

LSML #7

Распараллеливание нейросетей
и примеры применения

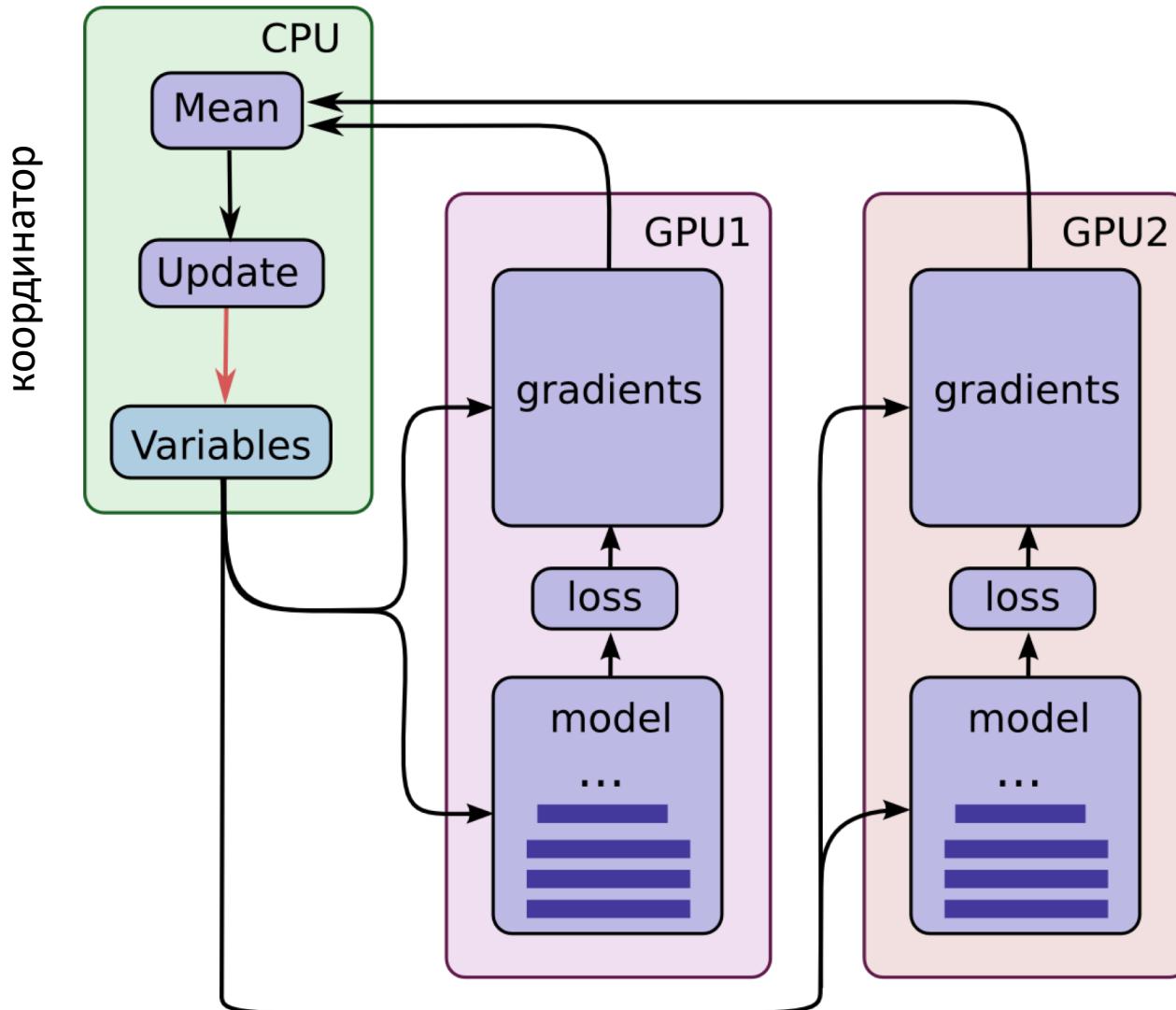
Как ускорять применение нейросети

- Если предсказываем для 1 примера:
 - Используют быстрый GPU (+ TensorRT) <https://venturebeat.com/2018/03/27/nvidia-speeds-up-deep-learning-inference-processing/>
 - Сжимают сети (teacher networks) <https://arxiv.org/pdf/1412.6550.pdf>
 - Квантуют веса сетей (быстрые INT8 операции) <https://www.tensorflow.org/performance/quantization> https://www.theregister.co.uk/2016/09/13/nvidia_p4_p40_gpu_ai/
 - Придумывают архитектуры для мобильных (MobileNet) <https://arxiv.org/pdf/1704.04861.pdf>
 - ...
- Если предсказываем для кучи примеров:
 - Embarrassingly parallel (предсказания делаются независимо)

Будем параллелизировать обучение

- Есть два подхода:
 - Data-parallel (по данным)
 - Model-parallel (по весам)

Data-parallel



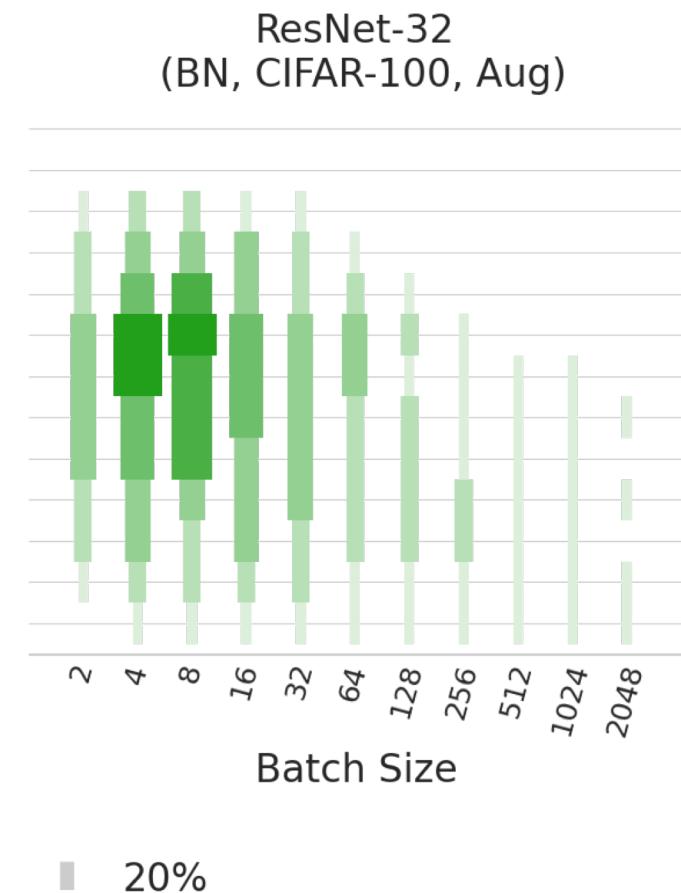
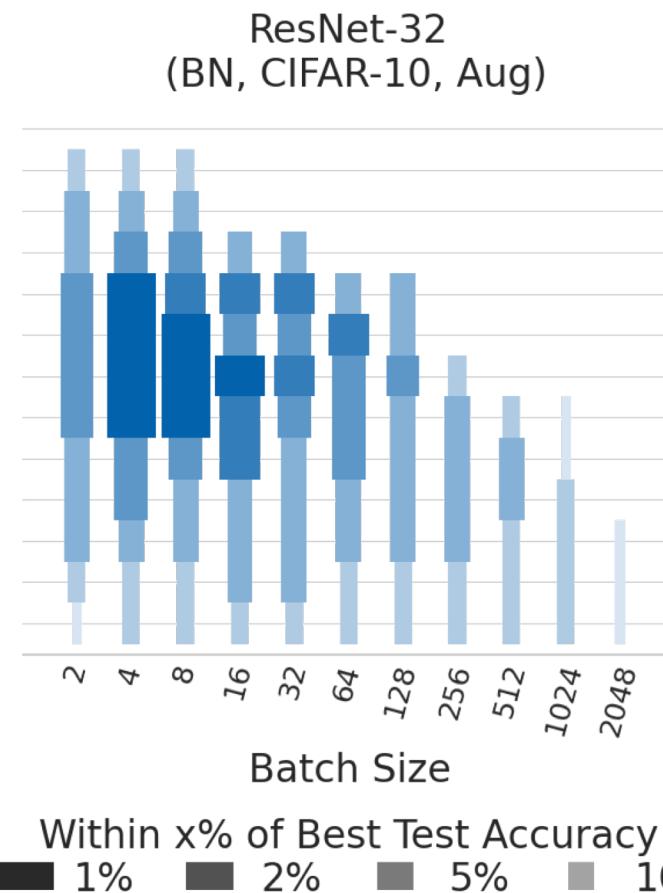
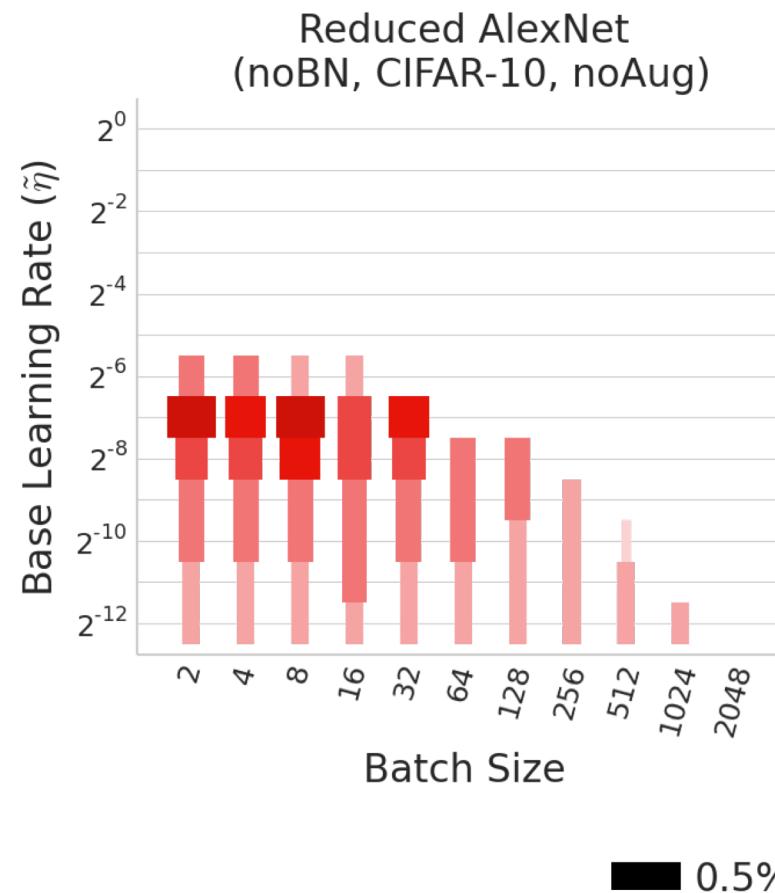
- Рассылаем копию модели на каждый GPU
- Применяем независимо к разным кусочкам батча
- Вклады в градиент отправляем **координатору** на CPU (или GPU), суммируем и заново рассылаем всем GPU

Data-parallel

- Все упирается в размер батча, сойдется ли с большим?
- Если модель большая, то долгие пересылки

Размер батча

- Кто-то говорит, что нужно брать маленький:



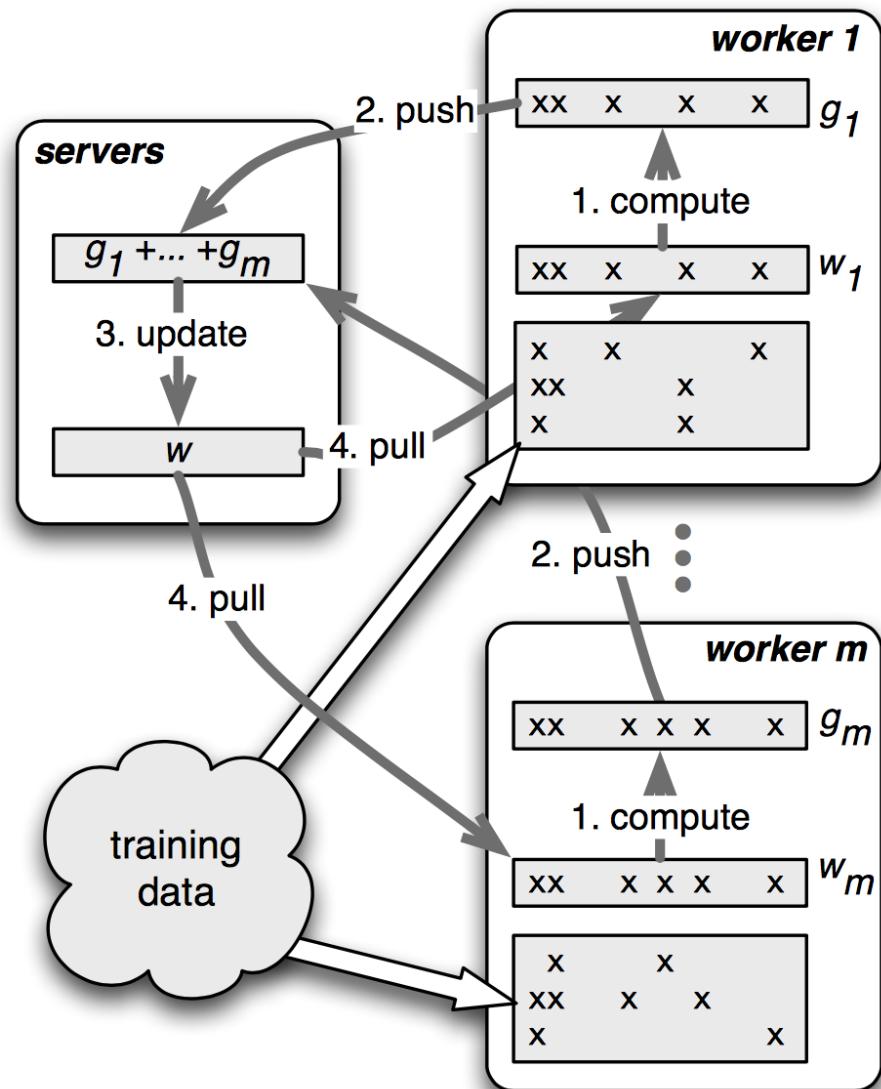
Размер батча

- Маленький плох для распараллеливания:
 - Не полностью утилизируется GPU
 - Маленькое количество воркеров

Data-parallel

- Все упирается в размер батча, сойдется ли с большим?
- **Если модель большая, то долгие пересылки**

Усложним координатор: Parameter Server (PS)



На примере линейной регрессии:

1. Воркеры читают свою часть данных и считают градиенты \mathbf{g}
2. Воркеры отсылают (**push to PS**) градиенты \mathbf{g} на сервер параметров
3. Сервер параметров обновляет свои веса \mathbf{w} на базе сообщений от воркеров
4. Воркеры просят у сервера параметров (**pull from PS**) обновленные веса \mathbf{w}

Parameter Server (PS)

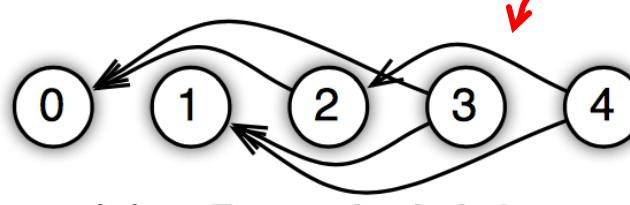
- PS может состоять из многих машин (шардирование параметров)
- Можно варьировать **консистентность** параметров, что позволяет делать вычисления асинхронно



(a) Sequential



(b) Eventual



(c) 1 Bounded delay

iter 10: gradient → push & pull → o

iter 11: gradient → push & pull → o

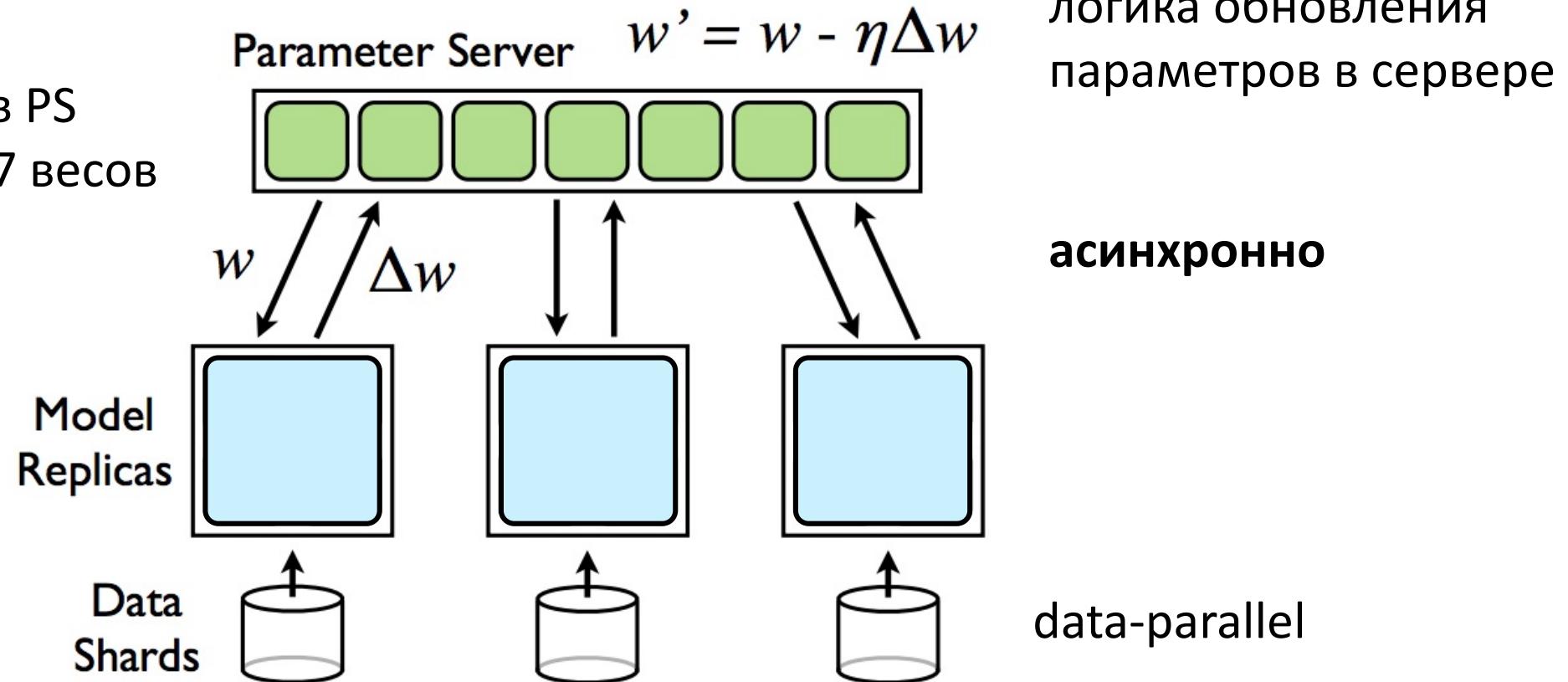
iter 12: gradient → pu

Пример:
оставать не
больше, чем
на 2 апдейта
весов

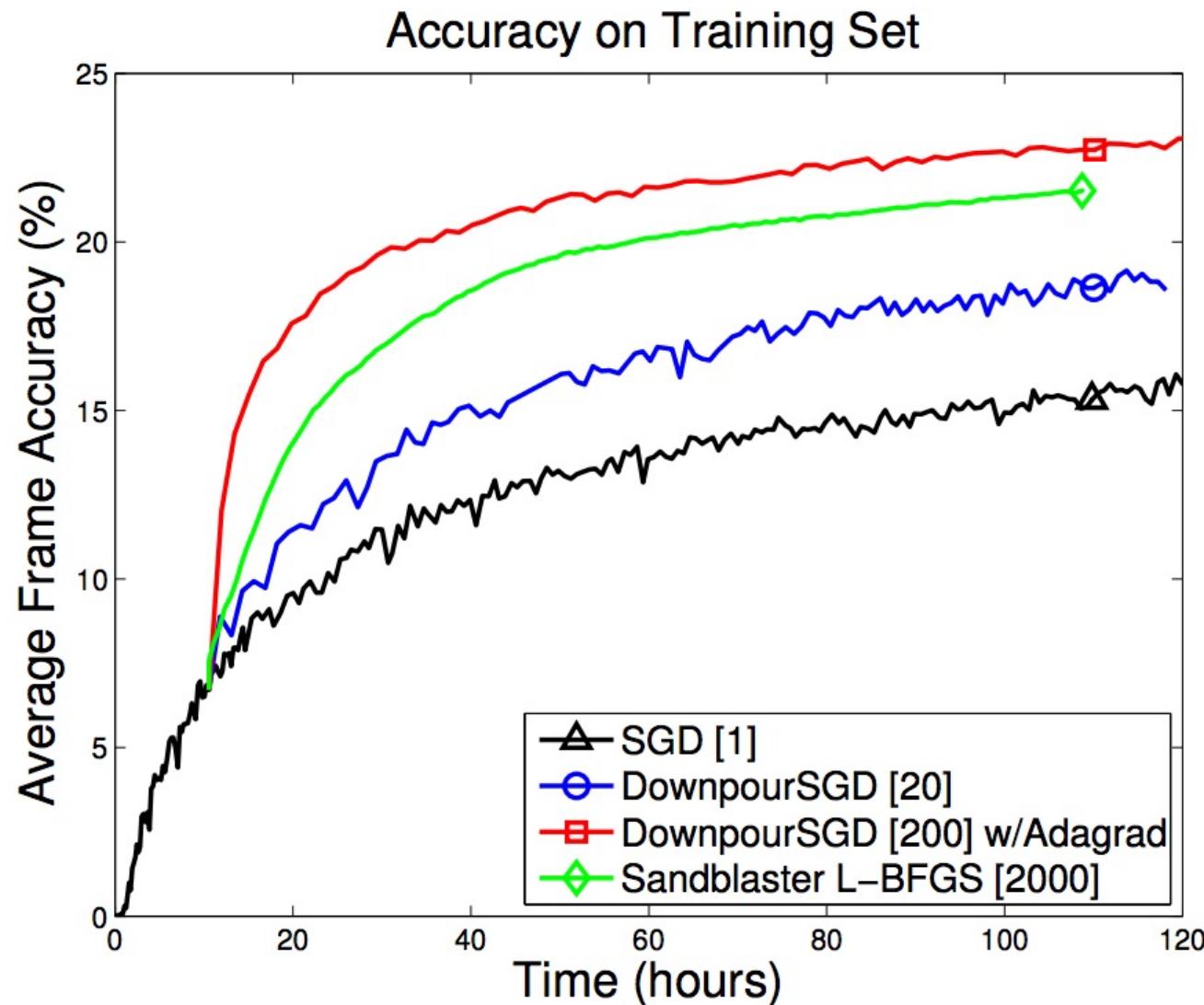
Уменьшим overhead от пересылок

- **Асинхронный SGD:** не будем ждать, пока прилетит вклад в градиент от всех машин, сделаем шаг с тем, что есть (возможно только локальный вклад)

каждая машина в PS
обрабатывает 1/7 весов



Асинхронный SGD от Google (DistBelief)



Delayed Block Proximal Gradient – пример с PS

Divide coordinates into B blocks. Set the order $b(1), \dots, b(T)$ and the maximal delay τ

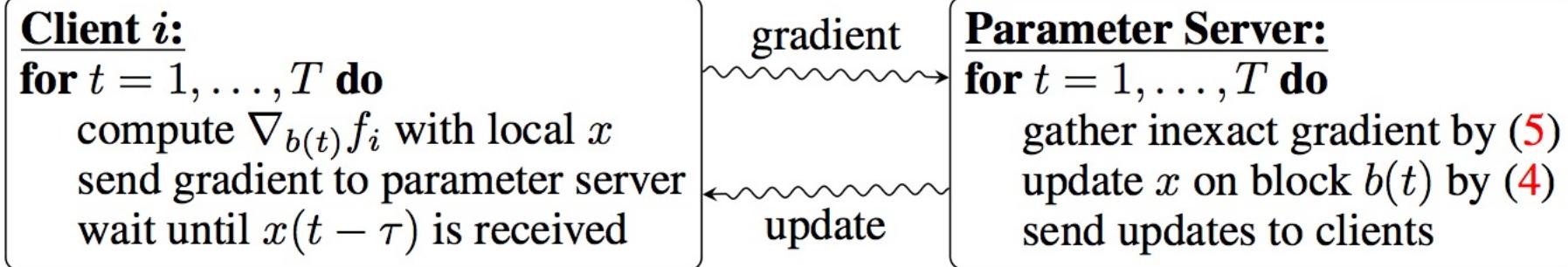


Figure 1: D2P, Distributed Delayed Proximal Gradient Methods. Both clients and the parameter server span several machines. All data sending and receiving are non-blocking.

- Линейная регрессия
- На итерации t обновляем веса в блоке $b(t)$
- Хотим на каждом воркере отставать не больше, чем на τ блоков
- Асинхронно посылаем апдейты весов

Delayed Block Proximal Gradient – пример с PS

Divide coordinates into B blocks. Set the order $b(1), \dots, b(T)$ and the maximal delay τ

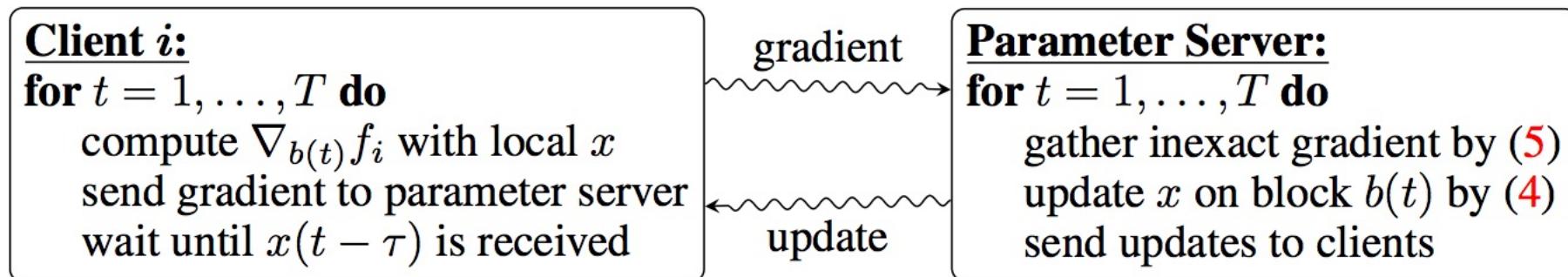
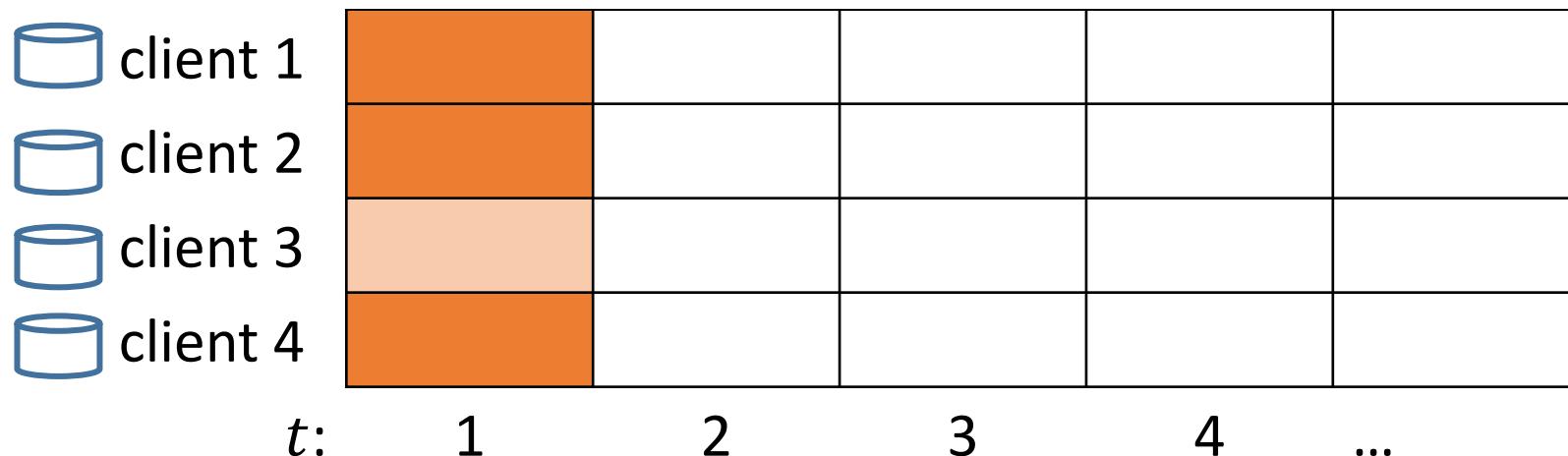


Figure 1: D2P, Distributed Delayed Proximal Gradient Methods. Both clients and the parameter server span several machines. All data sending and receiving are non-blocking.



К какому-то
моменту первую
итерацию
завершили 1, 2 и 4
воркеры

Delayed Block Proximal Gradient – пример с PS

Divide coordinates into B blocks. Set the order $b(1), \dots, b(T)$ and the maximal delay τ

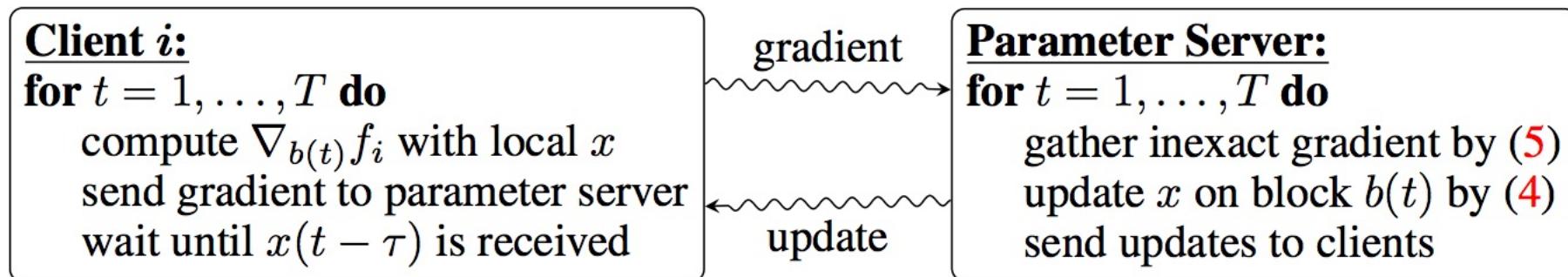
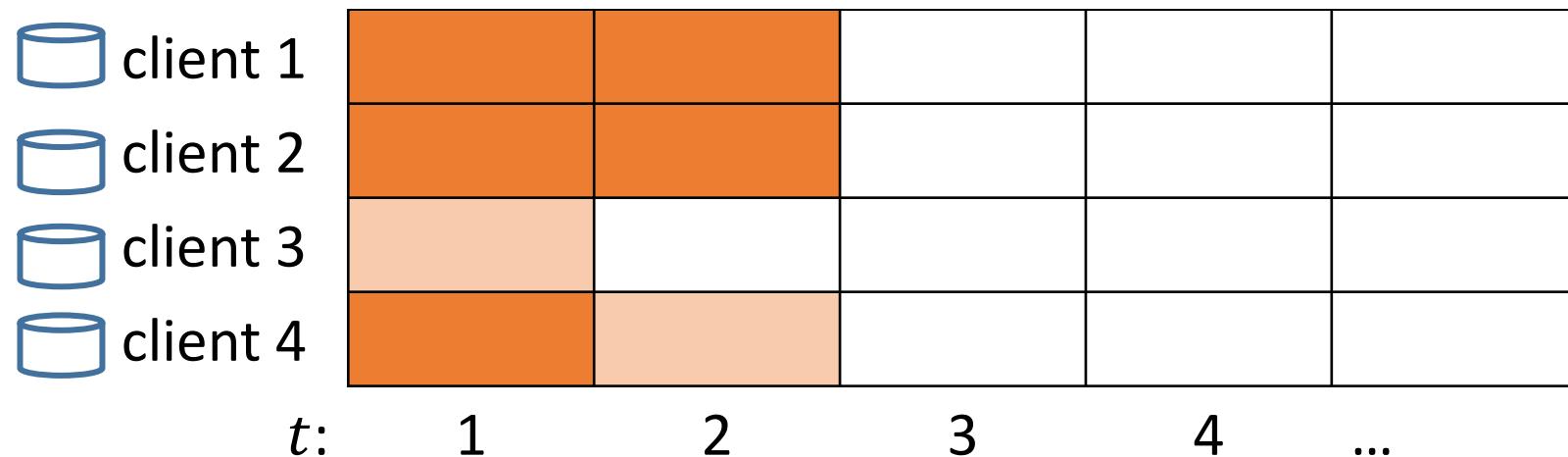


Figure 1: D2P, Distributed Delayed Proximal Gradient Methods. Both clients and the parameter server span several machines. All data sending and receiving are non-blocking.



Теперь вторую
итерацию
закончили 1 и 2
воркеры

Delayed Block Proximal Gradient – пример с PS

Divide coordinates into B blocks. Set the order $b(1), \dots, b(T)$ and the maximal delay τ

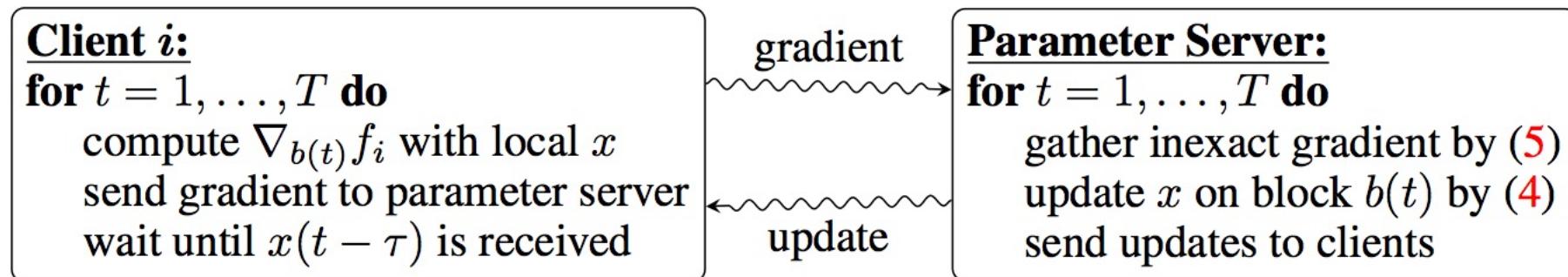
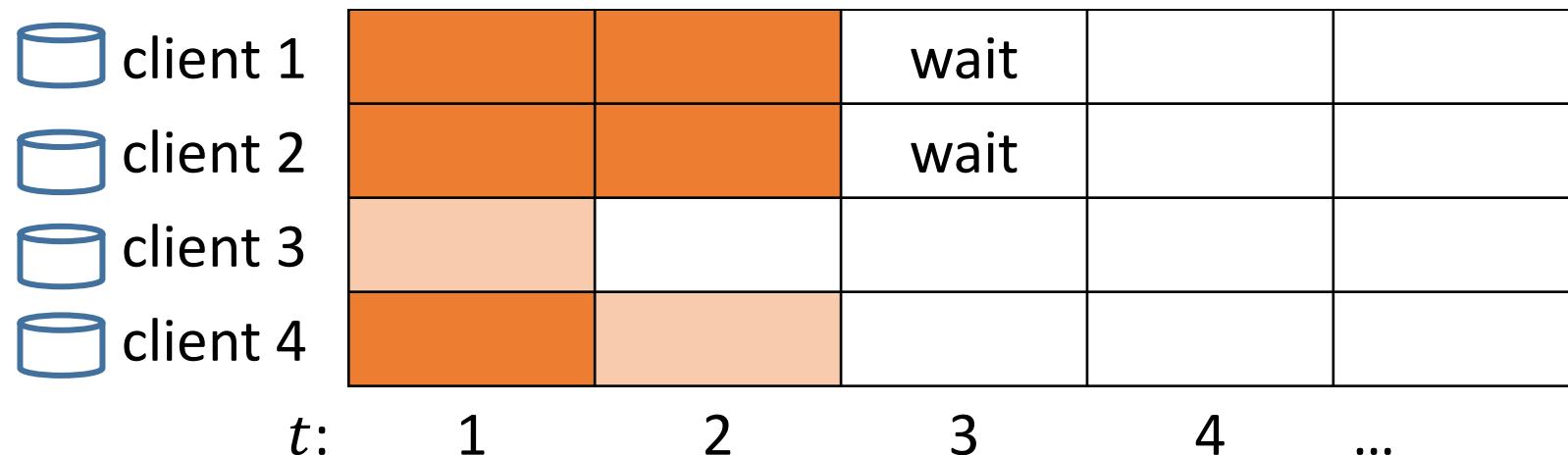


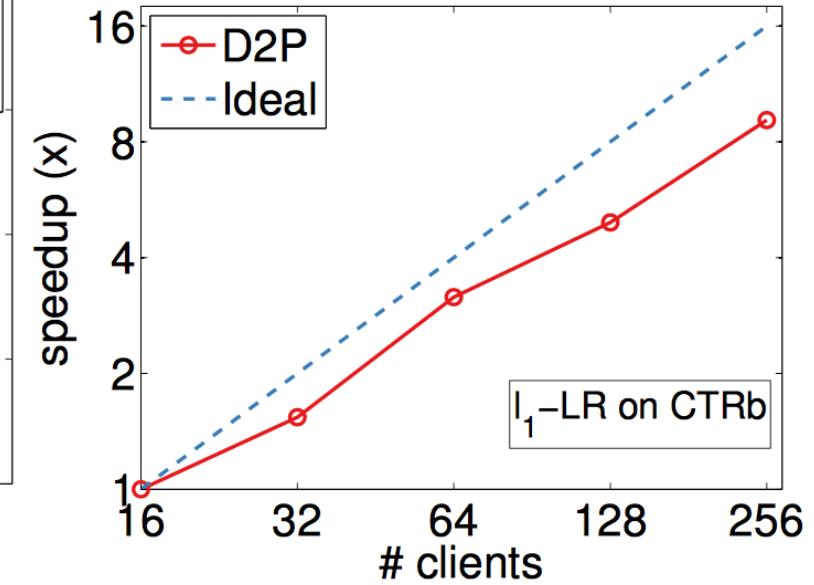
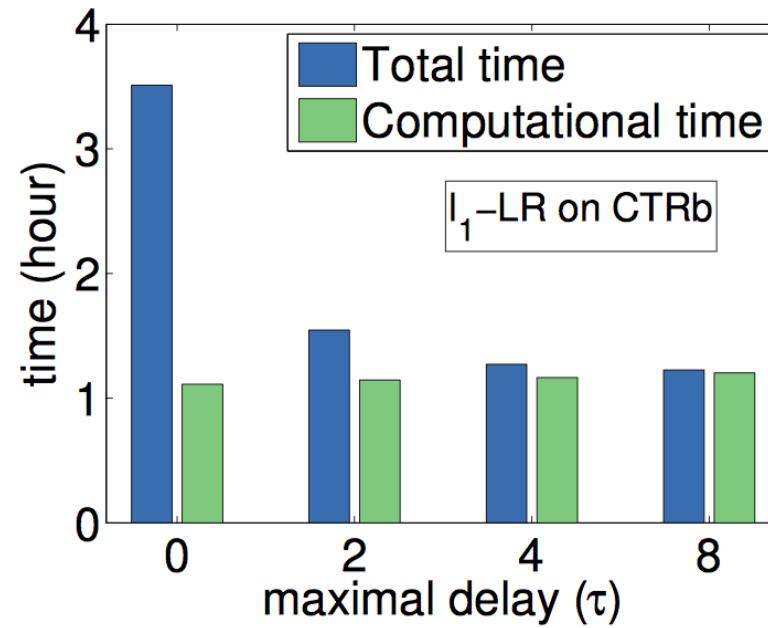
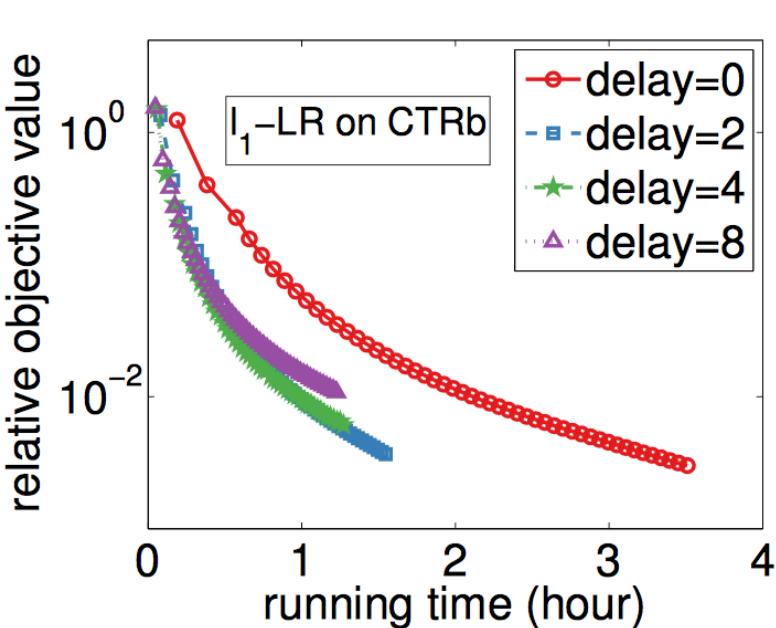
Figure 1: D2P, Distributed Delayed Proximal Gradient Methods. Both clients and the parameter server span several machines. All data sending and receiving are non-blocking.



При $\tau = 2$ мы не
можем продолжить
вычисления на 1 и
2 воркере

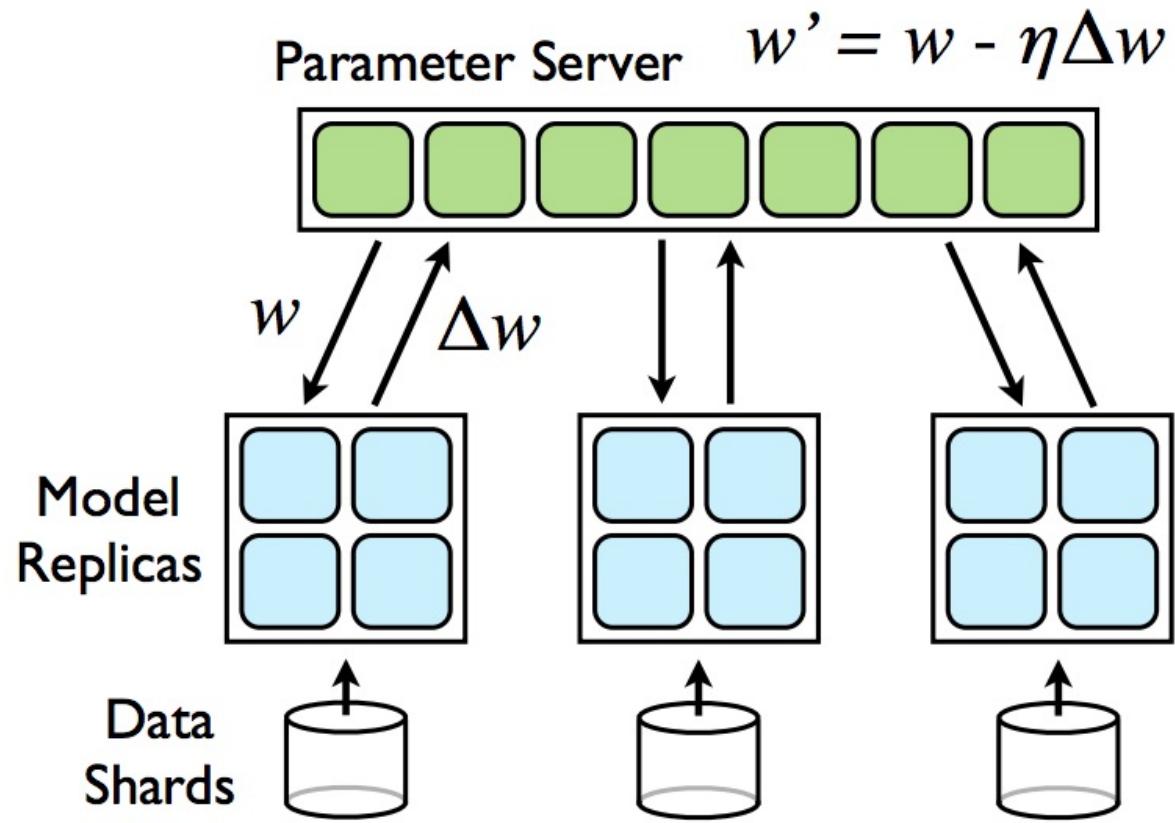
Delayed Block Proximal Gradient – пример с PS

- В τ раз меньше синхронизаций
- Замедление сходимости компенсируется ускорением одного шага

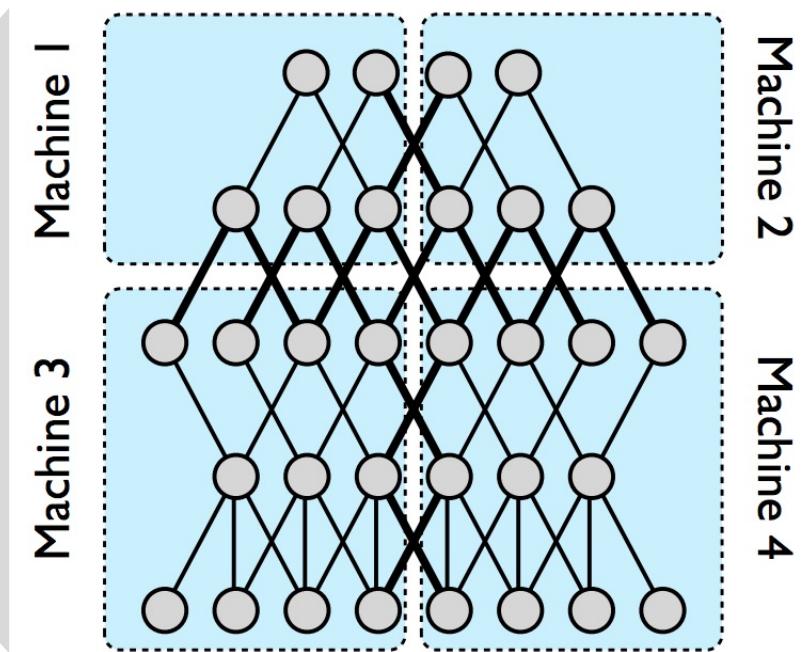


Model-parallel

- Режим веса на партиции (можем их обновлять асинхронно)

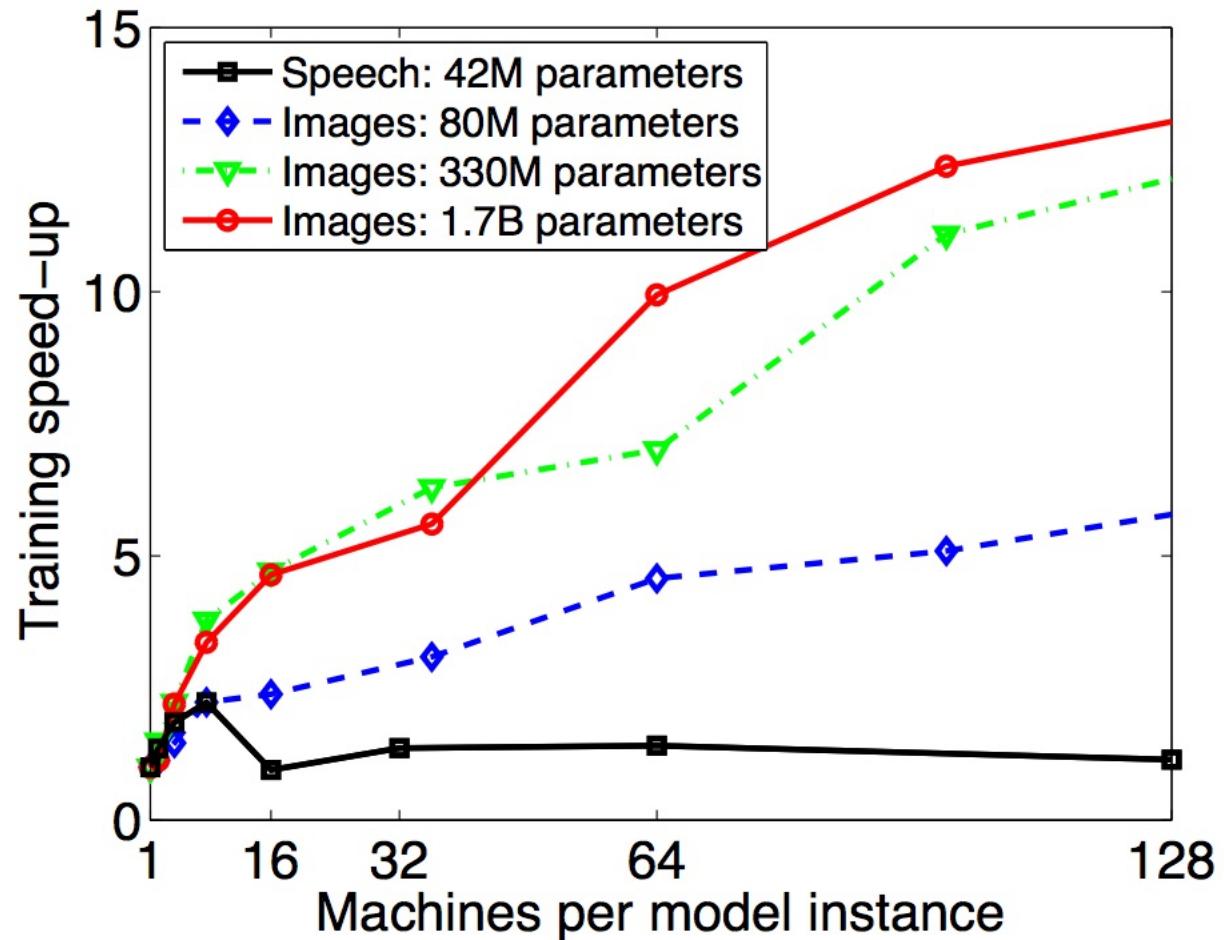


Для сверточных сетей ОК



Model-parallel в DistBelief

Для сверточных сетей ОК



Побили в свое время
state-of-the-art
на большом ImageNet

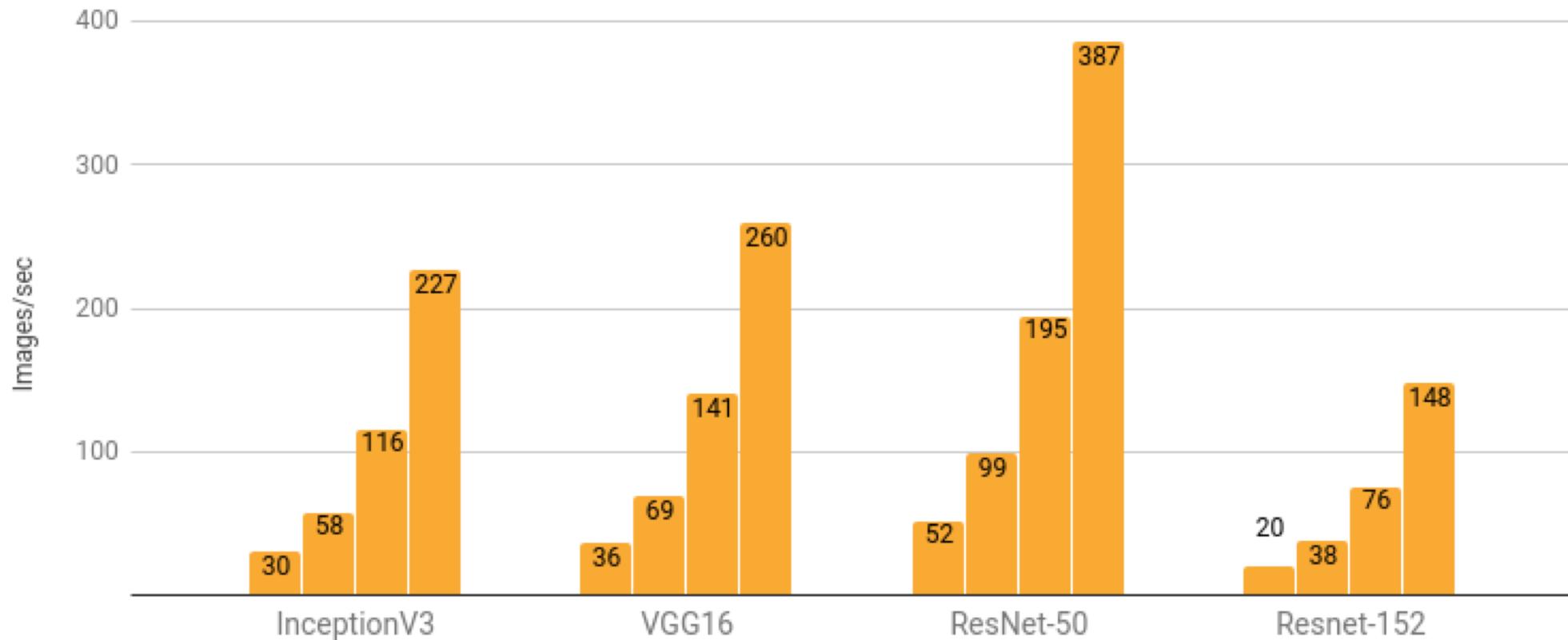
Model-parallel

- Сложнее реализовывать (нужны выходы нейронов от соседа)
- Не всегда нужно, как правило модель не очень большая

TF на одной машине с GPU

Data-parallel с огромным батчем (256 на каждый GPU)

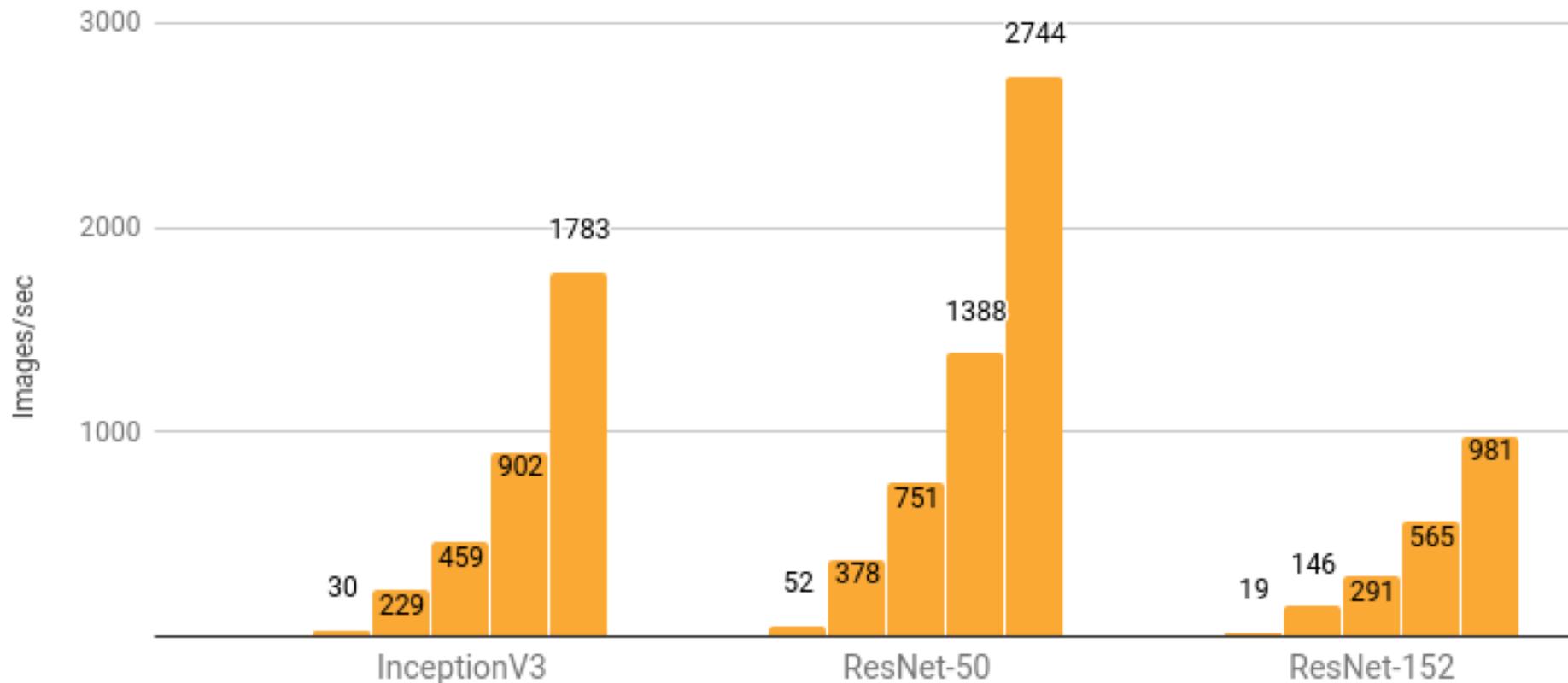
Training: NVIDIA® Tesla® K80 synthetic data (1,2,4, and 8 GPUs)



TF на кластере с GPU

Data-parallel с огромным батчем (64 на каждый GPU)

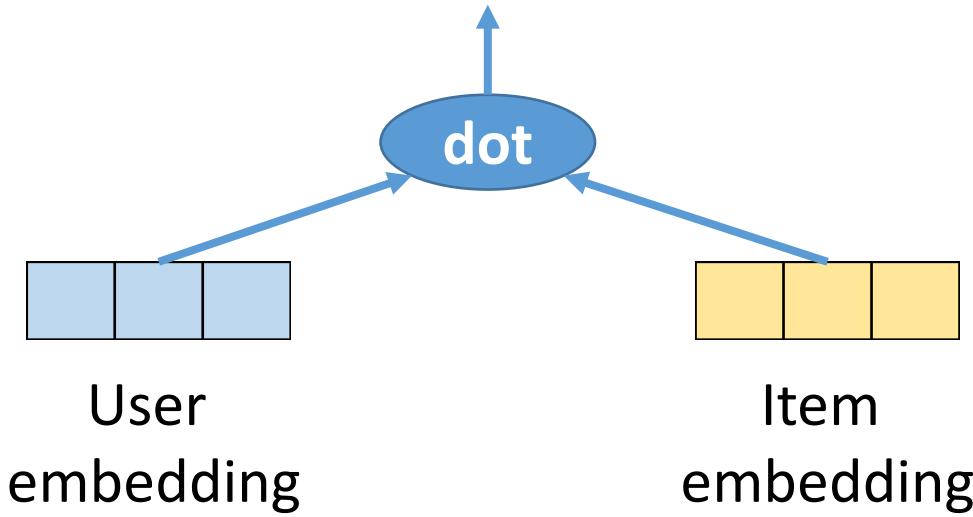
Training: NVIDIA® Tesla® K80 synthetic data (1,8,16,32, and 64)



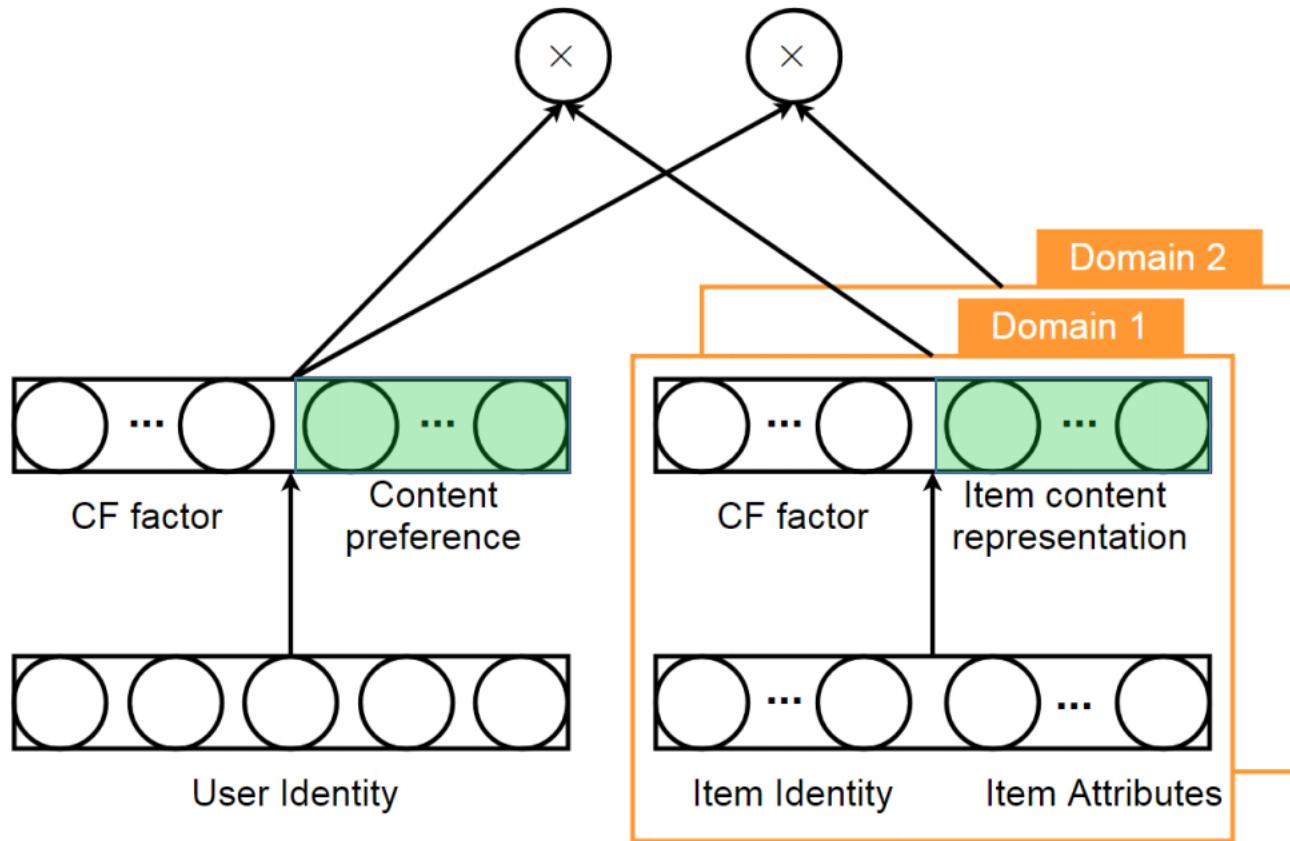
Примеры применений нейросетей

- Не только картинки и тексты
- Еще, например, рекомендации

Matrix Factorization Collaborative Filtering

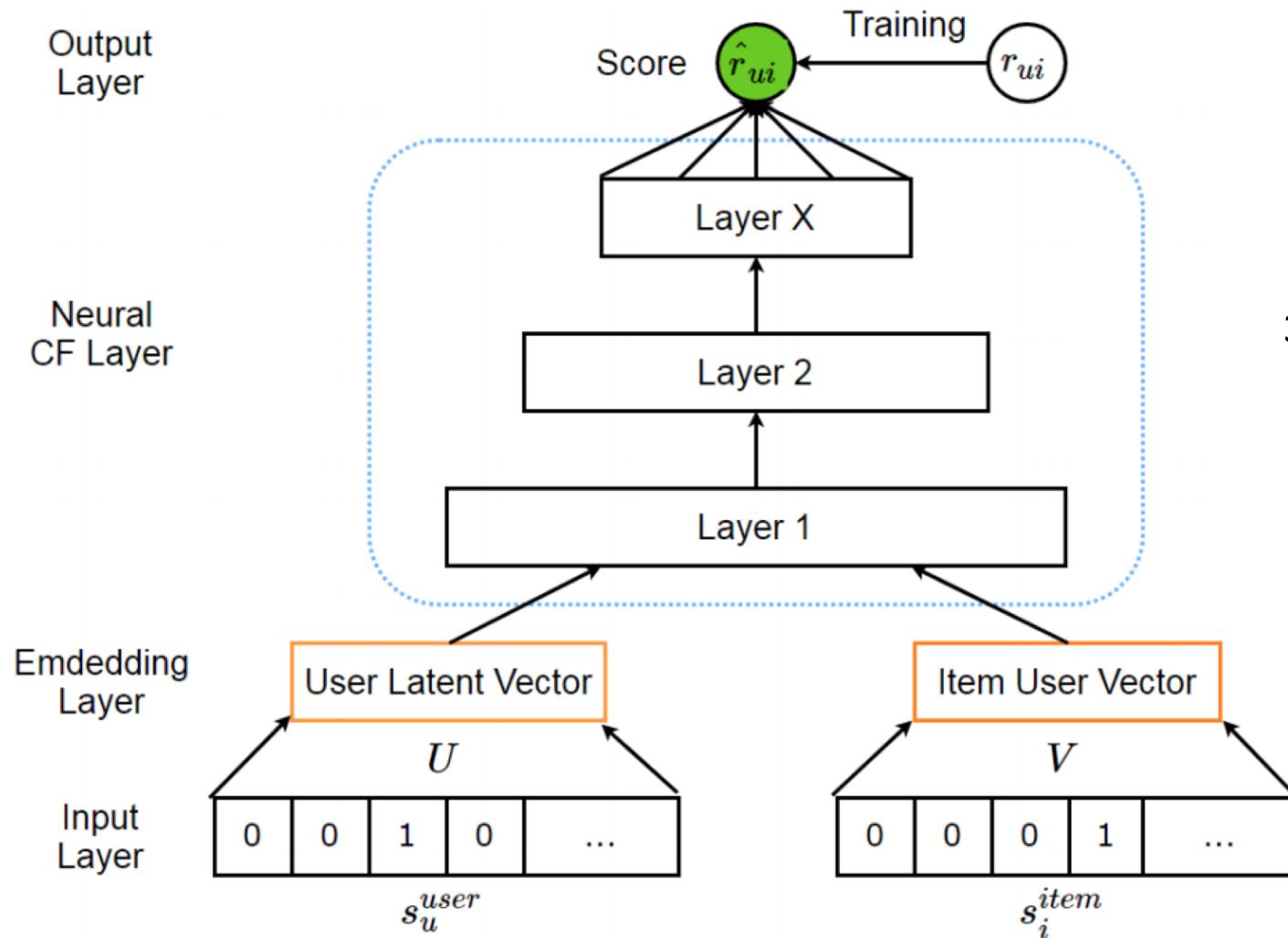


Cross-domain Content-boosted CF



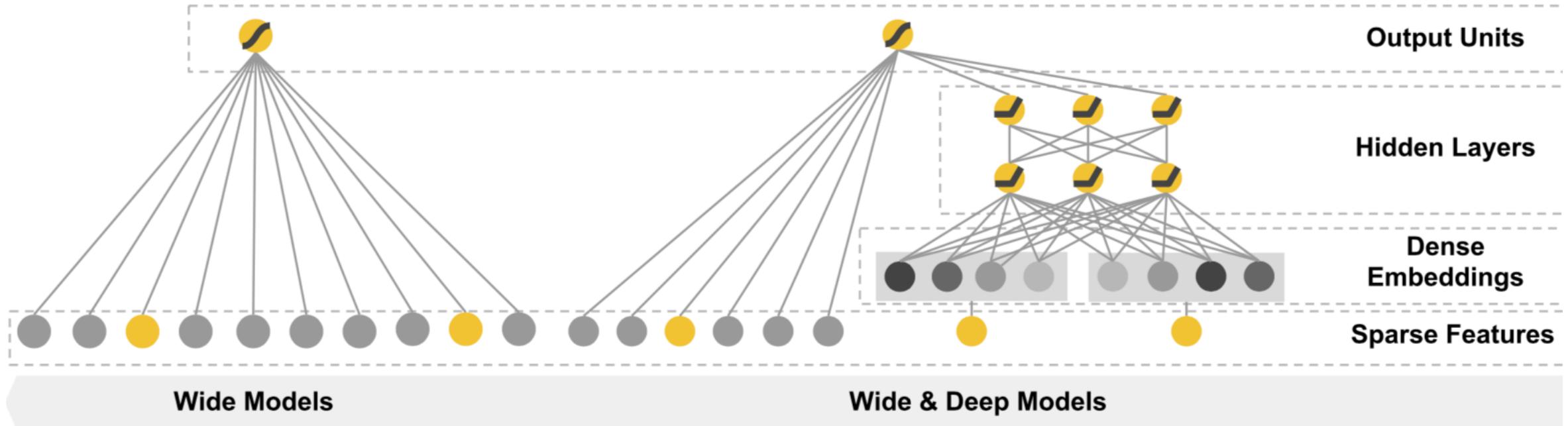
Имбедим
КОНТЕНТНЫЕ
признаки
айтемов

Neural Collaborative Filtering



Нелинейная
зависимость
имбедингов

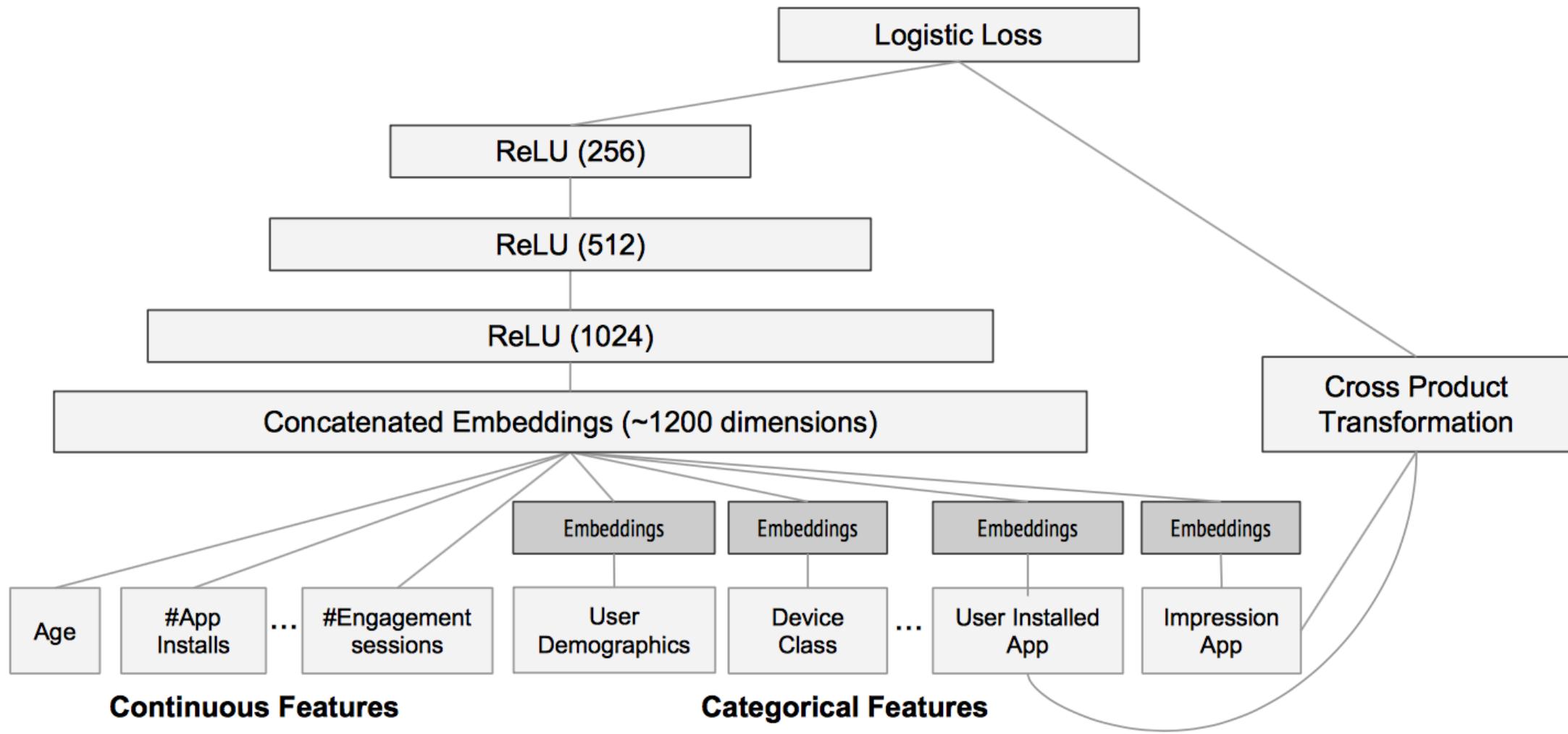
Wide & Deep Learning



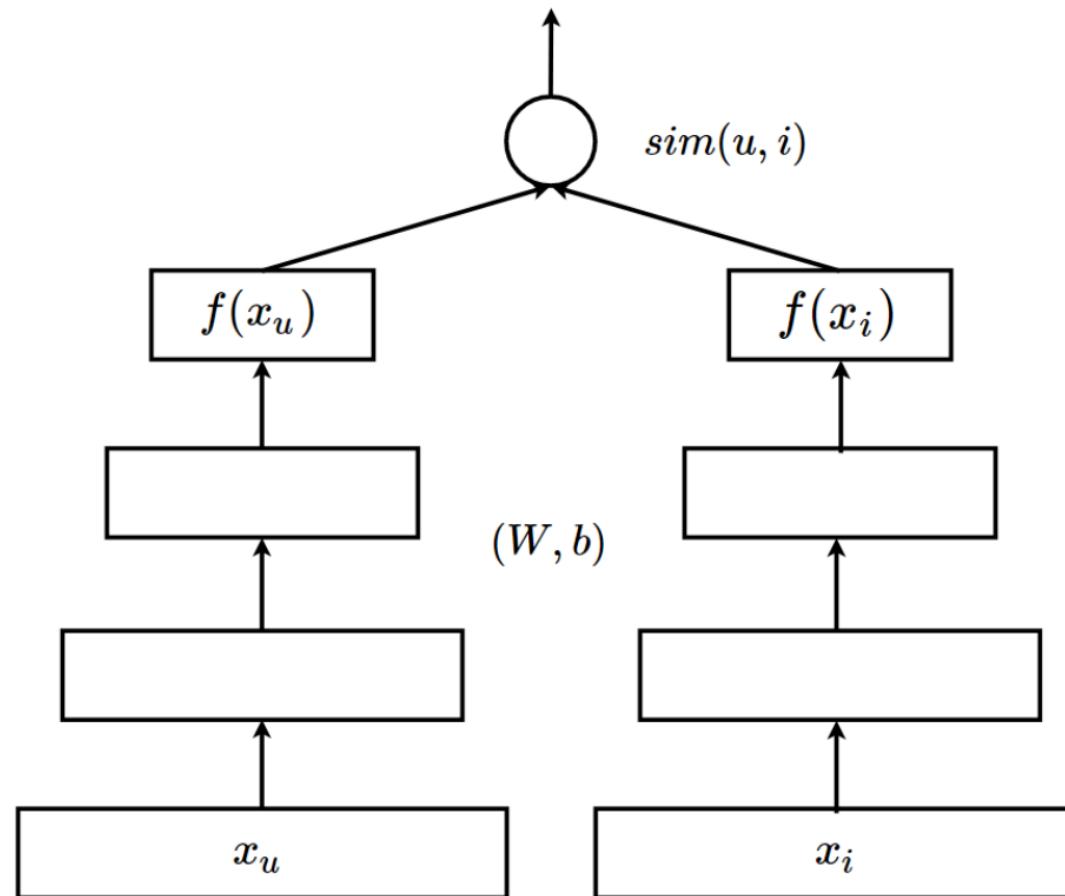
Добавляем
линейные
зависимости

https://www.tensorflow.org/tutorials/wide_and_deep

Wide & deep для рекомендаций в Play Store



DSSM



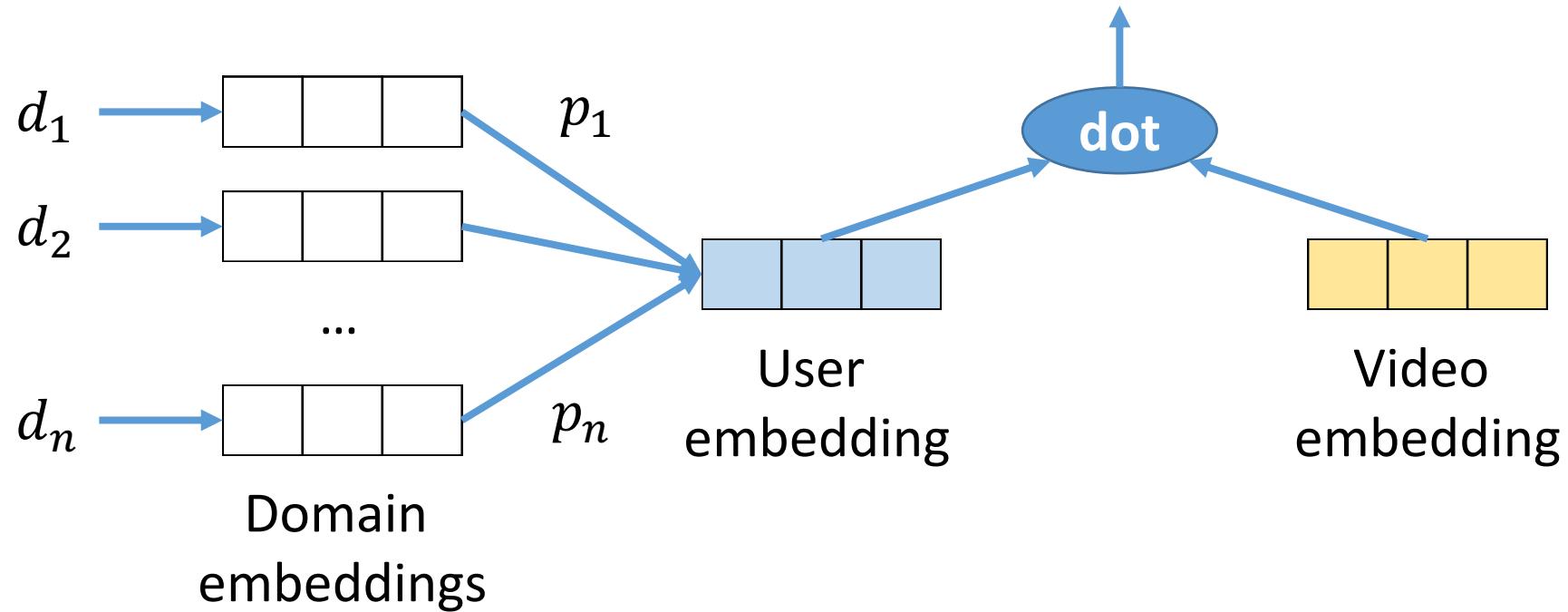
Тяжелые
имбеддинги
можно считать
в офлайне

Удобно делать рекомендации в TF

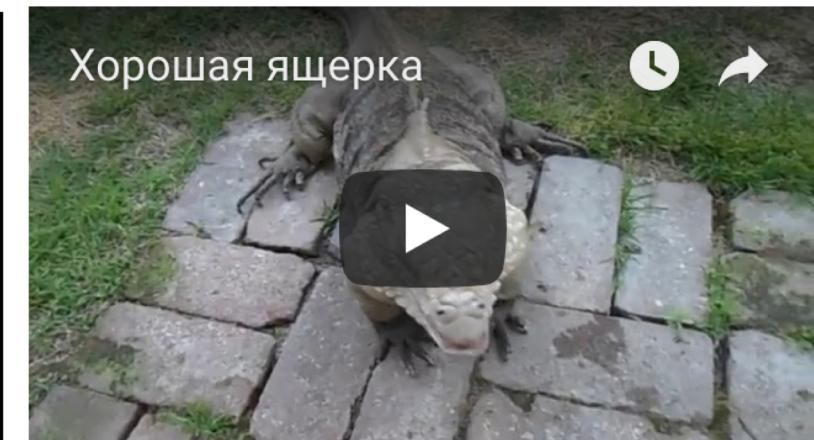
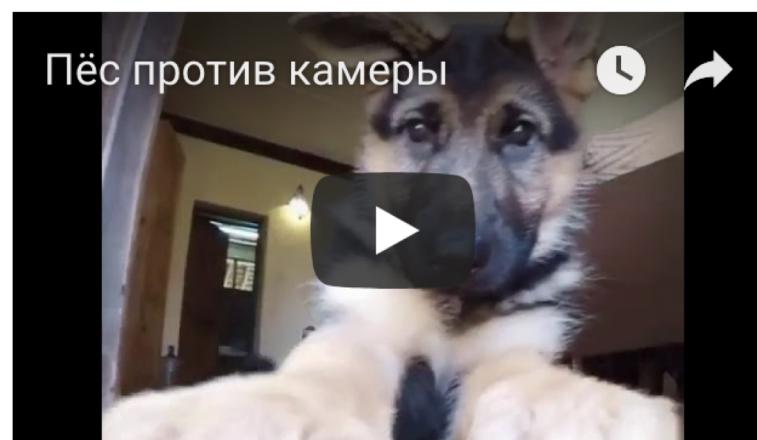
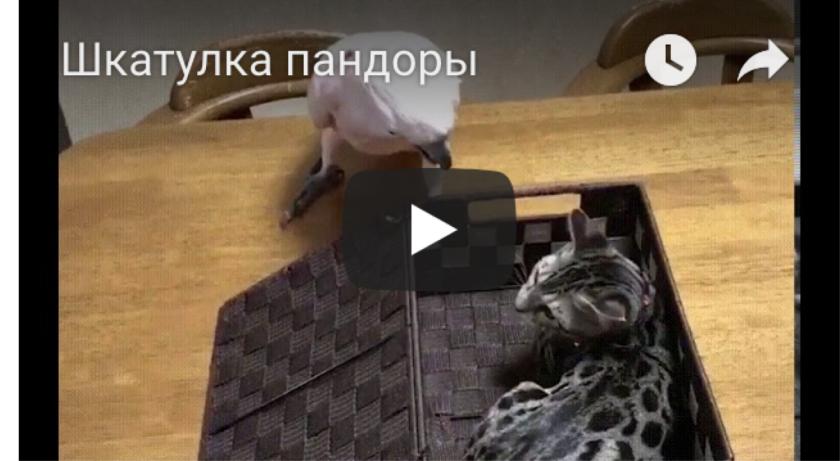
- Можно использовать крутые оптимайзеры (Adam, ...)
- Можно имбеддить картинки, текст, ...
- ...

Реальная задача: домены → видео

Про домены больше информации, решаем проблему холодного старта

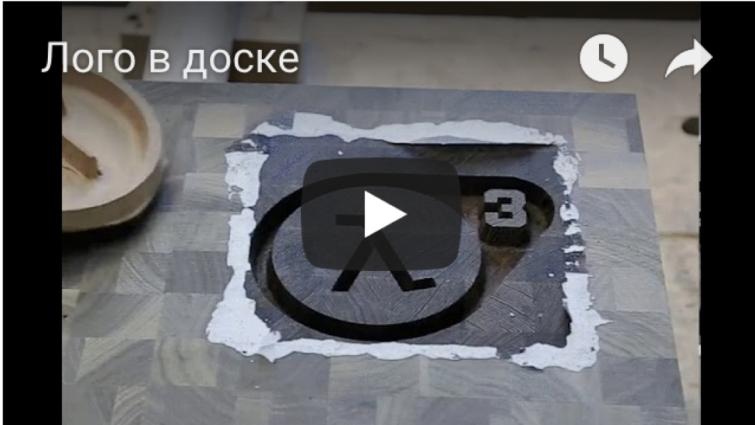


Похожие видео



Самые похожие на hi-tech.mail.ru

[0.94703776]



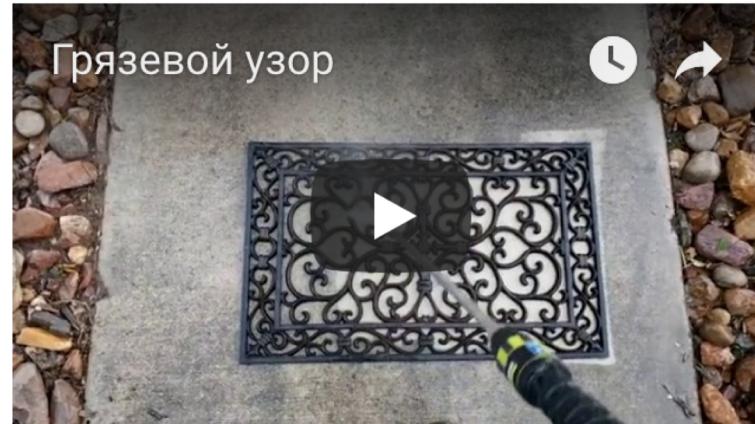
[0.92087275]



[0.9381]

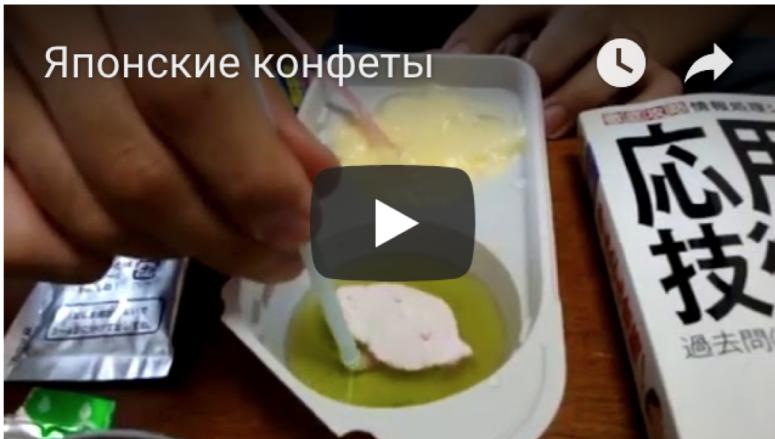


[0.9200871]



Самые похожие на goodhouse.ru

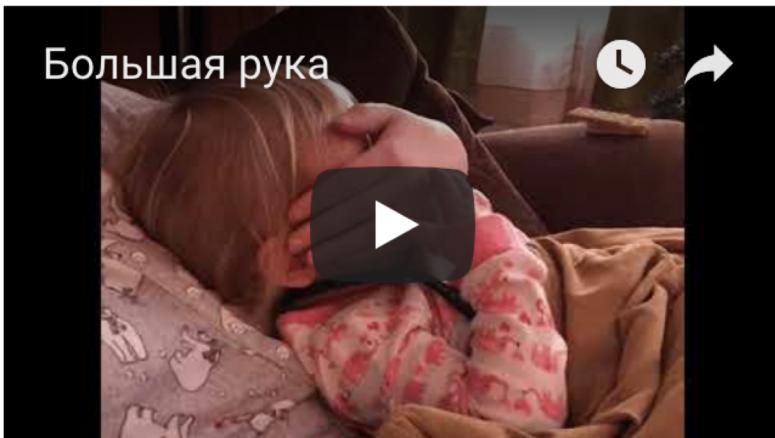
[0.89342886]



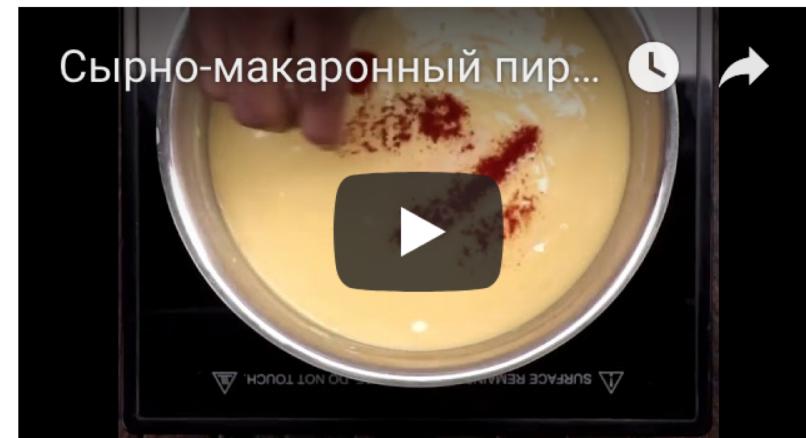
[0.8903276]



[0.89316946]



[0.88709587]



Ссылки

- Статья о Parameter Server

[https://www.cs.cmu.edu/~muli/file/parameter server osdi14.pdf](https://www.cs.cmu.edu/~muli/file/parameter_server_osdi14.pdf)

- Еще про PS

http://opt.kyb.tuebingen.mpg.de/papers/opt2013_submission_1.pdf