Rev#1: We thank the reviewer for the encouraging remarks.

Why the attention mechanism is only based on the Y series..... latent factors X,F?: This is a good observation indeed and we have thought of possible extensions in this direction. One method could be to treat the prediction from the local model $\hat{\mathbf{y}}_t^l$ and the predicted basis time-series $\mathbf{X}_t$ as covariates for predicting the actual values $\mathbf{y}_t$ and feed them into a temporal convolution network which is trained after the local and global models are trained.

Details about what the loss function is specifically to train the mean and residual forecasters in the local models?: These details are implicitly specified in Algorithm 1. The loss function used to train the mean forecaster is the normalized ℓ_1-loss (see eq. 1) between the predicted value and the mean of next w values of the original time-series. The residual forecaster is trained using the same loss, but with respect to the true residual values in the future time-range. We will add a text description in our revised version.

Rev#2: We thank the reviewer for the comments and suggestions.

Both the local and global models are not very original... : To the best of our knowledge, this is the first paper to propose a hybrid local and global model in the context of deep learning for time-series and we have a thorough discussion of prior work, in the paper. The global model is especially novel, as matrix factorization regularized by a temporal convolution network has not been attempted before. There are several differences from TRMF, such as training the factors and the network alternatingly through SGD, unlike TRMF. The local model also highlights the issue of normalization and diverse scales of different time-series, which is not commonly discussed in deep learning time-series papers.

The authors emphasize the difficulty of normalizing data and show that they are dealing with this problem ....: We would like to note that we emphasize the normalization issue as it is a commonly ignored. We propose the leveled network method, where the key idea is that after the predicted rolling means (from the leveling network) is subtracted, the remaining residual values have much less variation in scale and therefore the residual network can be trained more reliably. As an empirical evidence, we would like to point the reviewer to the unnormalized columns in Table 2, where we see that when the data is not normalized, the local only models like Temporal Conv., DeepAR, LSTM do not converge to a good solution at all, while the Local DLN model performs at par with the normalized versions. Moreover, we would like to point out that on the larger wiki dataset, the unnormalized versions of DeepGLO and local DLN models perform better than the normalized versions in several metrics. We agree with the reviewer that the residual values may also have variations among the time-series, however the variations are much less and therefore our proposed solution empirically works well in all the data-sets considered.

Organization of Section 4 and 5 and sizes of figures: We thank the reviewer for these suggestions and they will be incorporated into the revision of the paper.

Rev#3: We thank the reviewer for the comments and suggestions.

However, there is still a lack ....especially with regard to the specific roles of the global and local models....: In Table 2 from the paper, we separately provide the metrics from the hybrid DeepGLO model and the local only DLN model, which shows an improvement of DeepGLO over the local only model. In Table 1 below, we further provide the metrics from the global only DLN-MF model, in response to this question. We can see that the hybrid DeepGLO model is better than the local and global counterparts in all cases, thus proving that there is added value in having a hybrid model. We will add these additional results to the paper.

Algorithm	elec $n = 370$		traffic $n = 963$	
	Normalized	Unnormalized	Normalized	Unnormalized
DeepGLO	0.084/0.291/0.119	0.109/0.448/0.149	0.159/0.218/0.202	0.169/0.256/ 0.195
Local DLN	0.086/0.258/0.129	0.118/0.336/0.172	0.169/0.246/0.218	0.237/0.422/0.275
DLN-MF	0.255/0.687/0.449	0.349/0.696/0.539	0.247/0.281/0.291	0.176/0.234/0.203

Algorithm	PeMSD7(M)		
	MAE	MAPE (%)	RMSE
DeepGLO	3.81	**8.29**	**6.31**
STGCN(Cheb)	**3.57**	8.69	6.77
STGCN(1^{st})	3.79	9.12	7.03

Table 1: In the first table, we provide additional values for the global only DLN-MF part for DeepGLO. In the second table, we compare the DeepGLO model with the Spatio-Temporal Model from Yu. et al.

Comparison with Yu et al. Spatio-temporal Graph Convolutional Networks...: We have already cited the above paper. Following the request of the reviewer, we compare our DeepGLO model on the PeMSD7(M) dataset on the same test split on the same task (of predicting 45 min in to the future) as in the original paper. The results are in Table 1 and we can see that DeepGLO performs better in two metrics, even when DeepGLO does not have access to the weighted similarity graph, which is an additional input to the model in Yu et al. We will add these comparisons to our paper. In view of these new results, we hope that the reviewer reconsiders their rating of the paper.