

Лабораторная работа №4

Предобученные нейронные сети

Цель работы – Научиться использовать предобученные модели из torchvision, адаптировать их под новую задачу, проводить тонкую настройку (fine-tuning) и оценивать качество.

Задание: решить задачу классификации при помощи предобученной нейронной сети на примере датасета FashionMNIST, выполнить дообучение нейронной сети под задачу и оценить адекватность модели:

1. в качестве исходных данных использовать датасет изображений FashionMNIST: <https://www.kaggle.com/datasets/zalando-research/fashionmnist>;
2. выполнить предобработку данных, нормализацию, аугментацию при необходимости, сформировать тензоры;
3. выбрать одну из обученных моделей: VGG11 (`model = vgg11(weights=VGG11_Weights.IMAGENET1K_V1)`) или GoogleLeNet (`model = googlenet(weights = GoogLeNet_Weights.IMAGENET1K_V1)`), предобученные на ImageNet;
4. адаптировать первый слой под один канал (черно-белое изображение);
5. заменить выходной слой для решения задачи классификации FashionMNIST;
6. дообучить модель и выполнить проверку ее адекватности на тренировочных данных;
7. выполнить оценку адекватности модели на тесте;
8. выполнить классификацию одного выбранного из датасета объекта;
9. составить отчет о проделанной работе в соответствии с требованиями кафедры.

Требования к отчету. Отчет должен содержать постановку задачи, исходные данные, результаты решения задачи, необходимые иллюстративные материалы.

Требования к защите

Защита лабораторной работы происходит индивидуально. Система оценки – рейтинговая.

Критерии оценки:

- корректность выполненного исследования;
- адекватность полученных результатов;
- качество отчета;
- качество ответов на контрольные вопросы;
- срок выполнения работы.

Время выполнения работы – 4 академических часа.

Контрольные вопросы

1. Что означает предобучение нейронной сети и на каких датасетах обычно выполняется?
2. Какие преимущества даёт использование предобученной модели при обучении на новом датасете?
3. Что такое «замораживание слоев» и зачем оно нужно в контексте дообучения?
4. Какие части сети обычно замораживают, а какие оставляют обучаемыми при fine-tuning?
5. Почему иногда используют разные learning rate для разных слоев при дообучении предобученной сети?
6. Как меняется архитектура предобученной модели для новой задачи с другим числом классов?
7. Как можно оценить эффективность предобучения по сравнению с обучением модели с нуля?