

Отчёт сгенерирован автоматически на основе report.md.

Источник: report.md

Дата: 2025-12-15 00:10:16

Отчет по экспериментам: LeNet-5 + TensorBoard (вариант 9)

Теоретическое обоснование

- Мониторинг loss/accuracy показывает динамику качества, но не объясняет причины проблем.
- Гистограммы весов и градиентов позволяют диагностировать exploding/vanishing gradients и деградацию обучения.
- Гистограммы активаций по слоям выявляют насыщение (например, tanh уходит в ± 1) и “мертвые” слои.
- Отслеживание дельты весов помогает понять, происходит ли реальное обучение или модель “застыла”.
- ReduceLROnPlateau и LR-schedule ускоряют сходимость и повышают стабильность, early stopping предотвращает переобучение.
- Confusion matrix и примеры предсказаний дают интерпретируемый срез ошибок модели.

Постановка эксперимента

- Датасет: MNIST, $28 \times 28 \rightarrow$ padding до 32×32 , канал 1.
- Архитектура: LeNet-5 (Conv \rightarrow AvgPool \rightarrow Conv \rightarrow AvgPool \rightarrow FC \rightarrow FC \rightarrow Softmax).
- Логирование: TensorBoard (scalars + histograms) + custom callbacks (gradients, activations, deltas, confusion matrix, examples).
- Сломанные сценарии: высокий LR, плохая инициализация, отсутствие нормализации.
- Стратегии LR: fixed, step decay, cyclical (triangular).
- Профилирование: время эпохи и память при разных batch_size; CPU \leftrightarrow GPU через отдельные процессы.

Сводные результаты

- Таблица: runs_summary.csv

Сравнительный анализ

■■■ ■■■■■■■■ ■■ test_accuracy:						
run_name	lr_strategy	test_accuracy	best_val_accuracy	avg_epoch_time_sec	rss_peak_mb	
compare_cyclical	cyclical	0.9877	0.987667	3.243760	1677.035156	
compare_cyclical	cyclical	0.9877	0.987667	2.818743	1678.816406	
compare_cyclical	cyclical	0.9877	0.987667	2.874990	1676.062500	

broken_no_norm	fixed	0.9867	0.987167	1.952181	1613.105469
broken_no_norm	fixed	0.9867	0.987167	1.712022	1613.703125
broken_no_norm	fixed	0.9867	0.987167	1.843508	1614.503906
broken_no_norm	fixed	0.9867	0.987167	1.819445	1612.574219
compare_step_decay	step_decay	0.9858	0.987000	1.548318	1650.714844
compare_step_decay	step_decay	0.9858	0.987000	1.692023	1649.898438
compare_step_decay	step_decay	0.9858	0.987000	1.706978	1648.257812

Рекомендации по гиперпараметрам

- Всегда нормализуйте входные данные (например, MNIST: деление на 255). Без нормализации ухудшается сходимость и стабильность градиентов.
- Используйте устойчивые инициализации (Glorot/He). Слишком “широкая” RandomNormal может провоцировать нестабильные активации/градиенты и замедлять обучение.
- Слишком высокий learning rate часто приводит к колебаниям loss/accuracy или к NaN. Начините с 1e-3 (Adam) и включайте ReduceLROnPlateau / scheduler.
- По результатам сравнения, наиболее эффективная стратегия (test_accuracy максимум): compare_cyclical (strategy=cyclical). Рекомендуется использовать её как дефолт для похожих задач.
- Для ускорения выхода на плато используйте: ReduceLROnPlateau + сохранение best weights (ModelCheckpoint) + early stopping.
- Средняя корреляция между изменением весов и val_accuracy (по доступным прогонам): -0.827. Используйте этот сигнал как индикатор: если веса почти не меняются при низкой точности — вероятно vanishing gradients/слишком маленький LR.

Как быстро развернуть аналогичный эксперимент

- Используйте experiment_template.json из нужного run_dir как стартовую конфигурацию.
- Меняйте: lr.strategy, lr.lr/base_lr/max_lr, train.batch_size/epochs, включайте/выключайте callbacks.

Где смотреть визуализации

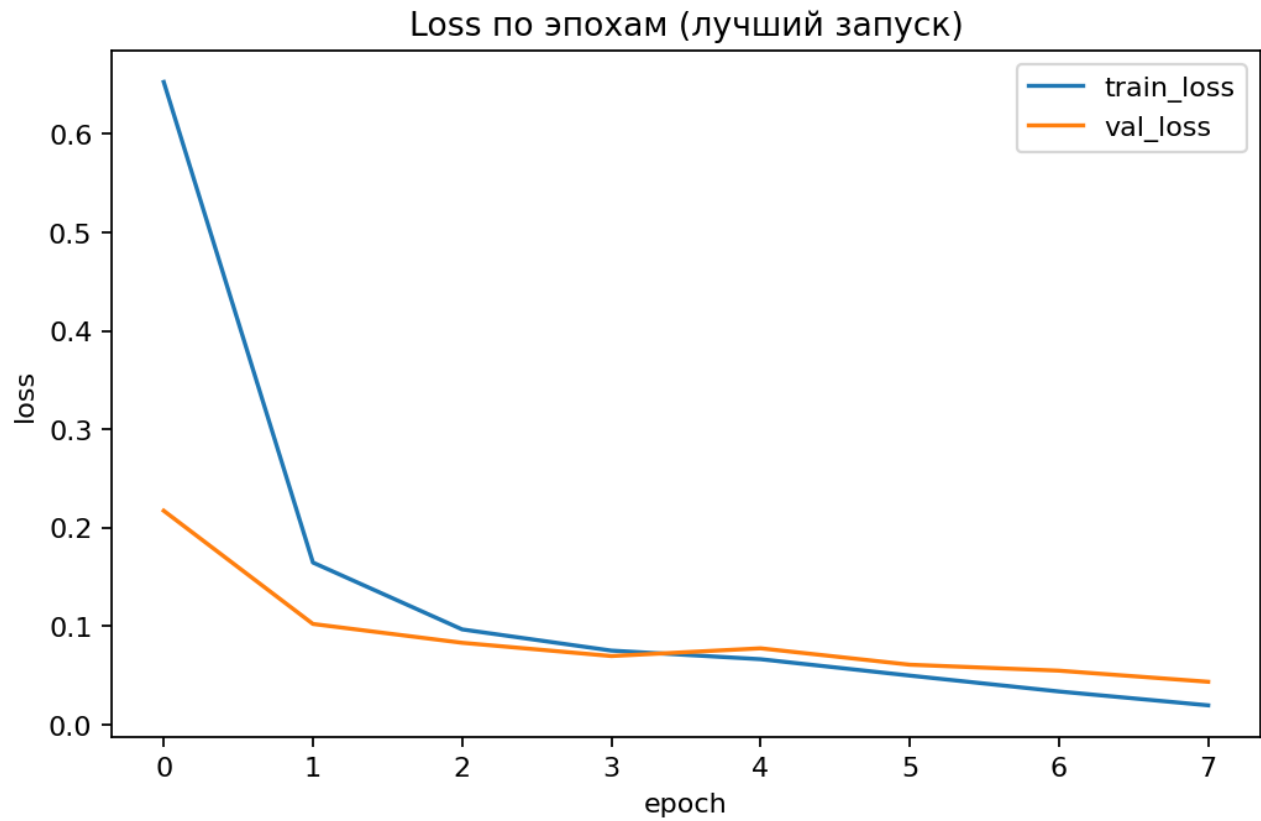
- Запустите TensorBoard: tensorboard --logdir runs
- Секции: Scalars (loss/accuracy/perf), Histograms (weights/grads/activations), Images (confusion matrix / examples), Graph.

Сводная таблица запусков (runs_summary.csv)

run_name	lr_strategy	lr	epochs_trained	best_val_accuracy	test_accuracy	avg_epoch_time_sec	rss_peak_mb
baseline_full_telemetry	fixed	0.001	8	0.9853333234786988	0.9848999977111816	1.6221855249998498	1536.40234375
baseline_full_telemetry	fixed	0.001	8	0.9853333234786988	0.9848999977111816	1.7527035250000154	1532.47265625
baseline_full_telemetry	fixed	0.001	8	0.9853333234786988	0.9848999977111816	1.897405424999988	1531.93359375
baseline_full_telemetry	fixed	0.001	8	0.9853333234786988	0.9848999977111816	1.7804962750001323	1532.0703125
bench_cpu	fixed	0.001	3	0.9711666703224182	0.9678999781608582	1.4311325666676566	1152.12890625
broken_bad_init	fixed	0.001	6	0.7459999918937683	0.7466999888420105	1.8478564333334664	1591.9453125
broken_bad_init	fixed	0.001	6	0.7459999918937683	0.7466999888420105	1.6687094499999755	1591.02734375
broken_bad_init	fixed	0.001	6	0.7459999918937683	0.7466999888420105	1.9542293166672	1589.421875
broken_bad_init	fixed	0.001	6	0.7459999918937683	0.7466999888420105	1.684482266666843	1591.2109375
broken_high_lr	fixed	0.001	6	0.1118333339691162	0.1134999990463256	1.8144812000000456	1568.58984375
broken_high_lr	fixed	0.001	6	0.1118333339691162	0.1134999990463256	1.6721889000000374	1568.5625
broken_high_lr	fixed	0.001	6	0.1118333339691162	0.1134999990463256	1.960506450000139	1565.44921875
broken_high_lr	fixed	0.001	6	0.1118333339691162	0.1134999990463256	1.6900842333328303	1567.62890625
broken_no_norm	fixed	0.001	6	0.9871666431427002	0.9866999983787536	1.8435079499998512	1614.50390625
broken_no_norm	fixed	0.001	6	0.9871666431427002	0.9866999983787536	1.819444766666796	1612.57421875
broken_no_norm	fixed	0.001	6	0.9871666431427002	0.9866999983787536	1.9521808999995	1613.10546875
broken_no_norm	fixed	0.001	6	0.9871666431427002	0.9866999983787536	1.7120215166663304	1613.703125
compare_cyclical	cyclical	0.001	8	0.987666666507721	0.9876999855041504	2.8749896624999565	1676.0625
compare_cyclical	cyclical	0.001	8	0.987666666507721	0.9876999855041504	3.243760049999992	1677.03515625
compare_cyclical	cyclical	0.001	8	0.987666666507721	0.9876999855041504	2.8187434250000933	1678.81640625
compare_fixed	fixed	0.001	8	0.9853333234786988	0.9848999977111816	1.870647062499984	1634.3828125
compare_fixed	fixed	0.001	8	0.9853333234786988	0.9848999977111816	1.7188767124998776	1633.58203125
compare_fixed	fixed	0.001	8	0.9853333234786988	0.9848999977111816	1.9565354874998773	1633.4453125
compare_fixed	fixed	0.001	8	0.9853333234786988	0.9848999977111816	1.631881100000328	1632.36328125
compare_step_decay	step_decay	0.001	10	0.9869999885559082	0.98580002784729	1.6920233699999698	1649.8984375

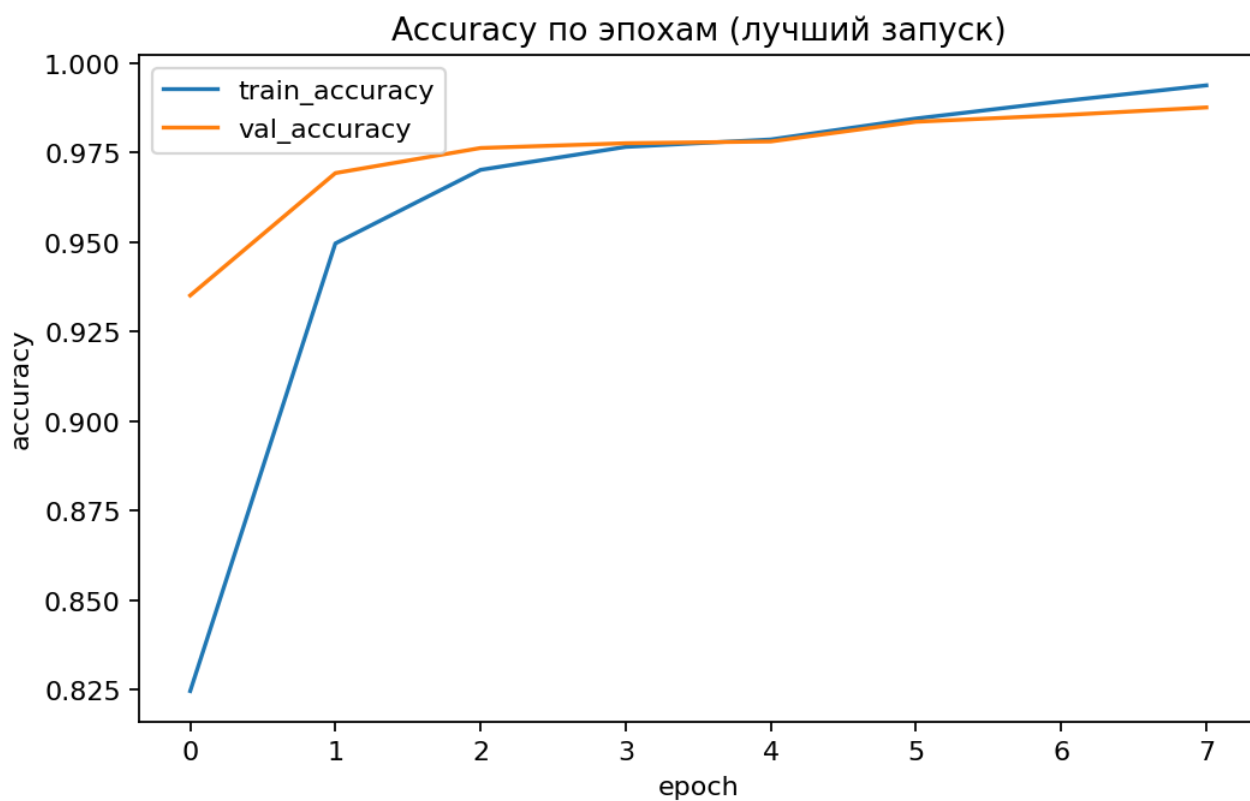
Визуальные материалы

Кривые loss (лучший запуск)



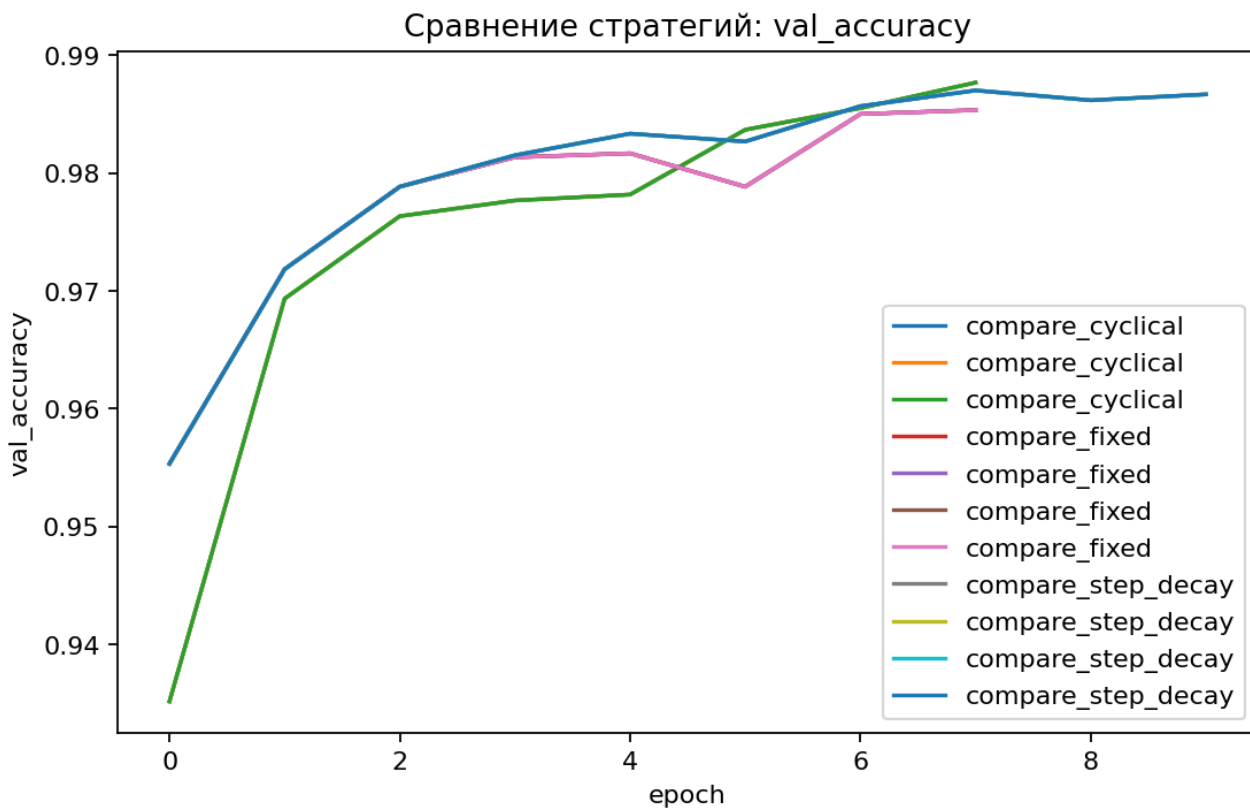
best_run_loss.png

Кривые ассигасу (лучший запуск)



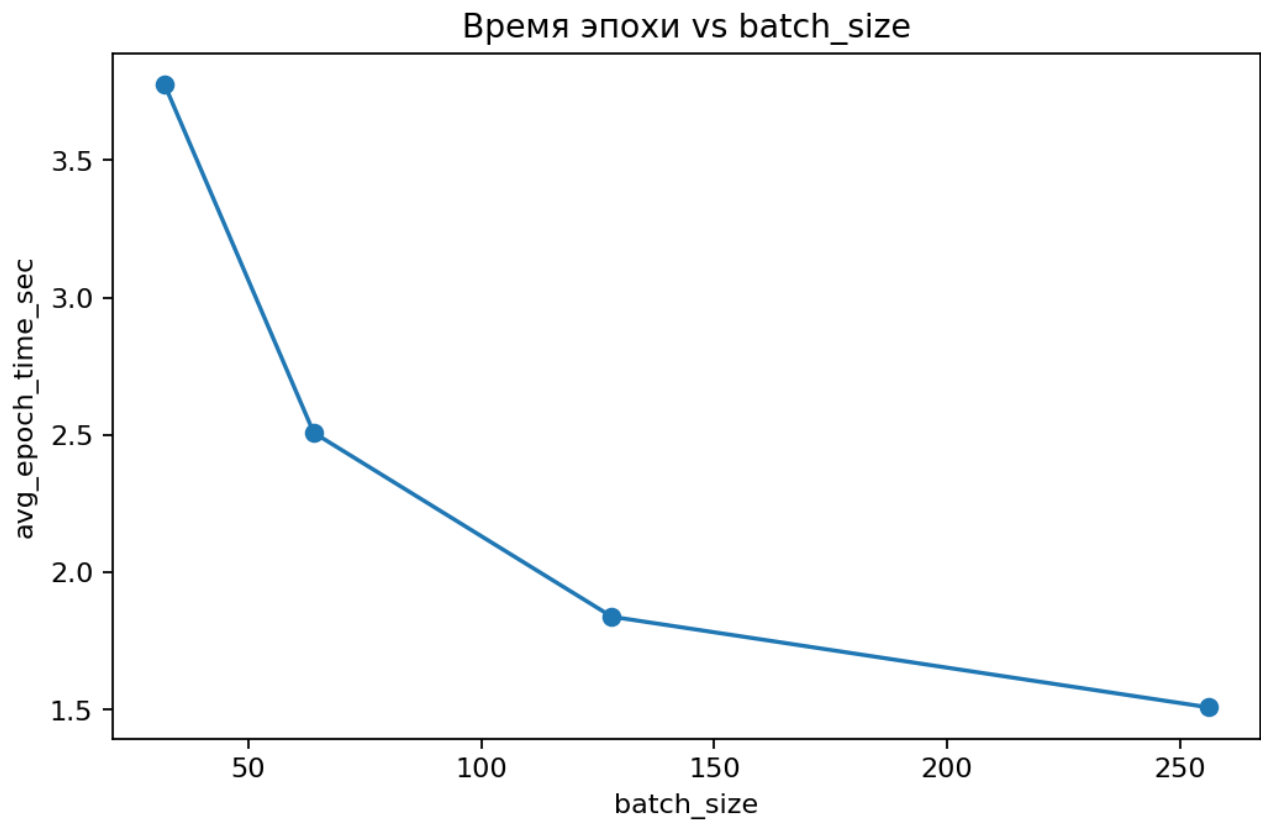
best_run_accuracy.png

Сравнение LR-стратегий по val_ассурасу



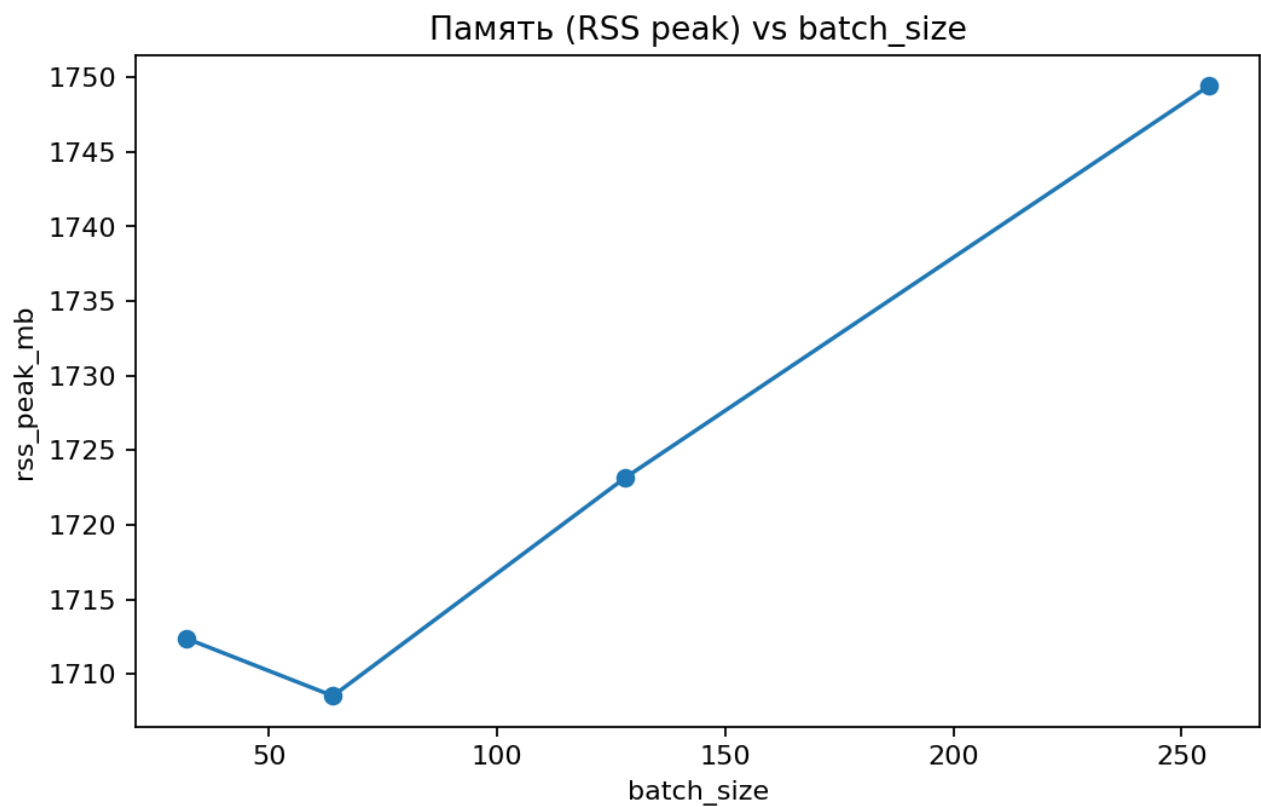
compare_lr_val_accuracy.png

Профилирование: время эпохи vs batch_size



profile_epoch_time_vs_batch.png

Профилирование: RSS peak vs batch_size



profile_rss_vs_batch.png