Thanks to all reviewers for your constructive suggestions. Responses are as follows.

Major characteristics/advantages of the proposed approach:

- QuatE considers relations as rotations in four dimensional space. It firstly rotates the head entities then do semantic matching between the rotated head entity and the tail entity. QuatE is a generalization of ComplEx, it keeps all the benefits of ComplEx. We showed that quaternion rotations are especially helpful for the knowledge graph embedding.
- It can greatly save the number of parameters. This is more significant on datasets without trivial inverse relations. For example, it reduced the number of parameters by 80.1% on FB15K-237, 60% on WN18RR, compared to the latest state-of-the-art model(RotatE).

Comparison with ComplEx by controlling the number of parameters and negative samples. For datasets WN18RR and FB15K-237, the reported results of ComplEx are achieved with embeddings size 200 while QuatE use embedding size 100. The numbers of parameters are the same, but QuatE outperforms ComplEx largely. We also ran ComplEx on WN18 using the same number of parameters and negative samples as QuatE. As shown in Table 1, QuatE still performs better than ComplEx.

Table 1: Results of ComplEx and QuatE with same number of parameters and negative samples.

	WN18		WN18RR	
#Params	40.96M		16.38M	
#neg	10		1	
Measures	MRR	Hit@10	MRR	Hit@10
ComplEx	0.942	0.952	0.44	0.51
QuatE	0.950	0.959	0.488	0.582

Most baselines are exhaustively tuned. The hyper-parameters of baselines are already exhaustively tuned. For example, the number of negative samples in the original ComplEx model are tuned from $\{1, 2, 5, 10\}$. Some neural network-based methods even use dropout and label smoothing to improve their performance. For QuatE, the number of negative samples are 10(WN18), 20 (FB15K), 1(WN18RR), 10(FB15K-237). The size is fair compared with ComplEx. If we set #neg=10 for FB15K, we can get MRR=0.781, Hit@10=0.899.

Number of epochs. The number of epochs needed of QuatE and RotatE are shown in Table 2, despite that we use uniform sampling, and rotatE use adversarial negative sampling, our method needs much less number of epochs than RotatE.

Table 2: Number of epochs needed of QuatE and RotatE.

Datasets	WN18	WN18RR	FB15K	FB15K-237
QuatE	1500	40000	5000	15000
RotatE	80000	80000	150000	150000

Discussion on the composition patterns. Composition patterns are commonplaces in knowledge graphs. Here, we pointed out that fixing the composition function may lead to sub-optimal performances as there are many ways of relation compositions. Our model does not fix the composition pattern of the model. If r_3 composes of r_1 and r_2, both TransE and RotatE assume there are only one determinate composition functions ($r_3 = r_1 + r_2$ or $r_3 = r_1 \circ r_2$). In these two models, r_3 has nothing to do with the entities. In QuatE, the r_3 is not only determined by relations r_1 and r_2, but also the entity embeddings. As such, the composition patterns are not fixed to one form, instead, relation r_3 is not only determined by r_1 and r_2 but also simultaneously influenced by entity embeddings.

MRR for each relation on WN18RR. The overall MRR improvement on WN18RR is 0.470 ->0.488. QuatE get improvements on seven relations, and are on par or fail on other relations. Note that the number of samples for each relation is different. Thus the overall improvement is weighted by the number of samples of each relation.

We also found that the relation normalization can improve the ComplEx model as well. But it is till worse than QuatE. In this ablation study, we did not tune the hyper-parameters but using the same ones as standard QuatE. After some tests, we found that the initialization scheme is optional on these four datasets, random initialization can get the same performance. This initialization scheme might be useful for other datasets.