|  |  |
| --- | --- |
| **| 학습주제** | * PyTorch를 사용하여 텐서 연산(덧셈, 뺄셈, 곱셈) 및 In-place 연산 학습 |
| **| 학습목표** | * 3x3 크기의 랜덤 텐서를 생성하는 방법을 학습한다. * 텐서 간의 덧셈, 뺄셈, 곱셈 연산을 수행하는 능력을 기른다. * Out-of-place 방식과 In-place 방식의 차이를 이해하고, In-place 연산을 수행하는 방법을 학습한다. |
| **| 학습개념** | * torch.rand() 함수는 0과 1 사이의 난수를 포함한 텐서를 생성하는 함수 * Out-of-place연산은 결과가 새로운 텐서에 저장되는 연산 방식 * In-place연산은 결과가 기존 텐서에 바로 반영되는 연산 방식 * 텐서 연산은 덧셈, 뺄셈, 곱셈등의 연산을 수행 |

|  |  |
| --- | --- |
| **| 학습방향** | |
| **| 지문** | 이번엔 AI 모델 학습을 위한 데이터 전처리 단계에서 텐서 연산을 수행해야 합니다. 먼저, 3x3 크기의 랜덤 텐서를 생성한 후, 두 텐서 간의 덧셈, 뺄셈, 곱셈을 수행하세요. 또한, In-place 연산을 사용하여 연산 결과가 원본 텐서에 직접 반영되는지 확인하세요. |
| **| 절차** | |
|  | torch.rand() 함수를 사용하여 3x3 크기의 랜덤 텐서를 생성합니다.  두 텐서 간의 덧셈, 뺄셈, 곱셈 연산을 Out-of-place 방식으로 수행하고, 결과를 출력하세요.  add\_() 함수를 사용하여 In-place 덧셈 연산을 수행하고, 결과를 출력하세요. |
| **| 요구사항** | |
|  | torch.rand() 함수를 사용하여 3x3 크기의 랜덤 텐서를 두 개 생성하세요.  두 텐서의 덧셈, 뺄셈, 곱셈 연산을 Out-of-place 방식으로 수행하세요.  add\_() 함수를 사용하여 In-place 덧셈 연산을 수행하고, 결과를 출력하세요. |