP and KV for each task i .

Algorithm 1 Overall training and evaluation procedure of Language-Guided Skill Scaling

- 1: **Input:** Token Pools KP and VP , global mask M_G , task mask list M_P^K , hyper-parameters μ , the number of tasks K , language embedding e , input \mathcal{I} , demonstration D
- 2: **Output:** Interacted Tokens O
- 3: Random initialize KP and VP , initialize M_G and M_P^K to all 0
- 4: **for** $k=1$ to K **do**
- 5: **Training**
- 6: $M_P^k \leftarrow e^k, KP$ ▷ Via Eq.4
- 7: **if** $k \neq 1$ **then**
- 8: $M_{P_sh}^k \leftarrow M_P^k \wedge M_G$ ▷ Shared
- 9: **if** $\text{sum}(M_{P_sh}^k) > \lfloor \mu \times j \rfloor$ **then**
- 10: $M_{P_sh}^k \leftarrow \text{Reduce tokens until to } \lfloor \mu \times j \rfloor$
- 11: $M_P^k \leftarrow \text{update } M_P^k \text{ via } M_{P_sh}^k$
- 12: $M_{P_sp}^k \leftarrow M_P^k \wedge \neg(M_P^k \wedge M_G)$ ▷ Specific
- 13: $M_G \leftarrow M_G \vee M_P^k, M_P^K \leftarrow M_P^k$
- 14: **for** epoch=1 to epochs **do**
- 15: **if** $k=1$ **then**
- 16: $K, V \leftarrow KP \times M_P^k, KV \times M_P^k,$
- 17: $S \leftarrow \Theta(\mathcal{I} \cdot K^\top), O \leftarrow S \cdot V$
- 18: **else**
- 19: $K_{sh}, V_{sh} \leftarrow KP \times M_{P_sh}^k, KV \times M_{P_sh}^k$
- 20: $K_{sp}, V_{sp} \leftarrow KP \times M_{P_sp}^k, KV \times M_{P_sp}^k$
- 21: $S \leftarrow \Theta(\mathcal{I} \cdot K_{sh}^\top(\text{detached}) + \mathcal{I} \cdot K_{sp}^\top)$
- 22: $O \leftarrow S \cdot V_{sh}(\text{detached}) + S \cdot V_{sp}$
- 23: **Evaluation**
- 24: **for** $i=1$ to k **do** ▷ Evaluation all tasks before task k
- 25: $M_P^i \leftarrow M_P^K, i$
- 26: $K, V \leftarrow KP \times M_P^i, KV \times M_P^i$
- 27: $O \leftarrow \mathcal{I}, K, V$ ▷ Via Eq. 3

TABLE I

PERFORMANCE OF THE PROPOSED METHOD COMPARED WITH STATE-OF-THE-ART, CORRESPONDING RESULTS ARE BORROWED FROM M2DISTILL AND LIBERO. THE REPORTED VALUES ARE AVERAGES FROM THREE SEEDS, INCLUDING THE MEAN AND STANDARD ERROR. ALL METRICS ARE MEASURED BASED ON SUCCESS RATES (%).

Methods	LIBERO-OBJECT			LIBERO-GOAL			LIBERO-SPATIAL		
	FWT(\nearrow)	NBT(\searrow)	AUC(\nearrow)	FWT(\nearrow)	NBT(\searrow)	AUC(\nearrow)	FWT(\nearrow)	NBT(\searrow)	AUC(\nearrow)
SEQUENTIAL	62.0 (\pm 1.0)	63.0 (\pm 2.0)	30.0 (\pm 1.0)	55.0 (\pm 1.0)	70.0 (\pm 1.0)	23.0 (\pm 1.0)	72.0 (\pm 1.0)	81.0 (\pm 1.0)	20.0 (\pm 1.0)
EWC [24]	56.0 (\pm 3.0)	69.0 (\pm 2.0)	16.0 (\pm 2.0)	32.0 (\pm 2.0)	48.0 (\pm 3.0)	6.0 (\pm 1.0)	23.0 (\pm 1.0)	33.0 (\pm 1.0)	6.0 (\pm 1.0)
ER [23]	56.0 (\pm 1.0)	24.0 (\pm 1.0)	49.0 (\pm 1.0)	53.0 (\pm 1.0)	36.0 (\pm 1.0)	47.0 (\pm 2.0)	65.0 (\pm 3.0)	27.0 (\pm 3.0)	56.0 (\pm 1.0)
LOTUS [16]	74.0 (\pm 3.0)	11.0 (\pm 1.0)	65.0 (\pm 3.0)	61.0 (\pm 1.0)	30.0 (\pm 1.0)	56.0 (\pm 1.0)	-	-	-
M2Distill [17]	<u>75.0</u> (\pm 3.0)	<u>8.0</u> (\pm 5.0)	69.0 (\pm 4.0)	<u>71.0</u> (\pm 1.0)	20.0 (\pm 3.0)	57.0 (\pm 2.0)	<u>74.0</u> (\pm 1.0)	11.0 (\pm 1.0)	61.0 (\pm 2.0)
T2S	75.0 (\pm 2.0)	4.0 (\pm 1.0)	<u>66.0</u> (\pm 2.0)	84.0 (\pm 1.0)	-1.0 (\pm 2.0)	70.0 (\pm 2.0)	74.0 (\pm 1.0)	0.0 (\pm 1.0)	63.0 (\pm 2.0)

IV. EXPERIMENTAL EVALUATION

A. Experimental Setup

Simulation Experiments. We conduct experiments in simulation scenarios using the task suites from the lifelong robot learning benchmark LIBERO [35]. Specifically, we select three suites, namely LIBERO-OBJECT (10 tasks), LIBERO-GOAL (10 tasks), and LIBERO-SPATIAL (10 tasks). These benchmarks evaluate the robot’s ability to transfer knowledge between different objects (declarative knowledge) and various instructions (procedural knowledge), respectively. Additional implementation details are provided in the supplementary materials.

Evaluation Metrics. To systematically evaluate the effectiveness of lifelong imitation learning methods for robotic manipulation tasks, we employ three standardized evaluation metrics: Forward Transfer (FWT), Negative Backward Transfer (NBT), and Area Under the Success Rate Curve (AUC), in alignment with prior research [16], [17]. These metrics are based on task success rates across training procedures, providing a more accurate and meaningful representation of actual manipulation performance. Specifically, FWT quantifies the policy’s adaptability and its ability to generalize effectively to new tasks, higher FWT scores indicate improved learning efficiency and successful transfer of prior knowledge. In T2S, we evaluate it after one epoch adaptation. In contrast, NBT evaluates how well the policy maintains knowledge acquired in previous tasks when exposed to new tasks, where lower values signify better retention and less interference. Lastly, AUC aggregates task success rates over the entire sequence of encountered tasks, serving as a comprehensive indicator of lifelong learning performance. Higher AUC values represent sustained effectiveness and robust task success throughout the robot’s operational lifetime. Denote $r_{i,j}$ as the agent’s performance on task j immediately after learning from the first i tasks. These metrics are defined as follows: $FWT = \sum_{m \in [M]} \frac{r_{m,m}}{M}$, $NBT = \sum_{m \in [M]} \frac{NBT_m}{M}$, and $AUC = \sum_{m \in [M]} \frac{AUC_m}{M}$, where $NBT_m = \frac{1}{M-m} \sum_{q=m+1}^M (r_{m,m} - r_{q,m})$, $AUC_m = \frac{1}{M-m+1} (r_{m,m} + \sum_{q=m+1}^M r_{q,m})$.

B. Compared Methods

We compare our method against the following baselines:

- **SEQUENTIAL**, which naively fine-tunes new tasks sequentially using the ResNet-Transformer architecture from LIBERO.
- **EWC** [24], a regularization-based continual learning approach that mitigates catastrophic forgetting by penalizing deviations from previously learned model parameters.
- **ER** [23], an Experience Replay baseline with a limit of 1000 trajectories in the replay buffer;
- **LOTUS** [16], a hierarchical imitation learning approach that integrates experience replay with open-vocabulary visual representation models for continuous discovery.
- **M2Distill** [17], a multi-modal distillation-based method for lifelong imitation learning that preserves a consistent latent space across vision, language, and action distributions throughout the learning process.

It is worth noting that, unlike parameter-efficient fine-tuning methods such as LoRA[37], adapters[38], and prompt tuning[39]—which rely heavily on the prior knowledge embedded in pre-trained models—our approach employs a dynamic model architecture that can be trained from scratch. This architecture enables progressive knowledge accumulation, thereby facilitating lifelong learning of embodied skills.

C. Quantitative Results

Table I provides a comprehensive evaluation of T2S against baseline methods in simulation scenarios. It demonstrates that T2S consistently surpasses most baseline methods across the three evaluation metrics. Specifically, our method exhibits notable effectiveness in mitigating catastrophic forgetting within the LIBERO-GOAL and LIBERO-SPATIAL task suites, achieving reductions in NBT by 21% and 11%, respectively, effectively resulting in zero forgetting. In the LIBERO-OBJECT suite, increased variability in task layouts introduces additional challenges for visual perception, marginally reducing the AUC. Nevertheless, T2S still consistently outperforms the state-of-the-art baseline, M2Distill, in terms of robustness against forgetting.

TABLE II

PERFORMANCE OF THE DIFFERENT TASK ORDERS, ALL METRICS ARE MEASURED BASED ON SUCCESS RATE (%).

Methods	LIBERO-GOAL		
	FWT(\nearrow)	NBT(\searrow)	AUC(\nearrow)
Default	84.0 (± 1.0)	-1.0 (± 2.0)	70.0 (± 2.0)
Shuffle	80.0 (± 1.0)	-1.0 (± 1.0)	69.6 (± 1.0)

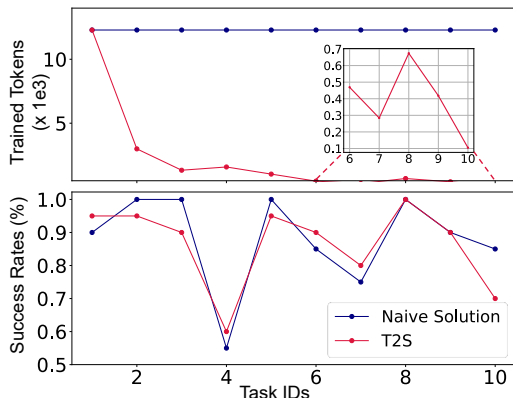


Fig. 4. We compare T2S with the naive solution in terms of the number of trainable tokens (for each task) and success rate, showing significantly fewer trainable parameters with little difference in success rates across tasks.

D. Ablation Analysis

Token Efficiency. we compare T2S with the naive approach of learning separate tokens for each task on the LIBERO-GOAL task suit. As shown in Fig. 4, the top half displays the number of tokens required for training each task in both methods, while the bottom half presents the success rate for each task. T2S can benefit from shared tokens selected from the token pool under the guidance of task instruction. Thus, it can be observed that the number of trainable parameters required for each task is reduced heavily, while achieving a comparable success rate. It indicates that the proposed method can effectively leverage shared knowledge across tasks and minimize the storage burden.

Knowledge Transfer. Fig. 5 illustrates how the allocation of shared and task-specific tokens influences the transfer of different types of knowledge. On one hand, procedural knowledge transfer generally outperforms declarative knowledge across different μ values, suggesting that procedural knowledge is more easily shared between tasks. On the other hand, at $\mu = 0.9$, where only 10% of task-specific tokens are required for each task, T2S still exhibits effective knowledge transfer, demonstrating that tokenized skills can be efficiently shared across tasks.

Task Order Robustness. To assess the sensitivity of our method to task ordering, we conduct experiments by randomly shuffling the original task sequence and repeating training five times. As shown in Table II, the forward transfer (FWT) varies depending on the informativeness of the initial tasks. However, the forgetting rate (NBT) remains consistently low across different permutations. This

TABLE III

PERFORMANCE OF THE DIFFERENT LANGUAGE GUIDANCE, ALL METRICS ARE MEASURED BASED ON SUCCESS RATE (%).

Methods	LIBERO-GOAL		
	FWT(\nearrow)	NBT(\searrow)	AUC(\nearrow)
Language	84.0 (± 1.0)	-1.0 (± 2.0)	70.0 (± 2.0)
TASK IDs	75.7 (± 1.0)	-2.0 (± 1.0)	67.2 (± 2.0)

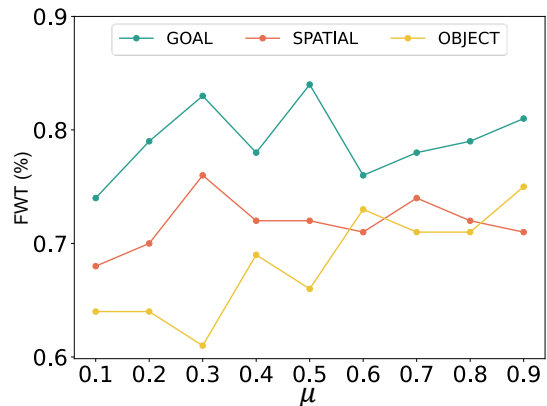


Fig. 5. The FWT of T2S for different μ values, including LIBERO-GOAL, LIBERO-SPATIAL, and LIBERO-OBJECT task suits, analyzing the impact of μ on different types of knowledge transfer.

robustness indicates that the proposed language-guided token sharing mechanism effectively preserves previously acquired knowledge, allowing the system to maintain stability under varying task orders.

Semantic Guidance. To evaluate the effectiveness of language-guided token selection, we perform an ablation study comparing it with a simpler baseline that activates tokens based solely on task IDs. As shown in Table III, although task IDs can distinguish different tasks, they lack semantic information regarding object categories and action intents. Consequently, this approach results in significantly lower success rates compared to natural language descriptions. These findings underscore the importance of semantically rich prompts for guiding token selection and enabling effective parameter shearing across tasks.

V. CONCLUSION

We propose the Tokenized Skill Scaling (T2S) framework for lifelong robot manipulation tasks, focusing on mitigating catastrophic forgetting and enabling new skill scaling. To achieve this, we transform parameter linear mapping in the traditional transformer to tokenize all parameters to tokens, which allows the model to scale easily through the extension of new tokens. Additionally, we introduce a language-guided skill scaling method to address the issue of linearly growing parameters and to facilitate knowledge sharing across tasks. Through quantitative evaluation of the LIBERO task suites, we demonstrate that our proposed method significantly outperforms baseline approaches in mitigating catastrophic forgetting.

VI. ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (Grant No. 62302167, U23A20343, 62476090, 62502159, W2521174), Science and Technology Commission of Shanghai Municipality (Grant No. 25511102700), Natural Science Foundation of Shanghai (Grant No. 25ZR1402135), Young Elite Scientists Sponsorship Program by CAST (Grant No. YESS20240780), Natural Science Foundation of Chongqing (Grant No. CSTB2023NSCQ-JQX0007, CSTB2023NSCQ-MSX0137, CSTB2025NSCQ-GPX0445), Open Project Program of the State Key Laboratory of CAD&CG (Grant No. A2501), Zhejiang University, Open Research Fund of Key Laboratory of Advanced Theory and Application in Statistics and Data Science-MOE, ECNU.

REFERENCES

- [1] S. Stepputtis, J. Campbell, M. Phielipp, S. Lee, C. Baral, and H. Ben Amor, "Language-conditioned imitation learning for robot manipulation tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 13139–13150, 2020.
- [2] A. Xie, L. Lee, T. Xiao, and C. Finn, "Decomposing the generalization gap in imitation learning for visual robotic manipulation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3153–3160, IEEE, 2024.
- [3] S. Schaal, "Learning from demonstration," *Advances in neural information processing systems*, vol. 9, 1996.
- [4] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, J. Peters, et al., "An algorithmic perspective on imitation learning," *Foundations and Trends® in Robotics*, vol. 7, no. 1-2, pp. 1–179, 2018.
- [5] N. Gavenski, O. Rodrigues, and M. Luck, "Imitation learning: A survey of learning methods, environments and metrics," *CoRR*, 2024.
- [6] J. Gong, K. Tong, Z. Chen, C. Yuan, M. Chen, Z. Zhang, X. Tan, and Y. Xie, "Human motion synthesis in 3d scenes via unified scene semantic occupancy," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.
- [7] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," *Advances in neural information processing systems*, vol. 30, 2017.
- [8] Y. Lee, S.-H. Sun, S. Somasundaram, E. S. Hu, and J. J. Lim, "Composing complex skills by learning transition policies," in *International conference on learning representations*, 2019.
- [9] J. Leike, D. Krueger, T. Everitt, M. Martic, V. Maini, and S. Legg, "Scalable agent alignment via reward modeling: a research direction," *arXiv preprint arXiv:1811.07871*, 2018.
- [10] S. Gu, L. Yang, Y. Du, G. Chen, F. Walter, J. Wang, and A. Knoll, "A review of safe reinforcement learning: Methods, theories and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- [11] J. Garcia and F. Fernández, "A comprehensive survey on safe reinforcement learning," *Journal of Machine Learning Research*, vol. 16, no. 1, pp. 1437–1480, 2015.
- [12] W. Liang, G. Sun, Q. He, Y. Ren, J. Dong, and Y. Cong, "Never-ending behavior-cloning agent for robotic manipulation," *arXiv preprint arXiv:2403.00336*, 2024.
- [13] T. Lesort, V. Lomonaco, A. Stoian, D. Maltoni, D. Filliat, and N. Díaz-Rodríguez, "Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges," *Information fusion*, vol. 58, pp. 52–68, 2020.
- [14] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of learning and motivation*, vol. 24, pp. 109–165, Elsevier, 1989.
- [15] S. Dohare, J. F. Hernandez-Garcia, Q. Lan, P. Rahman, A. R. Mahmood, and R. S. Sutton, "Loss of plasticity in deep continual learning," *Nature*, vol. 632, no. 8026, pp. 768–774, 2024.
- [16] W. Wan, Y. Zhu, R. Shah, and Y. Zhu, "Lotus: Continual imitation learning for robot manipulation through unsupervised skill discovery," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 537–544, IEEE, 2024.
- [17] K. Roy, A. Dissanayake, B. Tidd, and P. Moghadam, "M2distill: Multi-modal distillation for lifelong imitation learning," *arXiv preprint arXiv:2410.00064*, 2024.
- [18] H. Zhang, B.-B. Gao, Y. Zeng, X. Tian, X. Tan, Z. Zhang, Y. Qu, J. Liu, and Y. Xie, "Learning task-aware language-image representation for class-incremental object detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, pp. 7096–7104, 2024.
- [19] K. Huang, P. Li, J. Ma, T. Yao, and Y. Liu, "Knowledge transfer in incremental learning for multilingual neural machine translation," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15286–15304, 2023.
- [20] Z. Zheng, M. Ma, K. Wang, Z. Qin, X. Yue, and Y. You, "Preventing zero-shot transfer degradation in continual learning of vision-language models," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 19125–19136, 2023.
- [21] B. Kim, M. Seo, and J. Choi, "Online continual learning for interactive instruction following agents," *arXiv preprint arXiv:2403.07548*, 2024.
- [22] Y. Meng, Z. Bing, X. Yao, K. Chen, K. Huang, Y. Gao, F. Sun, and A. Knoll, "Preserving and combining knowledge in robotic lifelong reinforcement learning," *Nature Machine Intelligence*, pp. 1–14, 2025.
- [23] A. Chaudhry, M. Rohrbach, M. Elhoseiny, T. Ajanthan, P. K. Dokania, P. H. Torr, and M. Ranzato, "On tiny episodic memories in continual learning," *arXiv preprint arXiv:1902.10486*, 2019.
- [24] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al., "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.
- [25] Y. Zhu, P. Stone, and Y. Zhu, "Bottom-up skill discovery from unsegmented demonstrations for long-horizon robot manipulation," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4126–4133, 2022.
- [26] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.
- [27] X. Bi, D. Chen, G. Chen, S. Chen, D. Dai, C. Deng, H. Ding, K. Dong, Q. Du, Z. Fu, et al., "Deepseek llm: Scaling open-source language models with longtermism," *arXiv preprint arXiv:2401.02954*, 2024.
- [28] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.
- [29] L. Gong, D. He, Z. Li, T. Qin, L. Wang, and T. Liu, "Efficient training of bert by progressively stacking," in *International conference on machine learning*, pp. 2337–2346, PMLR, 2019.
- [30] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.
- [31] P. Wang, R. Panda, L. T. Hennigen, P. Greengard, L. Karlinsky, R. Feris, D. D. Cox, Z. Wang, and Y. Kim, "Learning to grow pretrained models for efficient transformer training," *arXiv preprint arXiv:2303.00980*, 2023.
- [32] C. Chen, Y. Yin, L. Shang, X. Jiang, Y. Qin, F. Wang, Z. Wang, X. Chen, Z. Liu, and Q. Liu, "bert2bert: Towards reusable pretrained language models," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2134–2148, 2022.
- [33] H. Wang, Y. Fan, M. F. Naeem, Y. Xian, J. E. Lenssen, L. Wang, F. Tombari, and B. Schiele, "Tokenformer: Rethinking transformer scaling with tokenized model parameters," in *Thirteenth International Conference on Learning Representations*, 2025.
- [34] M. J. Hausknecht and P. Stone, "Deep recurrent q-learning for partially observable mdps," in *AAAI fall symposia*, vol. 45, p. 141, 2015.
- [35] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone, "Liberio: Benchmarking knowledge transfer for lifelong robot learning," *Advances in Neural Information Processing Systems*, vol. 36, pp. 44776–44791, 2023.
- [36] M. Bain and C. Sammut, "A framework for behavioural cloning," in *Machine intelligence 15*, pp. 103–129, 1995.

- [37] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, *et al.*, “Lora: Low-rank adaptation of large language models.” *ICLR*, vol. 1, no. 2, p. 3, 2022.
- [38] N. Hounsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, “Parameter-efficient transfer learning for nlp,” in *International conference on machine learning*, pp. 2790–2799, PMLR, 2019.
- [39] B. Lester, R. Al-Rfou, and N. Constant, “The power of scale for parameter-efficient prompt tuning,” *arXiv preprint arXiv:2104.08691*, 2021.