Accelerating Complex-Valued Tensors with `torch.compile`

Why we're doing it, how we're doing it, and what it means for you

OpenTeams

About Me

Hameer Abbasi

Senior Software Engineer I
OpenTeams, Inc.

- I’m relatively hairless at the moment.
- It’s not cancer, it’ll grow back.
- Enjoy it while you can.
- I love:
  - My boo
  - My cats
  - Cooking
  - Hiking
  - Wasting all my free time coding
- Pls make this presentation interactive? I can has questions?

OpenTeams

The Problem: Complex Tensors Can't Be Compiled

- `torch.Tensors` with a dtype of `torch.complex*` can't be used in a (full-graph) `torch.compile` context.
- This is one of the places where JAX/TensorFlow are superior in performance. See, for example, #125718.
- World models and rotary embeddings can be written a lot more simply with complex ops than the equivalent (decomposed) real ops.

OpenTeams

The Solution: A Subclass for Complex `torch.Tensors`

- Define a `torch.Tensor` subclass which stores the real and imaginary parts of the Tensor. #167621 Operates separately on the real and imaginary parts of the tensor, using the decomposition of a given operation.
- Tweak it to get gradcheck working. #168248

OpenTeams

The Solution: Rewrite Ops on Complex Tensors

- Rewrite ops on complex tensors to ops on the subclass. #169832
- Reflect the inputs and outputs at the edges of compiled blocks.

OpenTeams

The Caveat: How to Store The Constituents?

- Storing the constituent parts as separate tensors has the advantage of being able to use the built-in matmul units on the GPU.
- Mirroring the memory layout has the advantage of preserving view semantics between the input and the output.

OpenTeams

The Other Caveat: How to Deal with Complex* Ops?

- Must lower to specific code-paths per back-end for some ops, such as FFT.
- Inductor must allow backends to generate code for new ops.

OpenTeams

The Bonus: `torch.bcomplex32`

- There currently isn't any `torch.complex?` dtype with components being `torch.bfloat16`. So we made one in #17383.
- The idea is that this can initially only be used in a `torch.compile` context. Afterwards, maybe "eager" kernels can be JIT compiled on-demand.

OpenTeams

Thanks for your Ear

habbasi@openteams.com

www.openteams.com

hameerabbasi