

# धातु

नीचे दिये गये स्क्रिप्ट को `torch.backends.mps` का उपयोग करके `torch` (CPU) की प्रदर्शन क्षमता का परीक्षण करने के लिए का उपयोग किया जाता है।

```
import torch
import argparse
import time

parser = argparse.ArgumentParser(description="Test torch with MPS or CPU.")
parser.add_argument("--device", type=str, default="mps", choices=["mps", "cpu"], help="Device to use (mps or cpu)")
args = parser.parse_args()

if args.device == "mps":
    if torch.backends.mps.is_available():
        print("Metal is available")
        device = torch.device("mps")
    else:
        print("Metal is not available, using CPU instead")
        device = torch.device("cpu")
elif args.device == "cpu":
    device = torch.device("cpu")
    print("Using CPU")
else:
    print("Invalid device specified, using CPU instead")
    device = torch.device("cpu")

# Create a tensor and move it to the specified device
x = torch.randn(5000, 5000, device=device)
y = torch.randn(5000, 5000, device=device)

# Perform a more complex computation
start_time = time.time()
result = torch.matmul(x, y)
for _ in range(10):
    result = torch.matmul(result, y)
end_time = time.time()

# Print the result
print(result)
print(f"Time taken: {end_time - start_time:.4f} seconds")
```

नतीजे प्रदर्शित करते हैं कि `tf.nn.conv2d` से काफी तेज़ है। `tf.nn.conv2d` पर क्रियान्वयन की समय अवधि `tf.nn.conv2d` की समय अवधि का केवल 0.2% है।

```
% python scripts/test_metal.py --device cpu
```

```
Time taken: 2.8784 seconds
```

```
% python scripts/test_metal.py --device mps
```

```
Time taken: 0.0061 seconds
```