



einsum

--	--	--	--	--	--

--	--	--	--	--

Free indices

Summation indices

-  $i \quad j$
-  k

--	--	--	--	--	--

- equation
 $a \otimes b \otimes k$
- equation
- equation "ik,kj->ij" "ik,kj->jl"
 einsum

--	--	--	--

--	--	--	--	--	--	--	--

- equation

"ik,kj"

"ij"
- equation

"..."

```
a = torch.randn(2,3,5,7,9)
# i = 7, j = 9
b = torch.einsum('...ij->...ji', [a])
```

--	--	--	--	--

bilinear transformation□

```
np_a = a.numpy()
np_b = b.numpy()
```

```

np_c = c.numpy()
np_out = np.empty((2, 5), dtype=np.float32)

np_out = torch.einsum('ik,jkl,il->ij', [a, b, c]).numpy()
# ik broadcast  ikl
# il broadcast  ikl
# 'ik,jkl,il->ij'  'ikl,jkl,ikl->ij'

for i in range(0, 2):
    for j in range(0, 5):
        #  ikl  k  l
        sum_result = 0
        for k in range(0, 3):
            for l in range(0, 7):
                sum_result += np_a[i, k] * np_b[j, k, l] * np_c[i, l]
        np_out[i, j] = sum_result

```



```

a = torch.rand(2,3)
b = torch.rand(3,4)
c = torch.einsum("ik,kj->ij", [a, b])
#  torch.mm(a, b)

```

equation

Revision #1

Created 11 January 2025 09:46:28 by Colin

Updated 3 March 2025 09:52:02 by Colin