A 6+T Parametrization

The algorithm followed for the distance-based loss function optimization is detailed below.

Algorithm 1 6+T parametrization used in UMD-fit

```
optim = LBFGS(params=[euler, trans, torsions], lr=0.5)
for i in range(max_steps)
    def closure():
      optim.zero_grad()
      coords = coords + trans
      rot_mat = rot_from_euler(euler)
      coords = einsum('rc,nc->nr', rot_mat, coords - com) + com
      aux_coords = coords.clone()
      for (i,j,k,l), value in zip(torsion_idxs, torsion_values):
        coords = update_dihedral(coords, idxs=[i,j,k,l], value=value)
      coords = kabsch(coords, aux_coords)
      loss = loss_func(coords, pocket_coords, pred_dist_mat)
      loss.backward()
      return loss
loss = optimizer.step(closure)
```

More details on the performance of $6 + T$ alone and the stereochemistry preservation in UMD-fit relative to Uni-Mol Dockingare shown in Table 1. The difference can be largely attributed to higher quality conformers in the distance optimization post-processing, but not improved modelling by the ligand processing module in Uni-Mol. We hypothesize this might be explained by incorrect training data (wrong stereochemistry) in original Uni-Mol training, thus yielding a model less sensitive to those features and specific geometries. Further work might involve retraining the original model with correct input features.

Table 1: Performance for different inference strategies for Uni-Mol Docking in the CASF-2016 test set

			$\% \leq$ symRMSD ($\uparrow$)					
Strategy	RMSD ($\downarrow$, Å)	symRMSD ($\downarrow$, Å)	0.5Å	1.0Å	1.5Å	2.0Å	3.0Å	5.0Å
Baseline	1.67	1.54	14.74	56.14	72.98	81.05	87.72	94.39
6+T	1.81	1.65	10.18	45.96	70.53	79.29	85.61	93.68
UMD-fit	1.75	1.57	12.63	51.58	72.28	80.70	87.72	94.03