Катастрофическое забывание в импульсных нейронных сетях

Денис Ларионов

Kaspersky Neuromorphic AI Conference 2025
13 ноября 2025

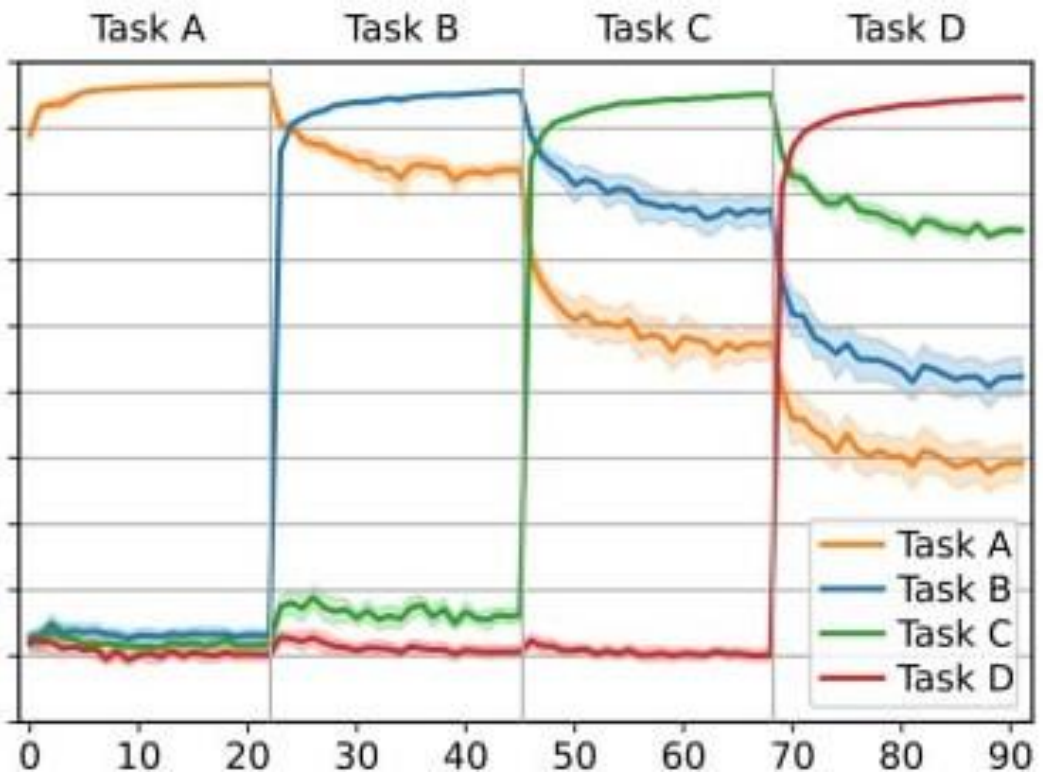
Непрерывное обучение (continual learning, CL)

Обучение нескольким задачам последовательно, как если бы все задачи были представлены одновременно.

В машинном обучении делается допущение о стационарности распределений данных. Адаптация к новому распределению данных приводит к ухудшению способности работать с предыдущими задачами – **катастрофическому забыванию** (catastrophic forgetting, CF).

Проблематика CL:

- Баланс пластичности обучения и стабильности памяти;
- Возможность различать отличия в распределениях данных как внутри, так и между задачами;
- Оптимизация использования вычислительных ресурсов.



Continual Learning with Columnar Spiking Neural Networks (Denis Larionov, Nikolay Bazenkov, Mikhail Kiselev, 2025)

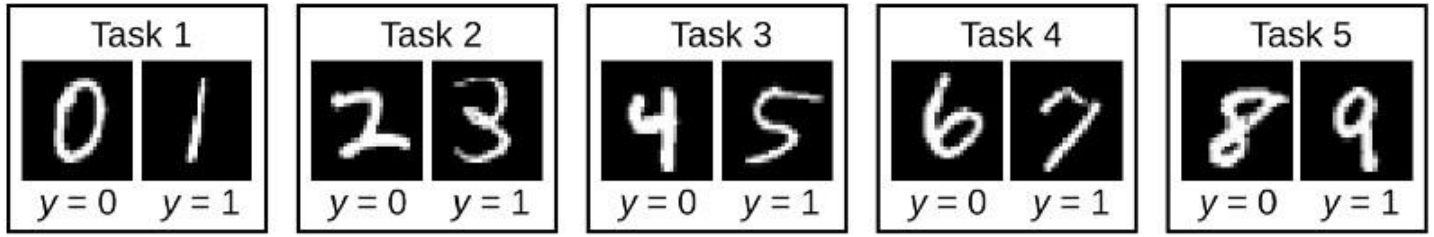
Типовые сценарии CL

Виды задач

- **task-based / task-free** - заданы ли границы задач;
- **task- / domain- / class-incremental** - доступны ли идентификаторы задач в тестировании, и если нет, должны ли они определяться;
- **streaming** - в один момент времени предъявляется только один пример;
- **online** - один пример предъявляется только один раз.

На примере splitMNSIT

- В task-incremental learning (**TIL**) нужно выбрать одну из двух цифр в известной задаче.
- В domain-incremental learning (**DIL**), также нужно выбрать один из двух вариантов, однако идентификатор задачи неизвестен (определить четное число или нет).
- В class-incremental learning (**CIL**) всегда предсказывается одна из 10 цифр.



Three types of incremental learning (van de Ven et al., 2022)

Фундаментальная причина забывания – градиентные методы обучения

Практически все успешные стратегии непрерывного обучения, представленные в литературе, предполагают использование градиентных методов оптимизации.

Использование градиентных методов является фундаментальной причиной катастрофического забывания.

Решение проблемы по существу может лежать в плоскости локальных методов обучения, которые позволяют сохранять и использовать информацию, связанную с локальными группами параметров, не используя всю сеть целиком.

Локальное обучение – изменение веса синапса зависит от состояния и активности только связанных с данным синапсом нейронов.

Локальность вычислений - подход к реализации эффективных распределенных вычислительных систем, основанных только на локальных сигналах от соседних физических узлов сети.

Локальность обучения позволяет обеспечить локальность вычислений, что принципиально невозможно при использовании градиентных методов обучения.

SoftHebb: Bayesian inference in unsupervised Hebbian soft winner-take-all networks (Moraitis et al., 2021)

Импульсные нейронные сети

Математической моделью, которая позволяет использовать STDP, конкуренцию и модуляцию обучения, являются импульсные нейронные сети (ИмНС).

ИмНС уже исследовались в контексте проблемы катастрофического забывания и не показали высокой эффективности (на примере трёхслойной сверточной сети). Одного лишь локального обучения недостаточно.

Решение проблемы катастрофического забывания лежит на более высоком уровне абстракций, где рассматриваются не отдельные нейроны, а их группы, организованные специфическим образом (микроколонки).

Основная идея колоночной организации сети заключается в том, что элементы знаний, которые должны переиспользоваться между задачами в непрерывном обучении, должны быть представлены более функциональными структурами, чем может обеспечить один нейрон.

Поэтому, в соответствии с принципом локальности вычислений, именно микроколонки, а не отдельные нейроны, должны выступать физическим субстратом для хранения полезных знаний в условиях непрерывного обучения.

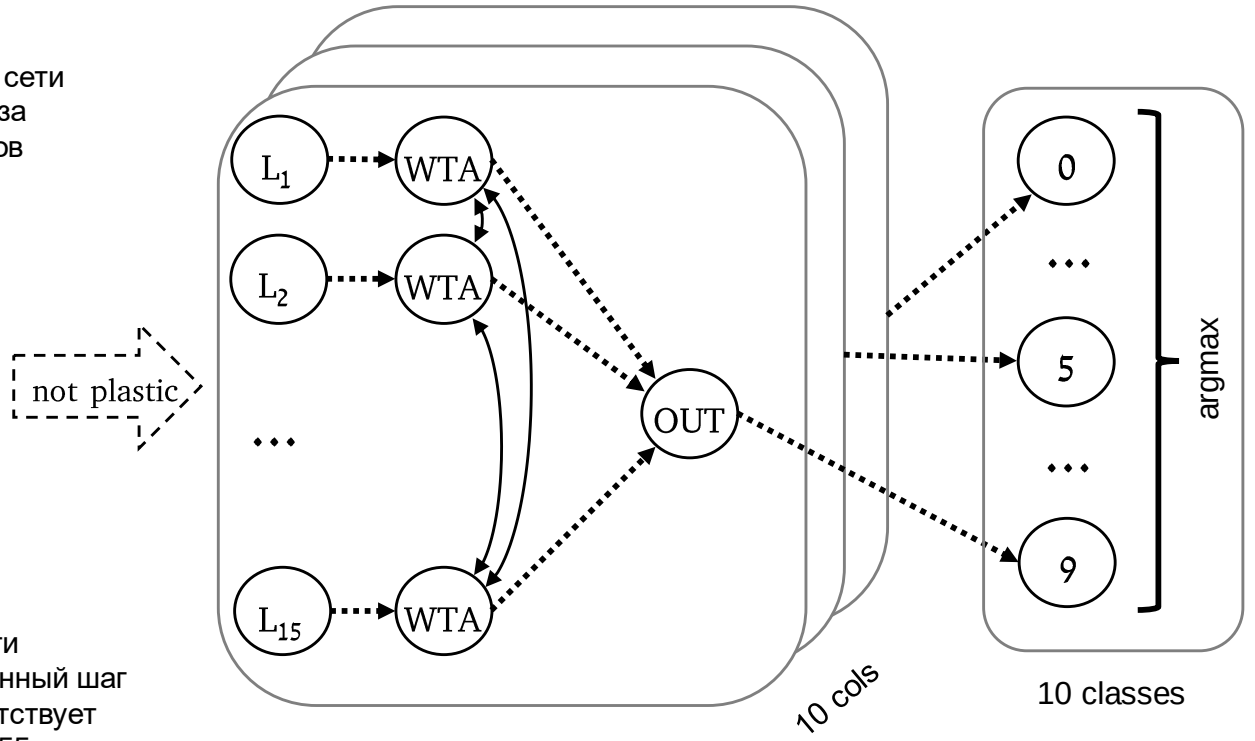
Continuous learning of spiking networks trained with local rules (Dmitry Antonov, Kirill Sviatov, Sergey Sukhov, 2022)

CoLaNET: Columnar Layered Network (inference)

Картинка представляется сети на 10 шагов вычислений, за которыми следует 10 шагов пустой картинки (silence).

Input image 28x28

Интенсивность пикселей соответствует вероятности сгенерировать спайк в данный шаг вычислений, где 0 соответствует нулевой вероятности, а 255 – максимальной (частотное кодирование).



Нейрон L, который первый сгенерировал спайк (spike), активирует свой нейрон WTA, который подавляет остальные WTA в колонке и пропускает спайк к OUT.

5

prediction

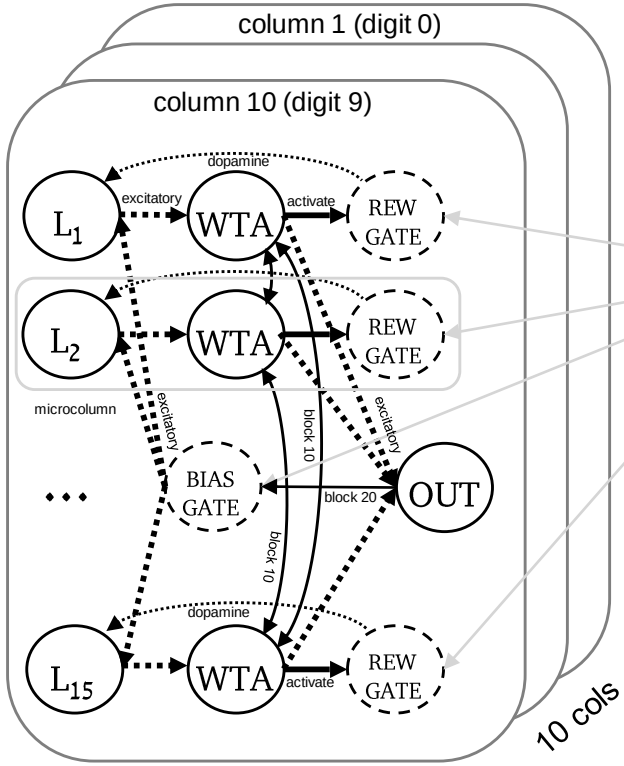
Класс определяется колонкой, которая генерировала большее число спайков за 20 шагов вычислений.

Classifying Images with CoLaNET Spiking Neural Network - the MNIST Example (Kiselev, 2024)

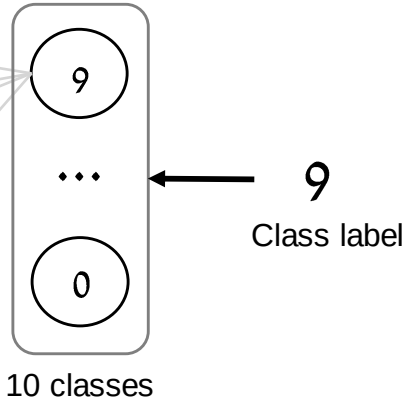
CoLaNET: Columnar Layered Network (обучение)

В начале обучения все веса нейронов L равны нулю. С ростом положительной части весов нейроны «закрепляются» на своих паттернах – сложнее активируются на случайные стимулы.

plastic



Метка предъявляется на 20 шагов. Каждый класс активирует свою колонку и подавляет остальные.



9
Class label

Обучение регулируется тремя факторами: анти-Хеббовской пластичностью (ослабление весов), дофамин-модулируемой пластичностью (усиление весов) и периодической синаптической ренормализацией.

Каждая колонка на нижних трёх слоях образована тройками нейронов L, WTA и REW GATE. Каждая такая тройка образует одну микроколонку. Одна микроколонка соответствует значительно отличающимся экземплярам (подклассам) одного класса.

Classifying Images with CoLaNET Spiking Neural Network - the MNIST Example (Kiselev, 2024)

Управление балансом пластичности обучения и стабильности памяти

Инструменты CoLaNET для непрерывного обучения

– адаптивный порог срабатывания пластичных нейронов;
– количество виртуальных синапсов, которые отвлекают на себя величину изменения веса при ренормализации;
– количество микроколонок в одной колонке.

В CoLaNET адаптивный порог срабатывания u_{tr} является суммой константной части и дополнительной добавки, которая пропорциональна сумме положительных весов:

$$u_{tr} = u_{const} + \alpha \sum_i w_i^+$$

где α - коэффициент пропорциональности, u_{const} - фиксированная часть порога (полагается равной 1), а w^+_i - положительные веса ($w_i > 0$).

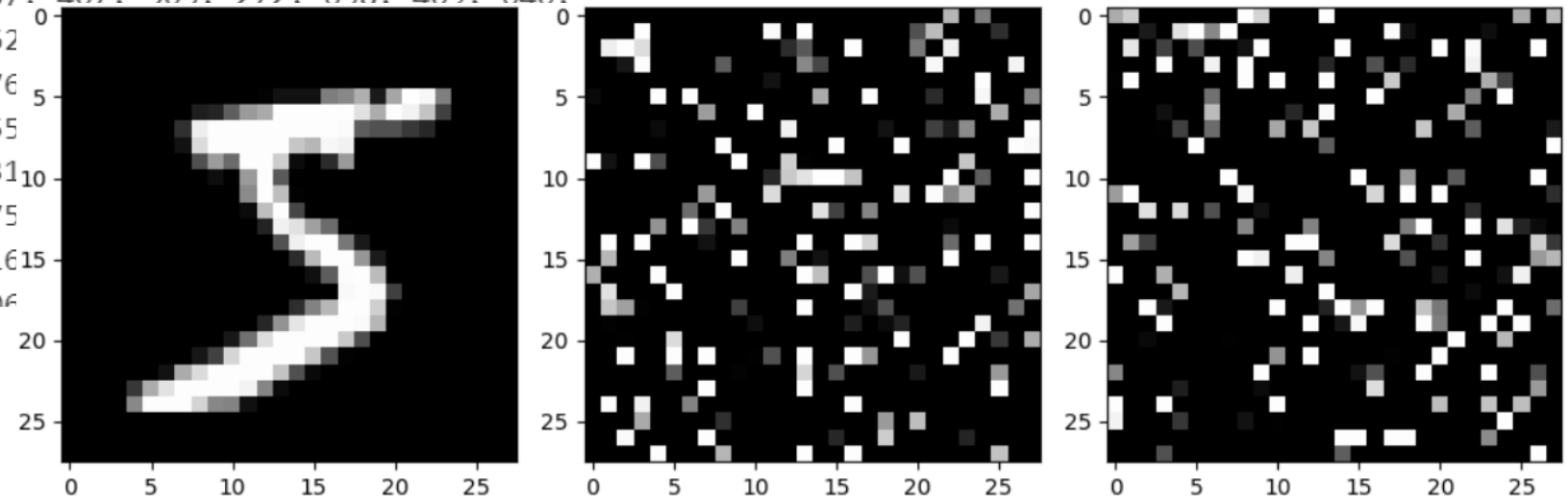
Механика адаптивного порога заставляет нейроны, у которых в процессе обучения появились большие веса, становиться менее чувствительными ко всем входным стимулам, что понижает вероятность сработать на случайный паттерн.

CoLaNET -- A Spiking Neural Network with Columnar Layered Architecture for Classification (Kiselev, 2024)

Эксперимент 1: Permuted MNSIT

```
np.random.RandomState(seed).permutation(28*28) # what is permutation
```

```
array([693,  85, 647, 392, 765,  14, 299, 711,  55,  31, 122, 172, 255,
       502, 636, 564,  75,  40, 412, 542, 710, 772, 239, 252, 156,  27,
       500, 777, 103, 365, 516, 236, 751, 529, 415, 401, 588, 230, 477,
       493, 608, 465,  79, 402, 513, 366,   8, 615, 215, 467, 251, 193,
       674, 518, 150, 609, 485, 204, 436, 572, 782,   1, 354, 471, 113,
       760, 491, 774, 434, 293, 315, 367, 407, 503, 272, 638, 409, 640,
       406, 663, 363, 316, 667, 196, 462
       351, 214,  18, 768, 458, 735, 676
       566, 162,  39, 702, 590, 283, 655
       726, 766, 142,   2, 776, 355, 231
       375, 382, 250, 756, 338, 740, 575
        37, 548,  48, 181, 492, 352, 416
       720, 210, 457, 499, 506, 268, 306
```



An empirical investigation of catastrophic forgetting in gradient-based neural networks (Goodfellow et. al., 2013)

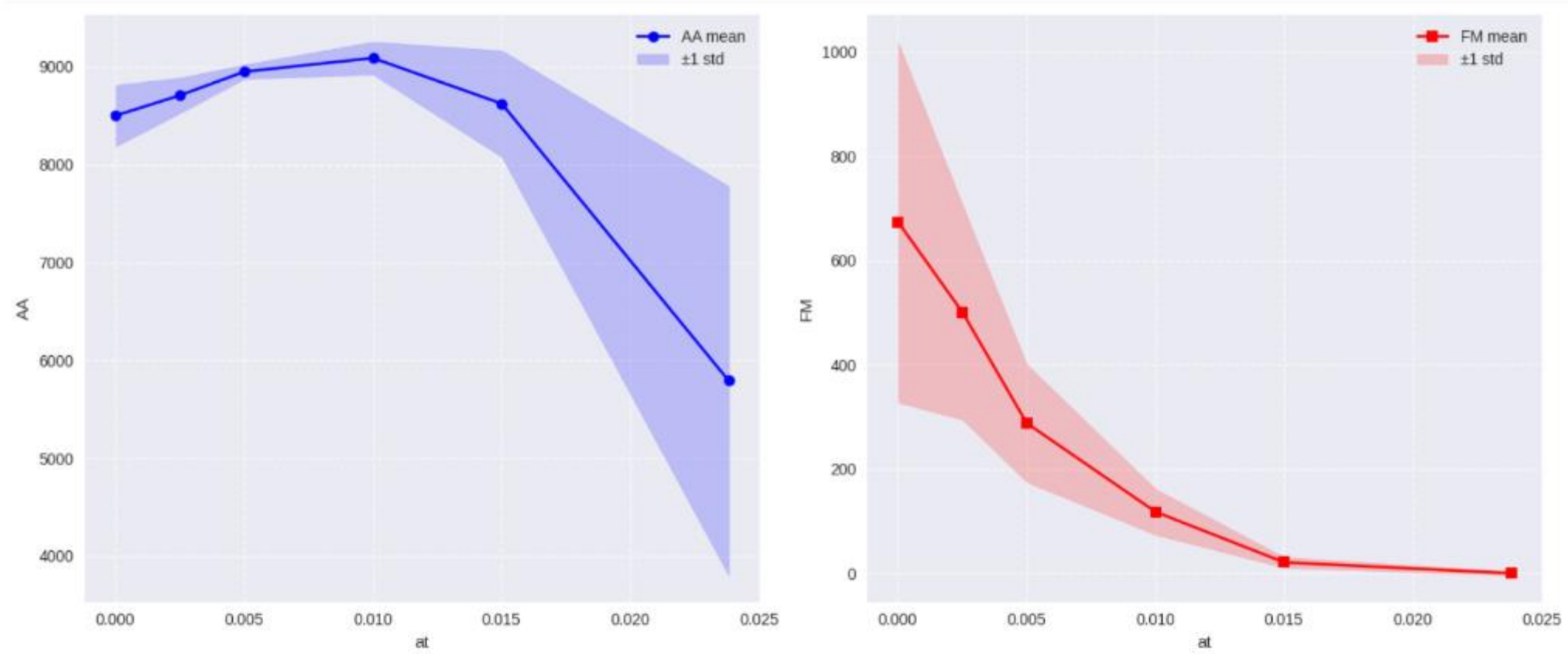
CoLaNET на Permuted MNSIT

Профиль деградации CoLaNET для 15 микроколонок и α=0.023817

Результаты свидетельствуют о смещённом балансе в сторону стабильности памяти, вместо пластичности обучения. Отсутствие забывания говорит о том, что микроколонки, которые специфицировались на одной задаче, закрепляются так сильно, что вообще перестают меняться на других задачах.

Iterations	Tasks									
	1	2	3	4	5	6	7	8	9	10
1	**94.36**									
2	94.34	**83.67**								
3	94.34	83.61	**34.17**							
4	94.34	83.61	34.68	**36.75**						
5	94.33	83.60	34.68	36.76	**25.59**					
6	94.45	83.60	34.68	36.76	25.59	**23.97**				
7	94.45	83.60	34.68	36.76	25.59	23.97	**12.01**			
8	94.43	83.60	34.68	36.76	25.59	23.97	12.03	**15.25**		
9	94.43	83.60	34.68	36.76	25.59	23.97	12.03	15.27	**11.87**	
10	94.43	83.60	34.68	36.76	25.59	23.97	12.03	15.27	11.87	**13.55**

Оптимальное значение адаптивного порога α



При значениях α выше 0.015 сеть практически перестает забывать. При значениях α от 0 до 0.01 растут как метрики точности, так и связанные с уменьшением меры забывания.

CoLaNET с оптимальным адаптивным порогом

Профиль деградации CoLaNET для 15 микроколонок и α=0.005 (был 0.023817)

Значением α, выше которого падает пластичность обучения на десяти задачах, является 0.005.

Iterations	Tasks									
	1	2	3	4	5	6	7	8	9	10
1	**91.03**									
2	89.35	**91.68**								
3	85.62	90.60	**92.29**							
4	85.13	90.54	91.90	**92.49**						
5	82.65	88.15	90.85	92.01	**93.06**					
6	80.41	87.83	90.41	91.14	92.77	**92.05**				
7	80.06	87.21	89.72	91.02	92.10	91.78	**91.94**			
8	78.85	86.55	89.29	90.58	91.08	91.48	91.89	**91.99**		
9	78.68	84.51	88.58	89.87	90.76	90.91	91.54	91.74	**91.94**	
10	78.42	84.00	88.45	89.38	90.14	90.69	91.42	91.59	91.63	**91.56**

CoLaNET на 45 микроколонках

Профиль деградации CoLaNET для 45 микроколонок и α=0.01

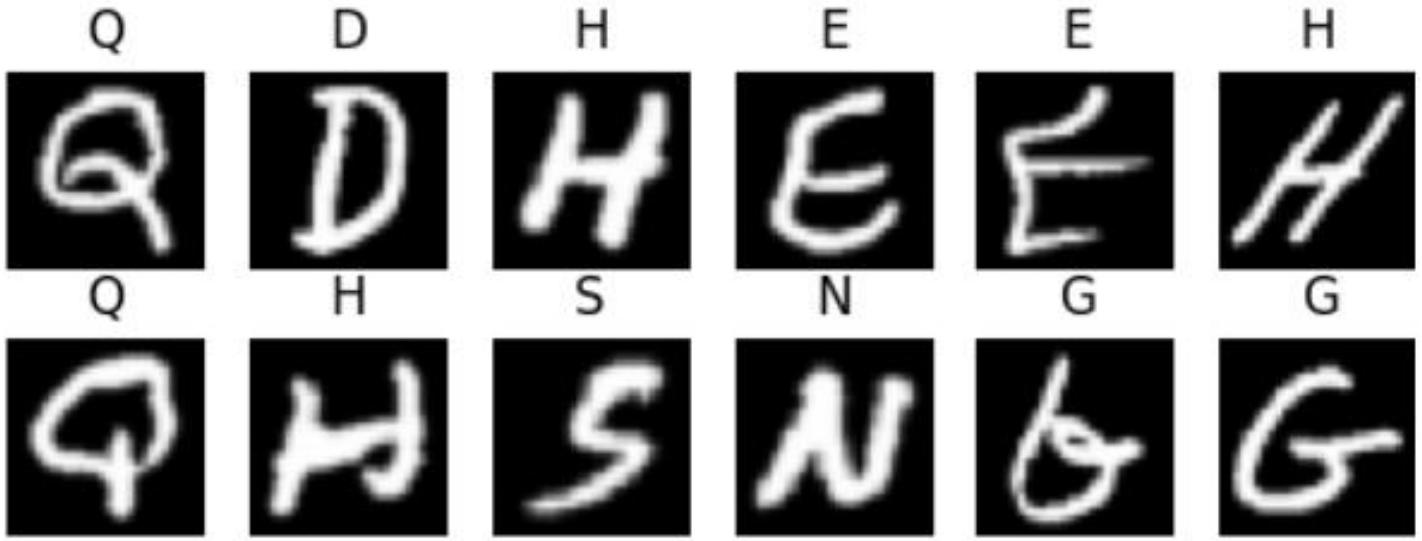
CoLaNET демонстрирует способность эффективно учиться на десяти задачах, и в то же время высокую устойчивость к забыванию, деградируя всего на 93.30 − 88.95 =4.35% на первой задаче после обучения девяти другим задачам.

Iterations	Tasks									
	1	2	3	4	5	6	7	8	9	10
1	**93.30**									
2	91.84	**93.23**								
3	91.03	92.51	**93.62**							
4	90.48	92.01	93.45	**93.87**						
5	90.19	91.86	93.02	93.48	**92.97**					
6	89.75	91.69	92.88	93.22	92.86	**93.01**				
7	89.57	91.36	92.82	92.95	92.79	92.99	**93.03**			
8	89.09	91.20	92.69	92.60	92.62	92.99	93.27	**91.73**		
9	89.00	91.19	92.46	92.57	92.47	92.92	92.96	91.69	**92.35**	
10	88.95	91.29	92.37	92.37	92.31	92.65	92.75	91.43	92.30	**91.22**

Эксперимент 2: Extended MNSIT

- Task1: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
- Task2: A, B, D, E, G, H, N, Q, R, S.

Из EMNIST в соответствии с Antonov2022 выбрано balanced разбиение, из которого используется десять классов (для соотнесения с MNIST): буквы A, B, D, E, G, H, N, Q, R, S. В каждой задаче 28 тыс. изображений.

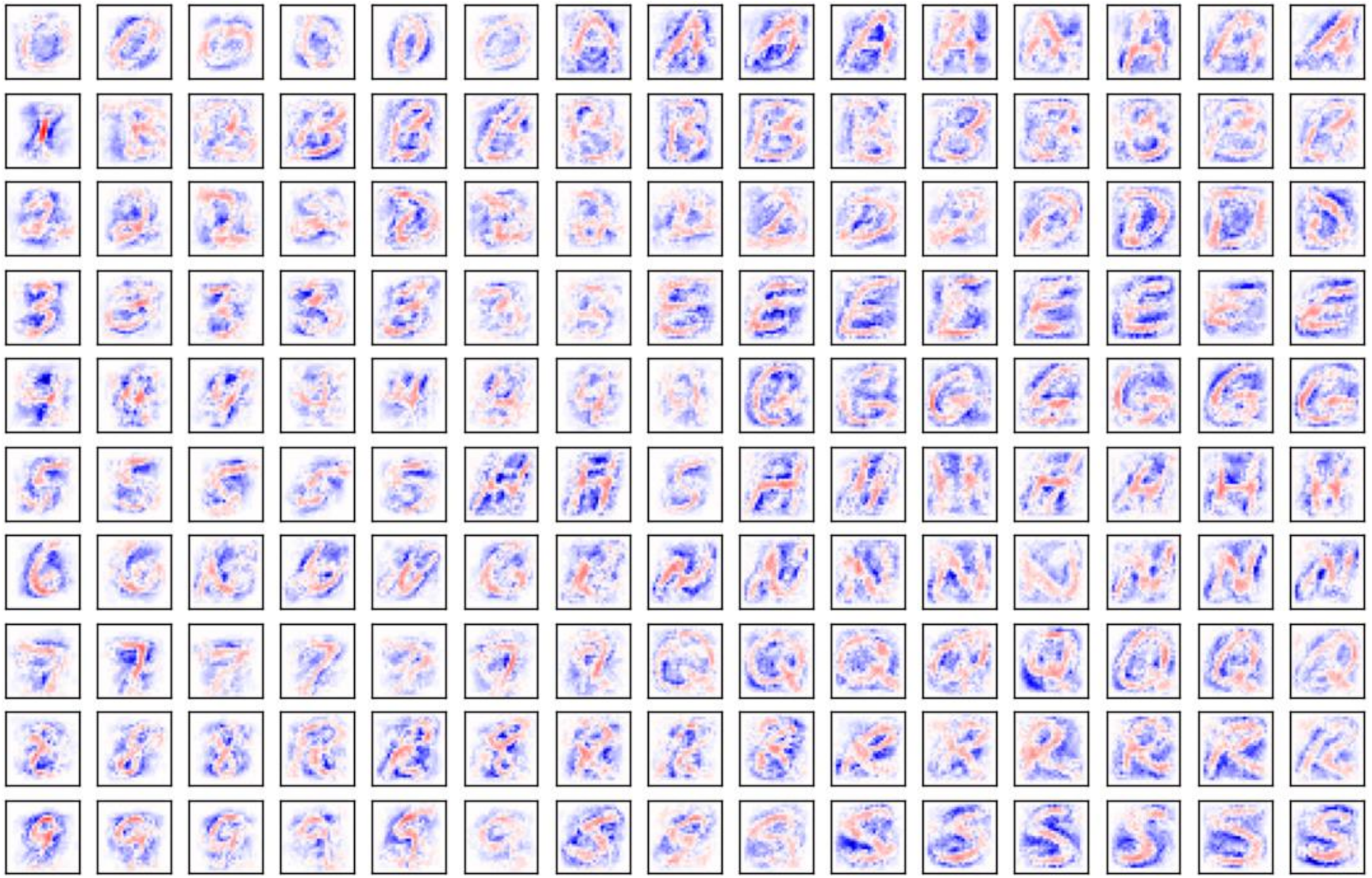


EMNIST: an extension of MNIST to handwritten letters (Cohen et. al., 2017)
Continuous learning of spiking networks trained with local rules (Dmitry Antonov, Kirill Sviatov, Sergey Sukhov, 2022)

CoLaNET на E/MNIST

Parameters α and ns	MNIST →EMNIST	MNIST	FM1	EMNIST →MNIST	EMNIST	FM2
0	88.67→82.50	25.09	63.58	79.72→88.65	19.27	60.45
0.00125	92.65→82.45	32.32	60.33	80.57→91.37	30.92	49.65
0.00125, 0	90.87→87.72	40.70	50.17	87.20→92.72	35.77	51.43
0.00125, 1k	91.87→87.50	43.40	48.47	87.25→92.70	44.80	42.45
0.00125, 10k	92.07→87.75	38.22	53.85	86.40→92.75	44.67	41.73
0.00125, 100k	91.15→87.17	40.39	50.76	86.62→93.45	44.19	42.43
0.0015	92.40→82.92	39.90	52.50	81.15→90.10	35.57	45.58
0.0015, 0	91.15→87.45	41.22	49.93	87.07→92.62	39.12	47.95
0.0015, 1k	91.97→87.90	41.82	50.15	88.22→93.17	43.92	44.30
0.0015, 10k	91.15→87.95	41.67	49.48	86.70→93.62	46.85	39.85
0.0015, 100k	92.70→88.35	42.90	49.80	88.10→93.52	48.77	39.33
0.002	92.60→81.10	49.02	43.58	80.10→89.47	38.05	42.05
0.002, 0	91.85→81.12	44.19	47.66	86.95→74.85	52.40	34.55
0.002, 1k	92.30→86.65	44.47	47.83	88.97→89.90	52.99	35.98
0.002, 10k	91.92→86.02	42.02	49.90	88.50→91.20	52.30	36.20
0.002, 100k	92.35→87.67	44.80	47.55	88.50→91.90	53.42	35.08
0.00238	92.42→78.35	56.87	35.55	78.49→86.65	38.37	40.12
0.00238, 0	91.10→65.52	52.75	38.35	83.25→29.70	55.95	27.30
0.00238, 1k	92.20→81.60	50.22	41.98	87.57→80.95	54.47	33.10
0.00238, 10k	92.47→79.64	53.02	39.45	88.07→86.70	58.02	30.05
0.00238, 100k	92.90→84.25	49.50	43.40	88.40→87.72	56.35	32.05
0.003	92.12→63.95	66.55	25.57	74.54→82.12	40.24	34.30
0.003, 0	91.72→26.55	61.00	30.72	69.17→44.40	44.07	25.10
0.003, 1k	93.35→59.77	63.72	29.63	82.20→36.65	54.72	27.48
0.003, 10k	91.60→44.19	66.87	24.73	85.92→67.62	56.47	29.45

CoLaNET после обучения на MNIST, потом на EMNIST



Сравнение результатов

10 задач Permuted MNIST

	AA	FM
Joint training	89.05 ± 0.27	
GEM [20]	74.57 ± 0.10	7.40 ± 0.11
AGEM [4]	69.50 ± 0.76	13.10 ± 0.63
MER [25]	75.75 ± 0.65	8.74 ± 0.73
MIR [1]	78.31 ± 0.63	7.15 ± 0.67
CTN [23]	79.70 ± 0.44	5.08 ± 0.44
NCC [30]	83.47 ± 0.43	3.44 ± 0.26
CoLaNET (15 microcolumns)	89.67 ± 0.16	2.75 ± 0.23
CoLaNET (45 microcolumns)	92.39 ± 0.13	0.94 ± 0.17

Сравнение эффективности разных подходов воспроизведения памяти на размере буфера 50 элементов на десяти задачах Permuted MNIST для одной эпохи обучения на каждой задаче. В нижней части приведены аналогичные результаты для CoLaNET в конфигурациях на 15 и 45 микроколонок. Результаты усреднены по десяти независимым экспериментам.

2 задачи E/MNIST

	MNIST→EMNIST	MNIST	FM
SNN [31]	90.8→78.4	48.1	42.7
Lateral inhibition	94.8→88.9	78.6	16.2
Pseudo-rehearsal	93.6→79.0	43.5	50.1
Self-reminder (0.25%)	93.6→74.0	74.5	19.1
Self-reminder (10%)	93.6→77.8	91.1	2.5
Noise regularization	93.9→87.8	67.2	26.7
Dropout	94.2→87.6	62.9	31.3
Frozen large weights	93.6→69.2	78.2	15.4
Langevin dynamics	93.6→78.3	82.2	11.4
Joint training	93.6→79.7	92.0	1.6
CoLaNET	92.9→84.3	49.5	43.4
CoLaNET + permutation	92.14→86.96	90.87 ± 0.32	1.27 ± 0.37

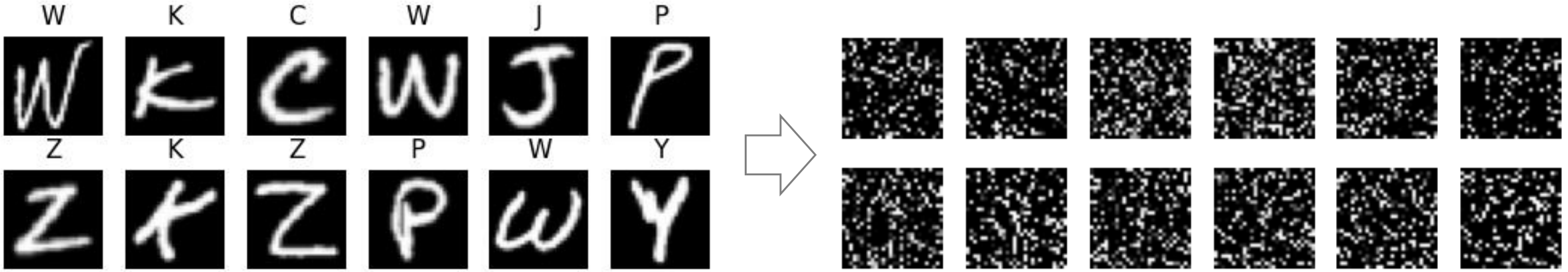
Результаты на Permuted MNIST не воспроизводятся на E/MNIST, так как в признаковом представлении задач присутствует общность. Убрать общность возможно ценой потери пространственных паттернов, применив к каждому датасету случайную перестановку (permutation).

Mitigating Forgetting in Online Continual Learning with Neuron Calibration (Yin et. al., 2021)
Continuous learning of spiking networks trained with local rules (Antonov et. al., 2022)
Continual Learning with Columnar Spiking Neural Networks (Larionov et al., 2025)

Эксперимент 3: Permuted E/MNSIT

- Task1: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
- Task2: A, B, D, E, G, H, N, Q, R, S.
- Task3: Z, X, Y, U, W, J, L, P, K, C.

К каждой задаче применяется случайная перестановка (одинаковая для всех изображений задачи). В признаковых пространствах больше нет общности, что позволяет значительно улучшить метрики CL.



	Task1	Task2	Task3
1	**93.30**		
2	91.84	**93.23**	
3	91.03	92.51	**93.62**

Контакты и материалы

tg: https://t.me/nrmairus
git: https://gitflic.ru/project/dlarionov/cl