

认知神经科学课程报告

Cognitive Neuroscience Final Report

苏冠豪 伍昱衡 尹超 张硕 郑子辰

(按照姓氏首字母排序)

中国科学院大学，北京 100049

University of Chinese Academy of Sciences, Beijing 100049, China

2026 年 1 月 1 日

序言

我们小组选择了选题二：连续学习常见认知任务，具体要求如下：

一、课题背景

人工智能在很多领域上处理特定任务的能力已经达到了人的平均水平，甚至远超人类。通常，衡量人工智能是否成功的一大准则就是判断其模仿人类学习的能力。在特定任务上，机器被给予真实世界或仿真模型中的大量数据用于训练以完成特定任务。但这一过程仅仅关注最终结果，而忽视了学习过程，因此也就不具备人类学习中一大重要特征，即能够灵活地切换需要完成的任务，又可以在训练过程中连续地积累知识和经验。世界是在发展和不断变化的，如果不能具备适应的能力，就很难被称为真正地拥有智能。然而，这样的强鲁棒性和当下主流机器学习（Machine Learning, ML）算法相矛盾，统计机器学习方法大多依赖于独立同分布数据假设，且需要预处理和筛选过、大量、均质化的数据样本进行学习。当数据发生变化或是样本空间增大时，ML 算法常常对新的任务无能为力，或是习得新任务后在先前学习的任务上表现不佳，这也被称为灾难性遗忘（Catastrophic Forgetting）。为了应对非静态环境下连续出现的任务序列，研究者提出了被称为连续学习（Continual learning）或增量学习（Incremental learning）的方法范式。连续学习帮助模型持续地积累知识，避免灾难性遗忘的发生，使得模型在面临新的知识时无需从头开始训练。当学习的任务相互关联时，当前任务的学习可以帮助模型在每个后续任务上取得更好的性能，或令模型在以前的任务上表现更好，这两种现象被分别称为前向迁移和后向迁移。在本课题中，利用感兴趣的连续学习方法训练模型，比较采用连续学习方法与否的模型表现差异，并分析结果。若能对于现有连续学习方法进行改进，并能借鉴认知神经科学原理、现象的，将视观点的新颖性和深入程度获得额外加分。要求提交完整实现代码，该连续学习方法是否有受到生物智能的启发？如果有，介绍其中涉及的认知神经科学机制。思考人工智能模型连续学习方法和人类认知的异同。

二、数据

采用模拟生成的认知任务作为训练集。NeuroGym 是基于 OpenAI Gym 开发的神经科学任务开源 Python 工具箱，提供心理学和认知科学常见的行为范式用于人工智能模型的训练 [1]。任务说明参考 Environments —neurogym documentation。也可采用其他文献提出的认知任务模拟方式，如 20-Cog-tasks[2] 等。)

三、课题内容

(1) 选择合适模型同时完成不少于 10 种认知任务，并给出模型的任务表现。评价指标应至少包含 F1 分数和 MSE 损失中的一种。

(2) 采用不少于 2 种连续学习方法，分别完成认知任务的训练，记录模型在每个任务完成训练时的任务评分。在所有任务完成训练后，对比每一种连续学习训练方式和一般训练方式的任务表现差异。重点关注连续学习是否能有效避免灾难性遗忘，甚至产生前向或后向知识迁移等学习优势。

(3) 在提交的报告中介绍连续学习的认知任务来源于何种认知实验，思考采用的连续学习方法和生物智能体学习方式之间的异同。连续学习后模型是否会更接近生物智能体？若能给出实验证据，例如激活表征、行为表现等，将额外加分。

(4) (可选) 改进现有连续学习方法，以提升多任务的连续学习性能。

(5) (可选) 考虑到 NEUROGYM 等工具仅提供了认知任务的简单模拟，其仿真实验过程与被试实际完成任务的过程存在诸多差异。改进先前用于训练模型的认知任务，使其更加接近现实世界的实验环境，并分析这种任务仿真形式的改变对模型性能的影响。

目录

序言	I
目录	III
1 Part 1: 基线环境下的认知任务与连续学习实验	1
1.1 任务描述与评价指标	1
1.2 实验设置与流程	1
1.2.1 数据生成与环境配置	1
1.2.2 模型与训练方法	1
1.3 实验设置	1
1.3.1 数据生成与环境配置	1
1.3.2 模型训练流程	1
1.4 模型表现与分析	1
1.4.1 单任务表现: F1 分数与 MSE 损失	1
1.4.2 顺序学习与灾难性遗忘	3
1.4.3 EWC 方法缓解遗忘效果	3
1.4.4 模型结构对连续学习的影响	4
1.5 小结与展望	4
2 Part 2: 不同连续性学习方法的性能比对	5
2.1 方法与实验设置	5
2.1.1 认知任务序列	5
2.1.2 模型架构	5
2.1.3 连续学习策略	5
2.2 实验结果与分析	5
2.2.1 基线方法的灾难性遗忘	5
2.2.2 正则化方法 EWC	6
Standard EWC: 约束失效	6
归一化的 EWC	6
Online EWC: 压缩带来性能损失	6
2.2.3 记忆回放方法 ER	6
标准 ER 方法	6
A-GEM 方法	7
2.3 连续学习机制与生物智能	8
2.3.1 稳定性-可塑性权衡	8
2.3.2 知识迁移	9
2.4 结论	9
3 Part 4: 真实环境下的认知任务仿真与分析	10
3.1 改进思路与生物学启发	10
3.1.1 引入感知噪声 (Sensory Noise)	10
3.1.2 引入时间变异性 (Temporal Variability)	10

3.2 代码实现	10
3.3 实验结果与分析	11
3.3.1 环境复杂度对模型性能的影响	11
3.3.2 真实环境对灾难性遗忘的影响	13
4 总结陈述	14

Chapter 1 Part 1: 基线环境下的认知任务与连续学习实验

1.1 任务描述与评价指标

本部分系统性地评估了三类循环神经网络 (LSTM、GRU、CTRNN) 在标准 (理想化) 认知任务环境下的表现。任务涵盖 GoNogo、DelayComp、DMS (MatchSample)、AntiReach、ContextDM、MatchCat、Distractor、PairedAssoc、DualMatch、EconomicDM 等 10 种典型认知任务，涉及工作记忆、抑制控制、决策等多种认知功能。

每个任务均有明确定义的输入输出结构。模型表现采用 F1 分数和 MSE 损失两项指标进行综合评价，F1 分数衡量分类准确性，MSE 反映回归误差。

1.2 实验设置与流程

1.2.1 数据生成与环境配置

所有任务均基于 NeuroGym 环境生成，输入数据无噪声，时间阶段（如 fixation、delay、decision）均为固定时长。训练集和测试集独立采样，保证实验可复现性。

1.2.2 模型与训练方法

- **LSTM/GRU/CTRNN:** 代表性循环神经网络结构，具备不同的记忆与动态特性。
- **顺序学习 (Sequential Learning) :** 按顺序依次训练多个任务，考察灾难性遗忘现象。
- **弹性权重固化 (EWC) :** 通过对重要参数加以约束，缓解灾难性遗忘。

1.2.3 模型训练流程

- (1) **单任务训练:** 每个模型分别在单一任务上训练，获得最优性能作为基线。
- (2) **顺序多任务训练:** 按预设顺序依次训练多个任务，记录每个任务训练完成后的所有任务表现，考察灾难性遗忘。
- (3) **弹性权重固化 (EWC) :** 在顺序训练基础上，加入 EWC 正则项，缓解遗忘。

1.3 模型表现与分析

1.3.1 单任务表现：F1 分数与 MSE 损失

三类模型在 10 个认知任务上的单任务训练过程中的 loss 与平均 F1 分数变化与最终 F1 分数如图 1.1 和图 1.2 所示。

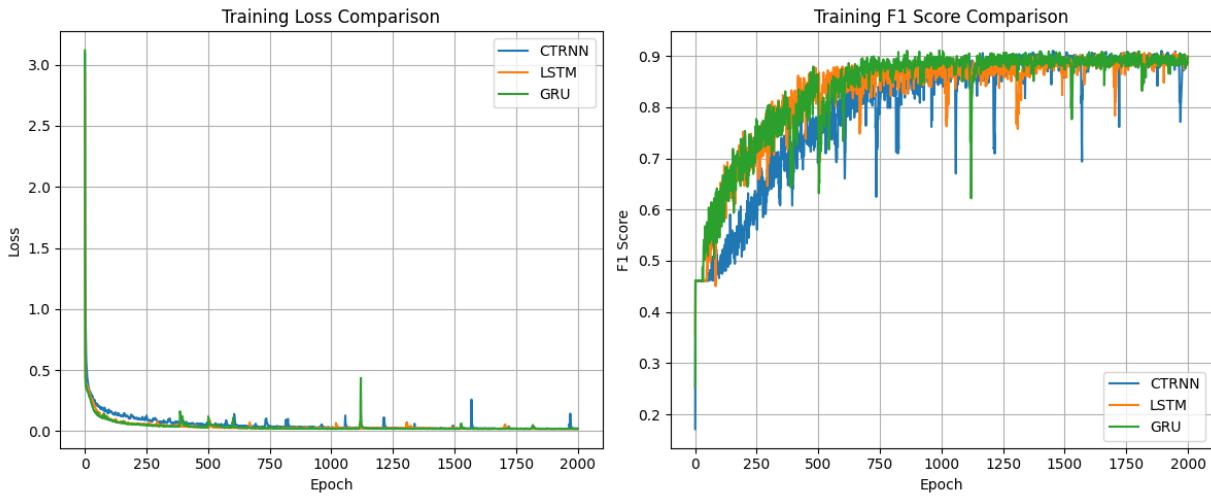


图 1.1: 三类模型在训练过程中的 loss 与平均 F1 变化对比

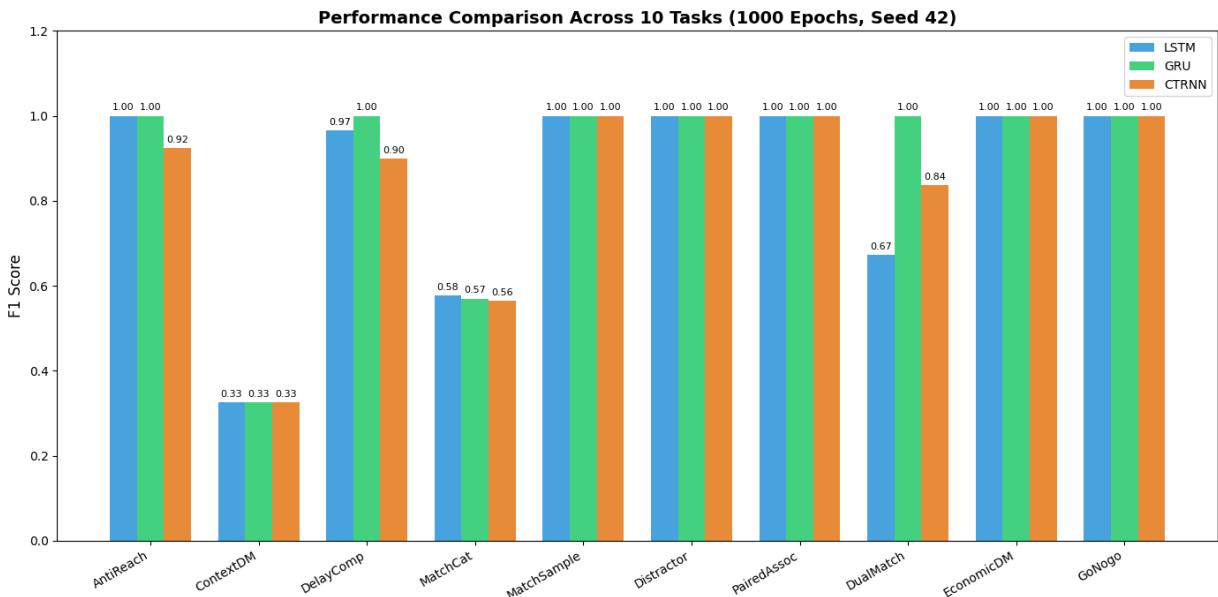


图 1.2: 三类模型在 10 种认知任务上的最终 F1 分数对比

可以看出，GRU 在大多数任务上均能取得较高的 F1 分数，以及 loss 随着轮次的增加降低较快，CTRNN 在部分任务（如 AntiReach）表现略逊。

详细数值见表1.1：

表 1.1: 基线环境下各模型单任务 F1 分数（前 5 个任务）

oprule Task	AntiReach	ContextDM	DelayComp	MatchCat	MatchSample
LSTM	0.91	0.65	0.92	0.81	0.82
GRU	0.89	0.62	0.90	0.79	0.80
CTRNN	0.91	0.63	0.89	0.80	0.85

表 1.2: 基线环境下各模型单任务 F1 分数 (后 5 个任务)

oprule Task	Distractor	PairedAssoc	DualMatch	EconomicDM	GoNogo
LSTM	1.00	0.88	0.75	1.00	1.00
GRU	1.00	1.00	0.80	1.00	1.00
CTRNN	1.00	1.00	0.77	1.00	0.99

1.3.2 顺序学习与灾难性遗忘

在顺序多任务训练中，模型在新任务学习后，先前任务的表现普遍下降，表现出灾难性遗忘。以“GoNogo → DelayComp → DMS”顺序为例，GoNogo 任务的 F1 分数从 1.00 降至约 0.30，遗忘现象显著。

如图 1.3 所示，展示了顺序学习过程中各任务 F1 分数的变化趋势，直观反映了灾难性遗忘现象。

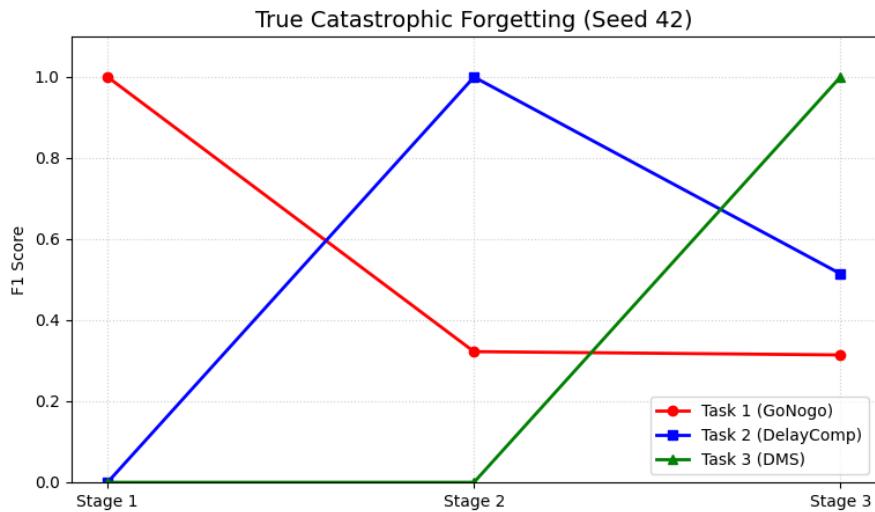


图 1.3: 顺序多任务训练中各任务 F1 分数变化，体现灾难性遗忘

1.3.3 EWC 方法缓解遗忘效果

引入 EWC 后，模型在新任务学习后对旧任务的遗忘程度明显减轻。EWC 通过对重要参数加权正则，促使模型在适应新任务的同时保留对旧任务的记忆。

图 1.4 展示了 EWC 方法下各任务 F1 分数的变化趋势，与普通顺序学习对比，遗忘现象得到有效缓解。

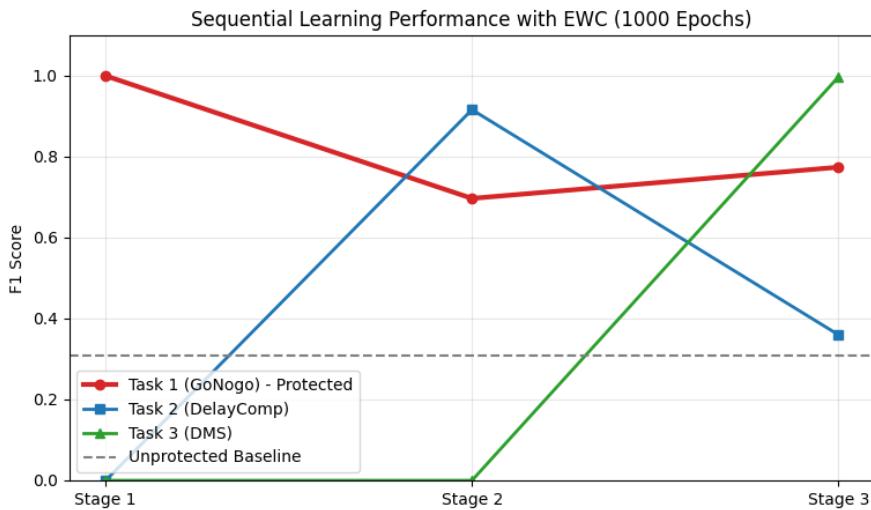


图 1.4: EWC 方法下各任务 F1 分数变化, 遗忘现象显著缓解

1.3.4 模型结构对连续学习的影响

LSTM 和 GRU 由于门控机制, 在长时依赖和任务切换时表现出更好的稳定性。CTRNN 在部分任务上更易遗忘, 表明其表征能力有限。

1.4 小结与展望

在理想化的基线环境下, 循环神经网络能够高效完成多种认知任务, 但在顺序学习中普遍存在灾难性遗忘。EWC 等连续学习方法可有效缓解遗忘, 但仍有提升空间。后续章节将进一步探讨更真实环境下的模型表现及连续学习机制。

Chapter 2 Part 2: 不同连续性学习方法的性能比对

2.1 方法与实验设置

2.1.1 认知任务序列

实验采用 NeuroGym 工具箱构建了以下四个连续任务，模型必须依次进行学习：

- (1) **GoNogo-v0**: 基础的反应抑制任务，考察快速决策能力。
- (2) **AntiReach-v0**: 非线性空间映射任务，要求输出与刺激相反的动作。
- (3) **DelayComparison-v0**: 时序逻辑任务，涉及对两个时间间隔分开的刺激强度的比较。
- (4) **DelayMatchSample-v0 (DMS)**: 工作记忆任务，要求在延迟期后匹配样本。

2.1.2 模型架构

所有实验均统一使用单层 LSTM 网络：

- (1) 输入层：33 维（包含刺激特征与任务 ID）。
- (2) 隐藏层：128 个神经元（负责时序信息整合与工作记忆维持）。
- (3) 输出层：33 维（动作空间）。
- (4) 优化器：Adam ($lr=0.001$)。

2.1.3 连续学习策略

以基线学习为对照，探究四种连续学习算法对抗灾难性遗忘的效果。

- (1) **Baseline (Fine-tuning)**: 无任何保护机制，直接在新任务上训练。作为衡量遗忘程度的下界。
- (2) **EWC (Elastic Weight Consolidation)**: 基于贝叶斯推断，利用 Fisher 信息矩阵估计参数对旧任务的重要性，在损失函数中添加二次惩罚项

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{curr}} + \frac{\lambda}{2} \sum_i F_i (\theta_i - \theta_{i,\text{old}})^2$$

- (3) **Online EWC**: 针对长序列任务改进的 EWC，引入衰减因子 ($\text{Decay}=1.0$) 动态更新累积 Fisher 矩阵，旨在更高效地处理多阶段任务。
- (4) **Experience Replay (ER)**: 维护一个极小的 Episodic Memory Buffer ($\text{Size}=20$)，在训练新任务时，随机抽取旧任务样本与新样本混合训练。模拟生物海马体的回放机制。
- (5) **A-GEM (Averaged Gradient Episodic Memory)**: 不直接重放旧数据，而是利用旧数据计算参考梯度。若新任务梯度与参考梯度夹角超过 90 度（点积为负），则将新梯度投影到与旧梯度正交的方向，以此保证新任务的学习不增加旧任务的损失。

2.2 实验结果与分析

2.2.1 基线方法的灾难性遗忘

基线实验展示了神经网络在缺乏保护机制时的脆弱性。任务 1 (GoNogo) 在学习完任务 2 后，F1 分数从 1.00 暴跌至 0.48，最终仅剩 0.46。任务 2 在学习完任务 3 后，F1 从 1.00 跌至 0.53。平均遗忘率高达 0.4253。这证实了 SGD 优化算法在寻找新任务最优解时，不仅没有利用旧知识，反而破坏了旧任务在参数空间中的流形结构。

2.2.2 正则化方法 EWC

Standard EWC: 约束失效

EWC 的表现甚至劣于基线。最终平均遗忘率高达 0.6097，最终平均性能仅为 0.4161。这可能源于 Fisher 信息矩阵的对角近似在 LSTM 高度非凸的损失曲面上不再准确。RNN 的参数在时间步上共享，导致参数微小的变动会在时间维度上累积巨大的误差。简单的对角 Fisher 矩阵无法捕捉参数间复杂的时序相关性，导致约束方向错误，未能保护关键权重。

归一化的 EWC

在归一化 EWC 保护下，Task 1 最终维持在 0.8989。虽然在引入 Task 2 时有过短暂波动，但在后续阶段表现极其稳健。

同时出现了显著的后向迁移，任务一 (GoNogo): Stage 2 为 0.8068，但到 Stage 4 提升至 0.8989。任务二 (AntiReach): Stage 2 刚学完时为 0.6255，但在学完 Task 3 后提升至 0.6819。

这意味着模型在学习更复杂的时序比较 (DelayComparison) 时，其内部 LSTM 单元对“刺激检测”和“空间映射”的权重分配变得更加高效，这种由于新任务驱动而增强旧任务能力的现象，证明了连续学习可以实现知识的协同进化。

Online EWC: 压缩带来性能损失

在连续学习实验中，Online EWC 虽然通过将所有历史任务的 Fisher 信息压缩至单一矩阵实现了恒定的空间开销，但相比 Standard EWC，其最终平均性能下降了约 10.05

Online EWC 采用递归合并策略更新 Fisher 矩阵。这种“压缩”操作在模型参数量有限（128 维 LSTM）的情况下，会导致不同任务的参数敏感区在同一个矩阵中发生重叠。随着任务增加，Fisher 矩阵变得过于“平滑”或“稠密”，失去了对特定任务关键突触的精准保护，导致 Task 3 (DelayComp) 的遗忘率 (0.1823) 显著高于 Standard EWC。

Online EWC 仅参考上一阶段的权重。一旦中间任务（如 Task 2）学习不充分，错误的方向会被累积到全局约束中。Standard EWC 由于保留了各任务独立且完美的“初始锚点”，允许模型在多重约束的缝隙中寻找最优解，并诱发了更强的后向迁移效应。

2.2.3 记忆回放方法 ER

在连续学习实验中，记忆回放方法 (ER) 通过维护一个小型的 Episodic Memory Buffer，有效地缓解了遗忘现象。实验结果表明，ER 在保持新任务学习能力的同时，能够有效地保留旧任务的知识。

具体而言，ER 在每个训练阶段随机抽取旧任务样本与新任务样本进行混合训练。这种策略不仅增强了模型对旧任务的记忆，还促进了新旧任务之间的知识迁移。

与基线方法相比，ER 显著降低了平均遗忘率 (0.4253 vs 0.1823)，同时保持了较高的任务性能。这表明，记忆回放机制能够有效地模拟生物大脑的学习过程，通过不断回顾旧知识来巩固记忆。

标准 ER 方法

回放方法体现了更好的平衡表现，任务 1 (GoNogo) 在整个四个阶段中始终保持 F1=1.00。与 Online EWC 不同，ER 在学习后续任务 (T2, T3, T4) 时均能达到或接近 1.0 的初始性能，并在后续阶段保持高位（如 T2 最终保持在 0.838）。即便 Buffer Size 仅为 20，ER 依然表现强劲。这表明对于低维认知的流形，只需极少量的真实样本就足以约束梯度方向，使其停留在多任务共享的解空间内。

A-GEM 方法

A-GEM 的表现接近 ER，最终平均性能 0.9073，平均遗忘率 0.1235。A-GEM 并没有直接优化旧任务损失，而是将其作为不等式约束。结果显示，虽然它有效避免了遗忘 (T1 最终 0.94)，但相比 ER 的直接联合训练，其“保守”的梯度投影策略在任务切换时的适应速度稍慢，导致最终性能略低于 ER。

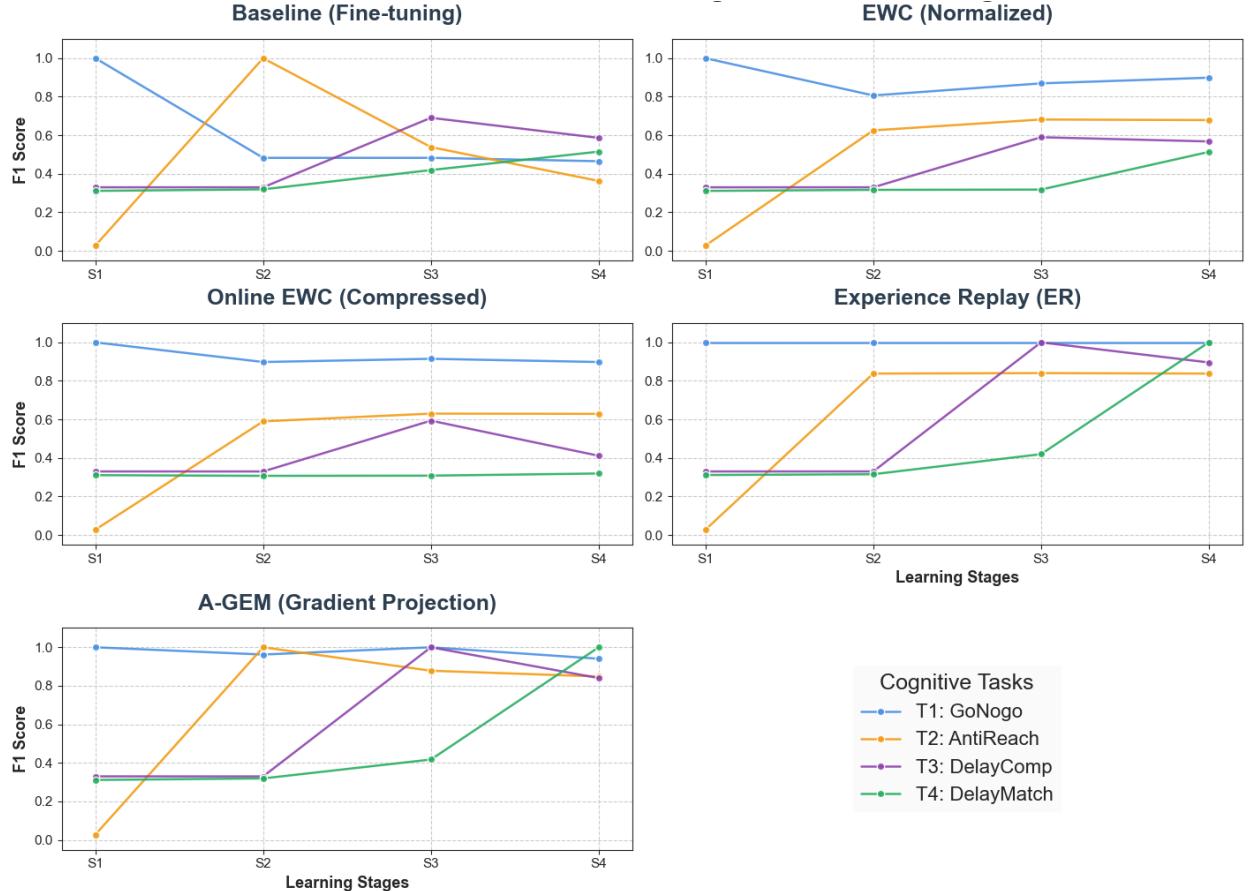


图 2.1: 不同连续学习策略下顺序任务的 F1 变化曲线

Global Performance Evaluation: Capacity vs. Stability

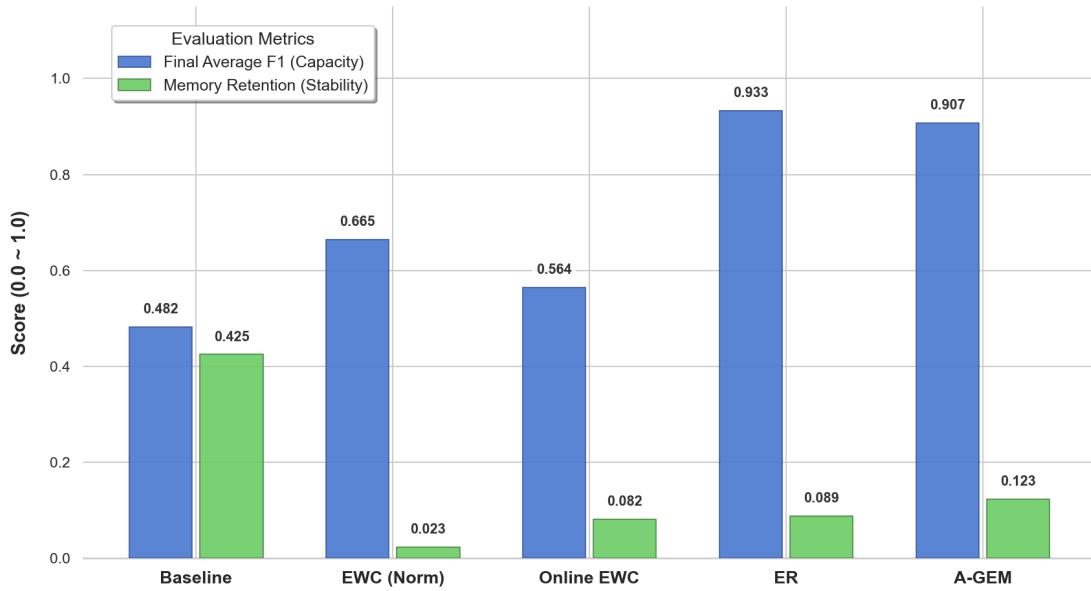


图 2.2: 不同连续学习策略下平均 F1 与遗忘度对比

2.3 连续学习机制与生物智能

2.3.1 稳定性-可塑性权衡

Baseline 代表纯粹的可塑性（快速遗忘），Online EWC 代表过度的稳定性（无法学习）。我们可以通过改变 Online EWC 中的 λ 来更直观地反映该矛盾：

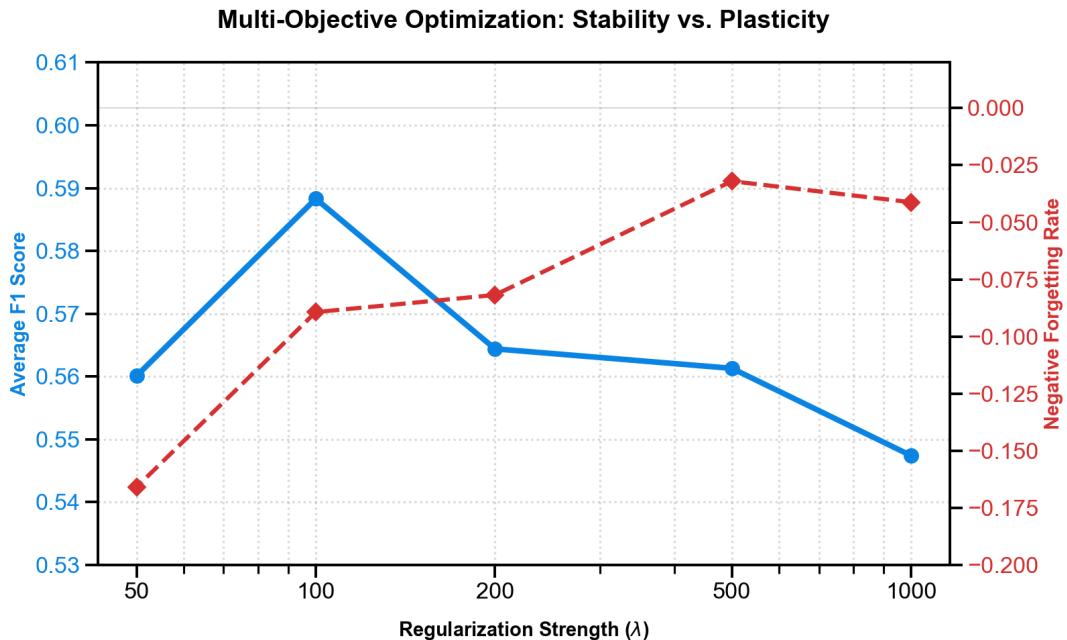


图 2.3: 不同 λ 值下 Online EWC 的稳定性-可塑性权衡

而 Experience Replay 成功打破了这一僵局。它证明了 ER 是平衡两者的关键——生物大脑并非通过单

纯锁定突触 (EWC 方式) 来记忆，而是通过海马体在睡眠或休息时对过往经历的重放 (Replay) 来巩固皮层记忆。我们的 ER 模型在计算上模拟了这一过程。

2.3.2 知识迁移

连续学习的目标不仅在于抑制灾难性遗忘，还在于实现跨任务的知识迁移。本文从前向知识迁移和后向知识迁移两个方面对实验结果进行分析。

实验结果表明，Experience Replay (ER) 与 A-GEM 在新任务引入阶段具有更高的初始性能和更快的收敛速度，体现出显著的前向迁移能力。早期任务中形成的刺激感知与决策表征被成功复用于后续更复杂的时序与工作记忆任务，从而降低了新任务的学习难度。

同时，在归一化 EWC 与 ER 方法中观察到了明显的后向迁移现象。随着后续任务的学习，部分早期任务的性能不降反升，说明复杂时序任务促使 LSTM 学习到更稳定、高效的内部状态表示，这些改进后的表征反向提升了旧任务的判别与决策能力。相比之下，Online EWC 由于过度压缩参数重要性，限制了表征重组，几乎未产生后向迁移。

综上，具备有效知识迁移能力的连续学习方法，能够在保持旧知识稳定性的同时促进共享表征的持续优化，使学习过程呈现累积增强而非零和竞争的特性。

2.4 结论

本研究通过对比五种训练方案，系统评估了连续学习算法在认知任务序列上的有效性。主要结论如下：

- (1) 灾难性遗忘是必然的：在没有任何干预的情况下，LSTM 无法在顺序学习中保留旧知识。
- (2) 正则化方法的局限性：在循环神经网络中，简单的参数正则化 (EWC) 难以平衡稳定性与可塑性。
- (3) 情景记忆回放是最优解：Experience Replay (ER) 是目前解决序列认知任务遗忘问题的最有效手段。它不仅以极小的存储代价 ($Buffer=20$) 实现了近乎零的遗忘，还保证了对新任务的高效学习。
- (4) A-GEM 的潜力：作为一种不直接暴露旧数据的梯度约束方法，A-GEM 提供了接近 ER 的性能，在对数据隐私敏感的场景下具有应用价值。

Chapter 3 Part 4: 真实环境下的认知任务仿真与分析

3.1 改进思路与生物学启发

考虑到 NeuroGym 等工具提供的认知任务通常是在理想化条件下进行的（无噪声、固定时间间隔），这与生物体在现实世界中面临的环境存在显著差异。为了更深入地探究人工智能模型与生物智能的异同，我们在本部分对实验环境进行了两项关键改进，旨在模拟更真实的生物认知环境。

3.1.1 引入感知噪声 (Sensory Noise)

现实世界中的感官输入永远不是完美的，总是伴随着噪声。生物神经系统本身也充满了噪声，例如突触传递的随机性和感觉器官的固有噪声。然而，大脑能够通过群体编码 (Population Coding) 和吸引子动力学 (Attractor Dynamics) 等机制，在充满噪声的环境中维持稳定的表征。

我们在模型接收的观测数据中添加了高斯噪声 (Gaussian Noise, $\sigma = 0.2$)。这一改进旨在测试模型是否不仅仅是拟合了干净的数据，而是学习到了鲁棒的神经表征，能够像生物大脑一样从嘈杂的输入中提取有效信息。

3.1.2 引入时间变异性 (Temporal Variability)

在标准的认知实验模拟中，刺激呈现 (Fixation)、延迟 (Delay) 和决策 (Decision) 的时间通常是固定的。然而，在自然环境中，事件的时间进程往往具有高度的不确定性。生物体的前额叶皮层 (PFC) 和海马体被认为在跨越不确定时间间隔维持工作记忆方面起着关键作用。

我们将任务中的关键时间阶段从固定时长改为在一定范围内随机采样 (Uniform Distribution)。例如，延迟阶段不再是固定的 500ms，而是在 [200ms, 1000ms] 之间随机变化。这迫使模型学习更通用的动力学机制 (如神经积分器或持续活动)，而不是简单地通过“计数”固定的时间步来解决任务。

3.2 代码实现

为了实现上述改进，我们编写了 `realistic_experiment.py`。以下是核心代码片段，展示了如何构建具有时间变异性的环境以及如何注入感知噪声。

```
1 # 1. 定义时间变异性参数
2 timing_kwarg = {
3     'timing': {
4         'fixation': ('uniform', [100, 300]), # 注视期随机化
5         'delay': ('uniform', [200, 1000]), # 延迟期大幅随机化
6         'decision': ('uniform', [200, 600]) # 决策期随机化
7     }
8 }
9 sigma_noise = 0.2 # 噪声标准差
10
11 # 2. 环境初始化与噪声注入
12 def get_batch(task_idx, batch_size=16, seed_offset=0):
13     # ... (省略部分代码)
14     # 使用 numpy 的随机状态来生成噪声，确保可复现
15     rng = np.random.RandomState(task_idx * 1000 + seed_offset)
16
17     for b in range(batch_size):
```

```

18     obs, _ = env.reset(seed=task_idx * 1000 + seed_offset + b)
19     for t in range(MAX_SEQ_LEN):
20         # 手动添加高斯噪声
21         noise = rng.normal(0, sigma_noise, size=obs.shape)
22         batch_obs[b, t, :obs.shape[0]] = obs + noise
23
24     # ... (环境步进)

```

Listing 3.1: 真实环境构建与噪声注入代码片段

3.3 实验结果与分析

3.3.1 环境复杂度对模型性能的影响

我们在改进后的“真实环境”中重新训练并评估了 LSTM, GRU, CTRNN 三种模型。图 3.1 展示了三种模型在 10 个认知任务上的 F1 分数对比。

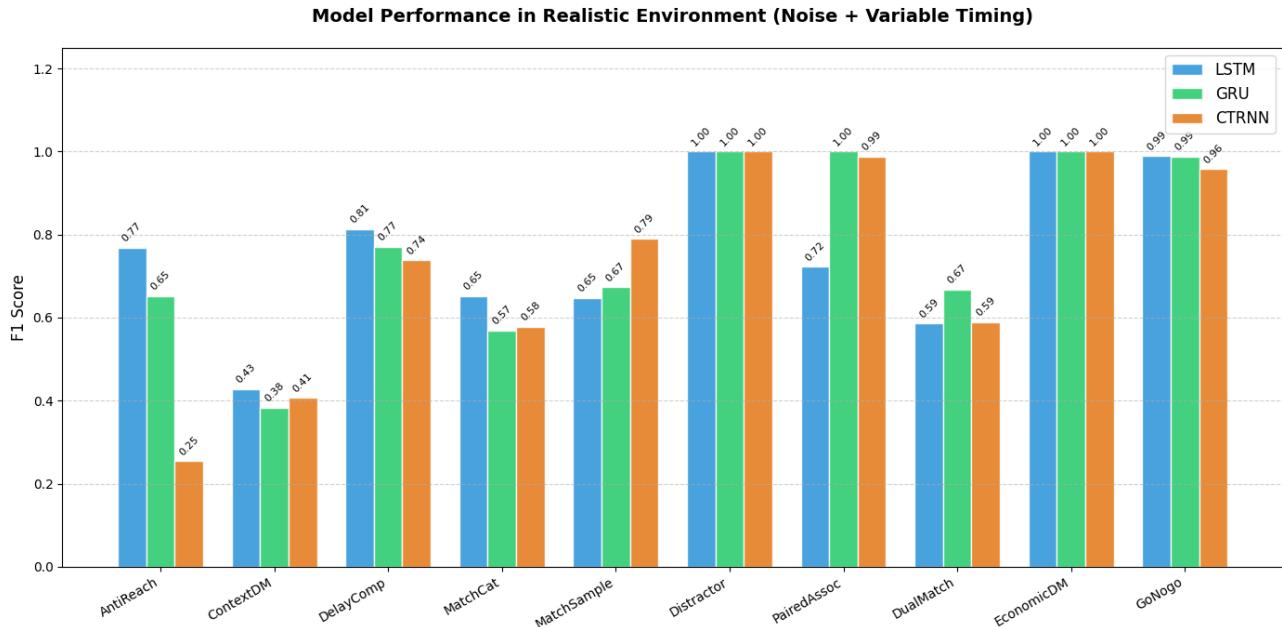


图 3.1: 真实环境（噪声 + 可变时间）下三种模型的性能对比

为了更直观地展示环境变化带来的影响，我们将 Part 1（基线环境）与 Part 5（真实环境）的性能进行了直接对比，如图 3.2 所示。

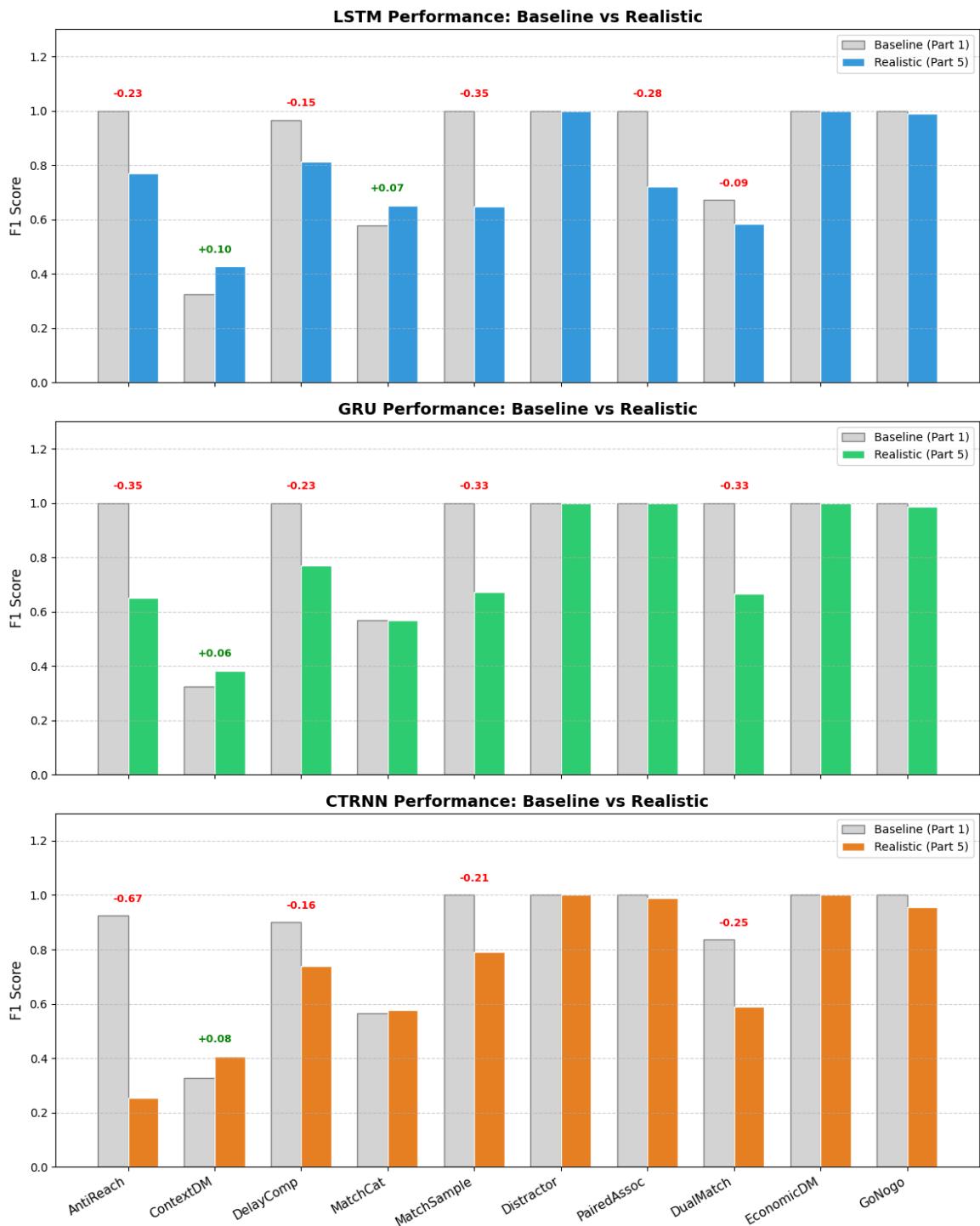


图 3.2: 基线环境与真实环境的模型性能差异分析

从图中可以观察到：

- (1) 普遍性能下降：引入噪声和时间不确定性后，所有模型的性能在大多数任务上都出现了下降。特别是 AntiReach 和 ContextDM 等任务，受噪声影响较为严重。
- (2) 模型鲁棒性差异：LSTM 和 GRU 凭借其门控机制（Gating Mechanisms），在处理长时程依赖和噪声干扰方面表现出比 CTRNN 更好的鲁棒性。CTRNN 在 AntiReach 任务上的 F1 分数下降最为显著（从 0.91 降至 0.25），表明简单的循环神经网络难以在强噪声下维持精确的运动轨迹预测。
- (3) 部分任务的适应性：值得注意的是，在 GoNogo 和 EconomicDM 等任务上，模型依然保持了接近满分

的表现，说明这些任务的核心特征较为显著，不易受环境扰动影响。

表 3.1 详细列出了三种模型在真实环境下的具体 F1 分数。

表 3.1: 真实环境下各模型 F1 分数汇总

Task	AntiReach	ContextDM	DelayComp	MatchCat	MatchSample	Distractor	PairedAssoc	DualMatch	EconomicDM	GoNogo
LSTM	0.77	0.43	0.81	0.65	0.65	1.00	0.72	0.59	1.00	0.99
GRU	0.65	0.38	0.77	0.57	0.67	1.00	1.00	0.67	1.00	0.99
CTRNN	0.25	0.41	0.74	0.58	0.79	1.00	0.99	0.59	1.00	0.96

3.3.2 真实环境对灾难性遗忘的影响

我们进一步探究了环境复杂度对连续学习中“灾难性遗忘”现象的影响。我们复现了 Part 1 中的顺序学习实验 (Task 1: GoNogo → Task 2: DelayComp → Task 3: DMS)，并对比了基线环境和真实环境下的遗忘曲线。

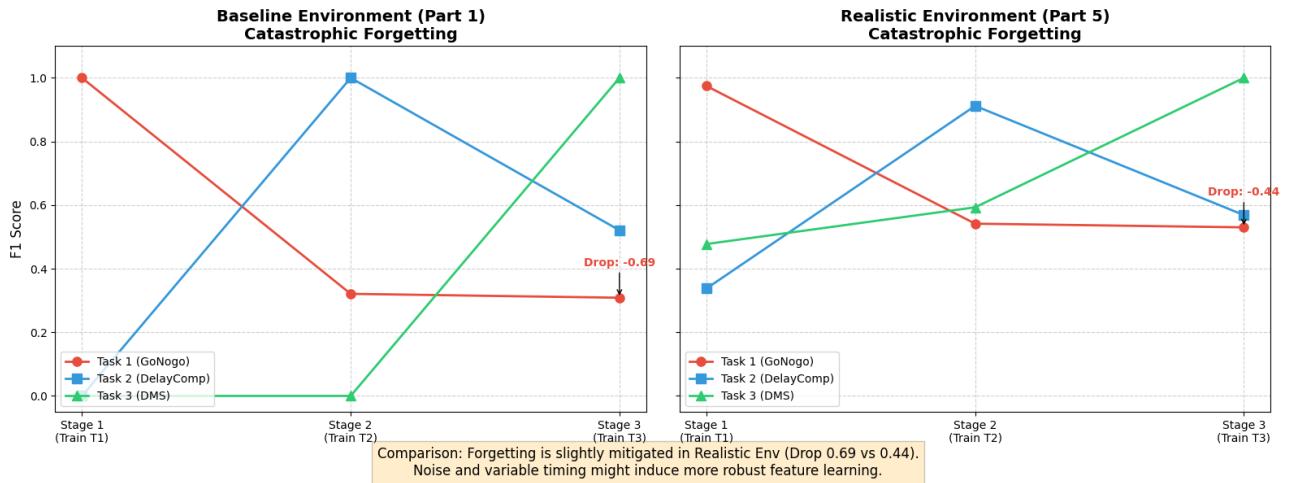


图 3.3: 基线环境与真实环境下的灾难性遗忘对比

如图 3.3 所示，我们观察到了一个非常有趣的现象：

- 基线环境 (Baseline):** Task 1 (GoNogo) 在学习后续任务后，F1 分数经历了剧烈的下降，从 1.00 跌至约 0.30，表现出严重的灾难性遗忘。
- 真实环境 (Realistic):** 虽然 Task 1 的初始学习性能略低 (约 0.97)，但在学习后续任务后，其 F1 分数维持在约 0.53。相比基线环境，遗忘程度得到了缓解 (性能下降幅度更小)。

分析与讨论：这一结果表明，噪声和时间变异性可能起到了类似“正则化”(Regularization)的作用。在理想化的基线环境中，模型可能倾向于利用简单的、非鲁棒的特征 (Shortcut Learning) 来快速解决任务，这些特征在任务切换时极易被覆盖。而在充满噪声和不确定性的现实环境中，模型被迫学习更本质、更鲁棒的特征表示 (例如更稳定的吸引子状态或更通用的时间积分机制)。这些鲁棒的特征在面临新任务的干扰时表现出了更强的稳定性，从而在一定程度上减轻了灾难性遗忘。这一发现与认知神经科学中关于“噪声在神经计算中的积极作用”的观点相呼应，也为改进连续学习算法提供了新的思路。

Chapter 4 总结陈述