Промежуточный отчет по проекту

Тема: Реализация модели по статье "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows"

Репозиторий проекта: https://github.com/RuslanGreenhead/swin_transformer/tree/main

Что уже сделано на данный момент:

- реализованы все структурные модули
- модули отлажены как по отдельности, так и в совокупности (ноутбук playground.ipynb). Так что, по сути дела, реализация сетки готова. Осталась реализация экспериментов.

Ключевые отличия моей реализации от авторской (список дополняемый, мог что-то упустить):

- во многих местах я использовал эйнштейновские нотации (einops.rearrange и torch.einsum). Из-за этого (на мой взгляд) некоторые операции получились более простыми и читаемыми (функции window_partition, window_merging)
- Windowed Attention (в коде - WMSA) написан так же через rearrange и einsum.
- Также во многих местах у меня отличается форма тензоров (я старался выдерживать «двумерную» структуру -> (windows, win_height, win_width, channels). В то время как авторы работают преимущественно со сглаженной формой -> (windows, win_height*win_width, channels)).
- - PatchMerging я сделал через свертку (а у авторов, честно говоря, какая-то чехарда..).
  (Вообще, на первых этапах работы, на стадии PatchEmbedding (когда я сначала патчил картинки, а затем навешивал линейное преобразование отдельным слоем) я был удивлен, когда оказалось, что авторы изящно реализовали обе эти операции через обычную свертку. И меня удивило, что в PatchMerging они так не сделали… Хотя тут как будто линейный downsampling и гораздо более прямой намек на свертку)
  P.S: Но есть шанс, что это просто я чего-то не понял
- Построение маски для attention у меня немного другое (и как будто более простое) – вынесено в функцию build_mask
- Я не прикручиваю к входному тезору (имею в виду тот, который выходит из PatchEmbedding) Absolute Position Encoding и Position Drop. Это есть в официальной реализации, но в статье про это почему-то ни слова. Поэтому я решил эти моменты опустить.

Что планируется сделать:

- Реализация экспериментов
  -- классификация на CIFAR (скорее в качестве тестирования и финального «отдебаживания»)
  -- классификация на ImageNet 1K
  -- детектирование на COCO
  (авторы еще сегментировали, но я не уверен, что на это хватит времени ..)
- И еще охота, конечно, поиграть с архитектурой. Из того, что приходило в голову – попробовать какой-нибудь другой downsampling.. Но тут я не знаю, насколько это перспективно и целесообразно.

(Комментарий: я работал у себя в колабе и собрал репозиторий пока только для отчета. Поэтому коммитов там мало)

Молоканов Р.А.