MBATT模型在Python中的编码框架

|  |
| --- |
| MBATT(  (output): Linear(in\_features=3200, out\_features=2, bias=True)  (comp\_model): comp\_encoding(  (Comp\_EncoderLayer): DMPNNEncoderLayer(  (activation): ReLU()  (dropout): Dropout(p=0.0, inplace=False)  (W\_i): Linear(in\_features=147, out\_features=320, bias=False)  (W\_h): Linear(in\_features=320, out\_features=320, bias=False)  (W\_o): Linear(in\_features=453, out\_features=320, bias=True)  )  )  (bi\_att0): BIATT(  (U): ParameterList(  (0): Parameter containing: [torch.float32 of size 40x40]  (1): Parameter containing: [torch.float32 of size 40x40]  (2): Parameter containing: [torch.float32 of size 40x40]  (3): Parameter containing: [torch.float32 of size 40x40]  )  (transform\_c2p): ModuleList(  (0-3): 4 x Linear(in\_features=40, out\_features=40, bias=True)  )  (transform\_p2c): ModuleList(  (0-3): 4 x Linear(in\_features=40, out\_features=40, bias=True)  )  (bihidden\_c): ModuleList(  (0-3): 4 x Linear(in\_features=40, out\_features=40, bias=True)  )  (bihidden\_p): ModuleList(  (0-3): 4 x Linear(in\_features=40, out\_features=40, bias=True)  )  (biatt\_c): ModuleList(  (0-3): 4 x Linear(in\_features=80, out\_features=1, bias=True)  )  (biatt\_p): ModuleList(  (0-3): 4 x Linear(in\_features=80, out\_features=1, bias=True)  )  (comb\_c): Linear(in\_features=160, out\_features=40, bias=True)  (comb\_p): Linear(in\_features=160, out\_features=40, bias=True)  )  (bi\_att1): BIATT(  (U): ParameterList(  (0): Parameter containing: [torch.float32 of size 40x40]  (1): Parameter containing: [torch.float32 of size 40x40]  (2): Parameter containing: [torch.float32 of size 40x40]  (3): Parameter containing: [torch.float32 of size 40x40]  )  (transform\_c2p): ModuleList(  (0-3): 4 x Linear(in\_features=40, out\_features=40, bias=True)  )  (transform\_p2c): ModuleList(  (0-3): 4 x Linear(in\_features=40, out\_features=40, bias=True)  )  (bihidden\_c): ModuleList(  (0-3): 4 x Linear(in\_features=40, out\_features=40, bias=True)  )  (bihidden\_p): ModuleList(  (0-3): 4 x Linear(in\_features=40, out\_features=40, bias=True)  )  (biatt\_c): ModuleList(  (0-3): 4 x Linear(in\_features=80, out\_features=1, bias=True)  )  (biatt\_p): ModuleList(  (0-3): 4 x Linear(in\_features=80, out\_features=1, bias=True)  )  (comb\_c): Linear(in\_features=160, out\_features=40, bias=True)  (comb\_p): Linear(in\_features=160, out\_features=40, bias=True)  )  (torsion\_embed): Embedding(10, 1024)  ) |

注：相似度评价模型的编码框架与MBATT相同，除了将输出部分的(output): Linear(in\_features=3200, out\_features=2, bias=True) 改为(output): Identity()

MBATT模型训练全部权重参数

|  |
| --- |
| 'output.weight', 'output.bias', 'comp\_model.Comp\_EncoderLayer.W\_i.weight', 'comp\_model.Comp\_EncoderLayer.W\_h.weight', 'comp\_model.Comp\_EncoderLayer.W\_o.weight', 'comp\_model.Comp\_EncoderLayer.W\_o.bias', 'bi\_att0.c\_param', 'bi\_att0.p\_param', 'bi\_att0.U.0', 'bi\_att0.U.1', 'bi\_att0.U.2', 'bi\_att0.U.3', 'bi\_att0.transform\_c2p.0.weight', 'bi\_att0.transform\_c2p.0.bias', 'bi\_att0.transform\_c2p.1.weight', 'bi\_att0.transform\_c2p.1.bias', 'bi\_att0.transform\_c2p.2.weight', 'bi\_att0.transform\_c2p.2.bias', 'bi\_att0.transform\_c2p.3.weight', 'bi\_att0.transform\_c2p.3.bias', 'bi\_att0.transform\_p2c.0.weight', 'bi\_att0.transform\_p2c.0.bias', 'bi\_att0.transform\_p2c.1.weight', 'bi\_att0.transform\_p2c.1.bias', 'bi\_att0.transform\_p2c.2.weight', 'bi\_att0.transform\_p2c.2.bias', 'bi\_att0.transform\_p2c.3.weight', 'bi\_att0.transform\_p2c.3.bias', 'bi\_att0.bihidden\_c.0.weight', 'bi\_att0.bihidden\_c.0.bias', 'bi\_att0.bihidden\_c.1.weight', 'bi\_att0.bihidden\_c.1.bias', 'bi\_att0.bihidden\_c.2.weight', 'bi\_att0.bihidden\_c.2.bias', 'bi\_att0.bihidden\_c.3.weight', 'bi\_att0.bihidden\_c.3.bias', 'bi\_att0.bihidden\_p.0.weight', 'bi\_att0.bihidden\_p.0.bias', 'bi\_att0.bihidden\_p.1.weight', 'bi\_att0.bihidden\_p.1.bias', 'bi\_att0.bihidden\_p.2.weight', 'bi\_att0.bihidden\_p.2.bias', 'bi\_att0.bihidden\_p.3.weight', 'bi\_att0.bihidden\_p.3.bias', 'bi\_att0.biatt\_c.0.weight', 'bi\_att0.biatt\_c.0.bias', 'bi\_att0.biatt\_c.1.weight', 'bi\_att0.biatt\_c.1.bias', 'bi\_att0.biatt\_c.2.weight', 'bi\_att0.biatt\_c.2.bias', 'bi\_att0.biatt\_c.3.weight', 'bi\_att0.biatt\_c.3.bias', 'bi\_att0.biatt\_p.0.weight', 'bi\_att0.biatt\_p.0.bias', 'bi\_att0.biatt\_p.1.weight', 'bi\_att0.biatt\_p.1.bias', 'bi\_att0.biatt\_p.2.weight', 'bi\_att0.biatt\_p.2.bias', 'bi\_att0.biatt\_p.3.weight', 'bi\_att0.biatt\_p.3.bias', 'bi\_att0.comb\_c.weight', 'bi\_att0.comb\_c.bias', 'bi\_att0.comb\_p.weight', 'bi\_att0.comb\_p.bias', 'bi\_att1.c\_param', 'bi\_att1.p\_param', 'bi\_att1.U.0', 'bi\_att1.U.1', 'bi\_att1.U.2', 'bi\_att1.U.3', 'bi\_att1.transform\_c2p.0.weight', 'bi\_att1.transform\_c2p.0.bias', 'bi\_att1.transform\_c2p.1.weight', 'bi\_att1.transform\_c2p.1.bias', 'bi\_att1.transform\_c2p.2.weight', 'bi\_att1.transform\_c2p.2.bias', 'bi\_att1.transform\_c2p.3.weight', 'bi\_att1.transform\_c2p.3.bias', 'bi\_att1.transform\_p2c.0.weight', 'bi\_att1.transform\_p2c.0.bias', 'bi\_att1.transform\_p2c.1.weight', 'bi\_att1.transform\_p2c.1.bias', 'bi\_att1.transform\_p2c.2.weight', 'bi\_att1.transform\_p2c.2.bias', 'bi\_att1.transform\_p2c.3.weight', 'bi\_att1.transform\_p2c.3.bias', 'bi\_att1.bihidden\_c.0.weight', 'bi\_att1.bihidden\_c.0.bias', 'bi\_att1.bihidden\_c.1.weight', 'bi\_att1.bihidden\_c.1.bias', 'bi\_att1.bihidden\_c.2.weight', 'bi\_att1.bihidden\_c.2.bias', 'bi\_att1.bihidden\_c.3.weight', 'bi\_att1.bihidden\_c.3.bias', 'bi\_att1.bihidden\_p.0.weight', 'bi\_att1.bihidden\_p.0.bias', 'bi\_att1.bihidden\_p.1.weight', 'bi\_att1.bihidden\_p.1.bias', 'bi\_att1.bihidden\_p.2.weight', 'bi\_att1.bihidden\_p.2.bias', 'bi\_att1.bihidden\_p.3.weight', 'bi\_att1.bihidden\_p.3.bias', 'bi\_att1.biatt\_c.0.weight', 'bi\_att1.biatt\_c.0.bias', 'bi\_att1.biatt\_c.1.weight', 'bi\_att1.biatt\_c.1.bias', 'bi\_att1.biatt\_c.2.weight', 'bi\_att1.biatt\_c.2.bias', 'bi\_att1.biatt\_c.3.weight', 'bi\_att1.biatt\_c.3.bias', 'bi\_att1.biatt\_p.0.weight', 'bi\_att1.biatt\_p.0.bias', 'bi\_att1.biatt\_p.1.weight', 'bi\_att1.biatt\_p.1.bias', 'bi\_att1.biatt\_p.2.weight', 'bi\_att1.biatt\_p.2.bias', 'bi\_att1.biatt\_p.3.weight', 'bi\_att1.biatt\_p.3.bias', 'bi\_att1.comb\_c.weight', 'bi\_att1.comb\_c.bias', 'bi\_att1.comb\_p.weight', 'bi\_att1.comb\_p.bias', 'torsion\_embed.weight' |