

# Мониторинг моделей глубокого обучения средствами библиотеки Keras

---

## Что такое многозадачное обучение?

---

Если есть  $n$  задач (традиционные подходы глубокого обучения направлены на решение только 1 задачи с использованием 1 конкретной модели), где эти  $n$  задач или их подмножество связаны друг с другом, но не совсем идентичны, Multi-Task Learning (MTL) поможет в улучшении обучения конкретной модели с использованием знаний, содержащихся во всех  $n$  задачах

## Для чего используется LearningRateScheduler?

---

Часто полезно уменьшить скорость обучения с течением времени. Обучая нашу сеть, мы пытаемся найти какое-то место в нашем ландшафте потерь, где сеть получает разумную точность. Это не обязательно должен быть глобальный минимум или даже локальный минимум, но на практике достаточно просто найти область с достаточно низкими потерями.

Если мы будем постоянно поддерживать высокий learning rate, мы можем перескочить эти области с низкими потерями, так как будем предпринимать слишком большие шаги.

Вместо этого мы можем уменьшить скорость обучения, тем самым позволяя нашей сети предпринимать меньшие шаги - эта сниженная скорость обучения позволяет нашей сети опускаться в районы потерь, которые «более оптимальны» и в противном случае были бы полностью пропущены из-за нашего learning rate.

Или наоборот, если мы будем постоянно поддерживать низкий learning rate, то сеть будет учиться очень долго или вообще не сможем вылезти из локального минимума, который далеко не является лучшим решением задачи минимизации потерь. Тогда надо скорость повышать.

Как видно из названия, LearningRateScheduler регулирует скорость обучения на основе определенного расписания. Например, в качестве расписания используется затухание времени или экспоненциальное затухание и другие. Чтобы реализовать эти затухания, Keras предоставил коллбэк, известный как LearningRateScheduler, который регулирует весовые коэффициенты на основе предоставленной функции затухания.