

## A IMPLEMENTATION DETAILS OF TILDE-Q SUBLOSSES

As we discussed in Sec. 4.2, TILDE-Q consists of three sublosses:  $L_{a.shift}$ ,  $L_{phase}$ , and  $L_{amp}$ . Our design rationale for selecting these sublosses is described in Sec. 4.1. In this section, we describe the detailed connection between the sublosses and the design rationale (Eqs. 1, 2, and 3).

**Amplitude Shifting** Given two sets of points with the same length  $T$ ,  $X, X' \in \mathbb{R}^T$ , let us define their distance using the signed distance function  $g : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ . Then, for each point  $x, x'$  in set  $X, X'$ , we can define a point-wise distance set  $D$  with  $g$  as below:

$$D = [g(x_1, x'_1), \dots, g(x_T, x'_T)] = [d_1, \dots, d_T].$$

When we design  $L_{a.shift}$ , we have one main question: given an arbitrary  $X, X'$ , and  $g$ , how do we build a loss function that is invariant to *any* arbitrary gap  $k$ . In this work, we have reformulated this task from *ensuring equal gaps between all points* into *making uniform distribution of the gaps* (i.e.,  $\sum_i p_{d_i} \log p_{d_i}$  on the interval  $[1, T]$ ). Please note that we convert gaps into relative values since an absolute domain requires information for  $k$  for each data sample. Without loss of generality, we can say that this problem is equivalent to the problem of entropy maximization. Let us suppose that we convert the distance set  $D$  into the probability distribution by *Softmax* function,  $p_{d_i} = \text{Softmax}(d_i)$ . In this case, we can say that our optimization problem is maximize the entropy as below:

$$\text{maximize } L = \sum_{i=1}^T p_{d_i} \log p_{d_i},$$

which is well-known to have its global optima with  $\forall_{i \in [1, T]} p_{d_i} = \frac{1}{T}$ . Therefore, we formulate  $L_{a.shift}$  as Eq. 5, which satisfies Eq. 1. Please note that the noise robustness of  $L_{a.shift}$  relies on that of the signed distance function,  $g$ . Since  $L_{a.shift}$  requires computation of  $g$  and *Softmax*, it takes  $O(n)$  time for its computation.

**Phase Shifting** To discuss phase shifting and periodicity of a time-series, Fourier transform is an inevitable factor. However, in the real-world dataset, a few problems arise: 1) we are unaware of the frequencies and periodicity of the data itself, and 2) a direct use of Fourier coefficients may be biased by noise. During the design phase, we aim to solve these problems with  $L_{phase}$ . To extract the main flow of time-series data (i.e., the dominant periodicity or frequencies), we first define the dominant frequencies based on their statistical significance. Let  $X \in \mathbb{R}^T$  as an input signal. In the machine learning domain, researchers commonly suppose the input signal follows normal distribution  $X \sim \mathcal{N}(0, I)$ . Accordingly, its Fourier coefficients on frequency  $k$  is:

$$\mathcal{F}(X) = \sum_{n=1}^T x_n e^{-i2\pi kn/T} \sim \mathcal{N}(0, T).$$

After Fourier transform, we define  $k$  as a dominant frequency if  $k$  is greater than  $\sqrt{T}$ , which indicates statistical significance. However, in some cases, we have only a short sample to represent signals or a noisy signal that has no periodicity, which does not yield a statistically significant  $k$ . To prevent such cases, in  $L_{phase}$ , we guarantee that at least  $\sqrt{T}$  number of frequencies are selected as dominant frequencies.  $L_{phase}$  requires  $O(n \log n)$  time for its computation, which is inherited from complexity of Fast Fourier Transform.

**Uniform Amplification** Although effective,  $L_{phase}$  has two limitations: 1) it is not perfectly phase shifting invariant as it is optimized with Fourier coefficients, and 2) aforementioned two subloss terms still make no consideration for uniform amplification invariance. Inspired by Paparrizos & Gravano (2015), we utilize normalized correlation for the uniform amplification. Specifically, we normalize correlation  $R$  as follows:

$$R(\mathbf{X}, \mathbf{Y}) = \frac{\text{Corr}(\mathbf{X}, \mathbf{Y})}{\sqrt{\text{Corr}(\mathbf{X}, \mathbf{X}) \cdot \text{Corr}(\mathbf{Y}, \mathbf{Y})}},$$

where *Corr* is cross-correlation or auto-correlation, and  $R$  is normalized correlation. By using this term, we have uniform amplification invariant measure. We utilize  $L_{amp}$  as the subcomponent with small  $\gamma$ , since tolerance for the multiplication factor (i.e., uniform amplification) has greater influence than addition or phase shifting. As  $L_{phase}$ , by using fast Fourier transform,  $L_{amp}$  takes  $O(n \log n)$  time.

**TILDE-Q Design Rationale:**  $\alpha$  and  $\gamma$   $L_{a.shift}$  is built for amplitude shifting and designed to be effective with both periodic and nonperiodic signals. In contrast,  $L_{phase}$  handles uniform amplification and is tailored to perform optimally with periodic signals. Since  $L_{a.shift}$  and  $L_{phase}$  complement each other, we set  $\alpha$  to balance them. For example, a large  $\alpha$  value will work well for nonperiodic signals and will have less penalty for amplitude shifting. Additionally, we utilize  $L_{amp}$  as a subcomponent to calibrate the results (e.g.,  $\gamma = 0.01$ ). With this design, while preserving the shape-awareness of TILDE-Q, users can control specific invariances or conditions. For example, users can increase the value of  $\alpha$  to emphasize nonperiodic modeling when a dataset has no particular periodicity. This user-oriented objective setting is one of the strengths of TILDE-Q and increases its utility.

## B DETAILED EXPERIMENT SETUP

**Dataset** In our experiment, we utilize eight datasets – Synthetic, ECG5000, and Traffic dataset for the simple model (i.e., Sequence-to-Sequence Gated Recurrent Unit) and ETTm1, Electricity (i.e., ECL), Traffic, Weather, Exchange, and ILI for the eight recent time-series forecasting models. For each dataset, we describe some metadata of them and the experimental setting, including the input length  $n$  and prediction window  $L$ . Our implementation could be found in Anonymous Github<sup>1</sup>.

**Synthetic:** As Le Guen & Thome (2019) describe, the Synthetic dataset is an artificial dataset for measuring model performance on sudden changes (step functions) with an input signal composed of two peaks. The amplitude and temporal position of the two peaks are randomly selected. Then the selected position and amplitude of the step are determined by a peak position and amplitude. We use 500 time-series for training, 500 for validation, and 500 for testing. For the Synthetic dataset, we set the input length as  $n = 20$  and the prediction window as  $L = 40$ . The generation code is provided in DILATE Github<sup>2</sup>.

**ECG5000:** This dataset is originally a 20-hour long ECG (Electrocardiogram), downloaded from Physionet<sup>3</sup> and archived in UCR Time Series Classification Archive (Dau et al., 2019). The data is split by each heartbeat and processed in equal lengths (140). In the training, we use 500 for training, 500 for validation, and 4000 for testing. We take the first  $n = 84$  steps as input and predict the last  $L = 56$  steps.

**Traffic:** Traffic dataset is a collection of 48 months (2015-2016) hourly road occupancy rate (between 0 to 1) data from the California Department of Transportation<sup>4</sup>. For the GRU model, we utilize univariate series of the first sensor, a total of 17544 data points as Le Guen & Thome (2019) do. We set our problem as forecasting  $L = 24$  future occupancy rates with  $n = 168$  historical data (past week). We use 60% of the data for training, 20% for validation, and the rest for evaluation. For recent time-series forecasting models, we conduct multivariate time-series forecasting and adopt hyperparameter settings from Time-Series-Library Github<sup>5</sup>.

**ETT:** The ETT (Electricity TraNSformer Temperature) dataset, published by Zhou et al. (2021), is 2-year data collected from two separate counties in China, including ETTm1 dataset. Each data point has a target value of “oil temperature” and other 6 power load features. As Zhou et al. (2021) do, we split them into 12/4/4 months for the training/validation/testing. Detailed settings, such as the input and output length and hyperparameter setting, are based on the information at Time-Series-Library Github 5.

**ECL:** The ECL (Electricity Consuming Load) is a dataset recorded in kWh every 15 minutes from 2012 to 2014, for 321 clients. In our experiment, we split them into 15/3/4 months for the train/validation/test, as Zhou et al. (2021) do. Note that we use the same hyperparameter settings in the ETTm1 dataset.

**Weather:** Weather dataset contains 21 meteorological indicators from Weather Station of the Max Planck Biogeochemistry Institute in 2020 with 10-minute interval.

<sup>1</sup><https://anonymous.4open.science/r/TILDE-Q-9E54>

<sup>2</sup><https://github.com/vincent-leguen/DILATE>

<sup>3</sup><https://physionet.org/>

<sup>4</sup><http://pems.dot.ca.gov>

<sup>5</sup><https://github.com/thuml/Time-Series-Library>

Table 3: Experimental results on six real-world datasets. We compared TILDE-Q and MSE by conducting extensive experiments with eight models. For all baselines, we set input sequence length  $T = 96$  except ILI dataset. For ILI dataset, we set input sequence length  $T = 36$ . Avg. means the average results from all four prediction lengths. We have colored the best training metric in **red**.

Models	iTransformer				PatchTST				Crossformer				TimesNet				DLinear				FEDformer				NSformer				Autoformer				
Methods	MSE	MSE	TILDE-Q	MSE	MSE	MSE	TILDE-Q	MSE	MSE	MSE	MSE	TILDE-Q	MSE	MSE	MSE	MSE	TILDE-Q	MSE	MSE	MSE	MSE	TILDE-Q	MSE	MSE	MSE	MSE	MSE	MSE	MSE	MSE	MSE		
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	
ETTS	96	0.344	0.378	<b>0.334</b>	<b>0.365</b>	0.323	0.363	<b>0.318</b>	<b>0.351</b>	0.361	0.407	<b>0.344</b>	<b>0.380</b>	<b>0.338</b>	0.379	0.343	<b>0.377</b>	0.346	0.373	<b>0.336</b>	<b>0.360</b>	0.384	0.420	<b>0.364</b>	<b>0.410</b>	0.420	0.416	<b>0.404</b>	<b>0.407</b>	0.506	0.482	<b>0.452</b>	<b>0.447</b>
	192	0.381	0.394	<b>0.380</b>	<b>0.390</b>	<b>0.368</b>	0.389	<b>0.372</b>	<b>0.389</b>	0.423	0.449	<b>0.395</b>	<b>0.413</b>	0.399	0.407	<b>0.389</b>	<b>0.400</b>	0.382	0.392	<b>0.380</b>	<b>0.386</b>	0.440	0.452	<b>0.442</b>	<b>0.443</b>	0.488	0.446	<b>0.474</b>	<b>0.439</b>	0.609	0.518	<b>0.596</b>	<b>0.513</b>
	336	0.418	0.416	<b>0.416</b>	<b>0.415</b>	0.405	0.405	<b>0.397</b>	<b>0.404</b>	0.612	0.575	<b>0.436</b>	<b>0.447</b>	0.423	0.425	<b>0.411</b>	<b>0.419</b>	0.414	0.413	<b>0.415</b>	<b>0.410</b>	0.495	0.476	<b>0.468</b>	<b>0.468</b>	0.585	0.444	<b>0.553</b>	<b>0.498</b>	0.635	0.534	<b>0.553</b>	<b>0.490</b>
	720	0.488	0.487	<b>0.481</b>	<b>0.450</b>	<b>0.458</b>	0.444	<b>0.461</b>	<b>0.443</b>	0.742	0.652	<b>0.533</b>	<b>0.517</b>	0.498	0.464	<b>0.462</b>	<b>0.447</b>	0.475	0.453	<b>0.473</b>	<b>0.445</b>	0.519	0.490	<b>0.496</b>	<b>0.482</b>	0.609	0.535	<b>0.638</b>	<b>0.530</b>	0.593	0.528	<b>0.562</b>	<b>0.509</b>
	Avg.	0.408	0.412	<b>0.403</b>	<b>0.405</b>	<b>0.387</b>	0.400	<b>0.387</b>	<b>0.397</b>	<b>0.353</b>	0.521	<b>0.427</b>	<b>0.439</b>	0.415	0.419	<b>0.401</b>	<b>0.411</b>	0.404	0.408	<b>0.401</b>	<b>0.404</b>	0.461	0.459	<b>0.439</b>	<b>0.451</b>	0.533	0.470	<b>0.512</b>	<b>0.465</b>	0.586	0.516	<b>0.541</b>	<b>0.492</b>
Electricity	96	0.148	0.240	<b>0.146</b>	<b>0.238</b>	0.181	0.270	0.181	0.261	0.147	0.248	<b>0.146</b>	<b>0.245</b>	0.167	0.270	<b>0.166</b>	<b>0.269</b>	0.210	0.302	<b>0.199</b>	<b>0.278</b>	<b>0.196</b>	0.310	<b>0.197</b>	<b>0.308</b>	<b>0.166</b>	<b>0.269</b>	0.172	0.274	0.201	0.316	0.200	<b>0.313</b>
	192	0.166	0.255	<b>0.164</b>	<b>0.255</b>	0.187	0.276	<b>0.186</b>	<b>0.267</b>	0.162	0.261	<b>0.161</b>	<b>0.260</b>	0.185	0.286	<b>0.184</b>	<b>0.284</b>	0.210	0.305	<b>0.197</b>	<b>0.279</b>	0.211	0.323	<b>0.210</b>	0.318	0.288	<b>0.185</b>	<b>0.286</b>	0.224	0.335	0.224	<b>0.331</b>	
	336	0.178	0.271	<b>0.176</b>	<b>0.270</b>	0.203	0.291	<b>0.201</b>	<b>0.282</b>	0.193	0.291	<b>0.181</b>	<b>0.282</b>	0.204	0.305	<b>0.200</b>	<b>0.300</b>	0.223	0.319	<b>0.208</b>	<b>0.292</b>	0.238	0.348	<b>0.228</b>	<b>0.339</b>	0.202	0.302	<b>0.195</b>	<b>0.297</b>	0.252	0.355	0.242	<b>0.346</b>
	720	0.225	0.310	<b>0.214</b>	<b>0.301</b>	0.245	0.325	<b>0.242</b>	<b>0.316</b>	0.250	0.334	<b>0.230</b>	<b>0.324</b>	0.224	0.320	<b>0.216</b>	<b>0.314</b>	0.258	0.350	<b>0.244</b>	<b>0.325</b>	0.263	0.367	<b>0.260</b>	<b>0.364</b>	0.227	0.322	<b>0.294</b>	<b>0.316</b>	0.324	0.396	<b>0.263</b>	<b>0.363</b>
	Avg.	0.179	0.269	<b>0.175</b>	<b>0.266</b>	0.204	0.291	<b>0.203</b>	<b>0.282</b>	0.188	0.284	<b>0.181</b>	<b>0.278</b>	0.195	0.295	<b>0.192</b>	<b>0.292</b>	0.225	0.319	<b>0.210</b>	<b>0.294</b>	0.227	0.330	<b>0.225</b>	<b>0.332</b>	0.196	0.295	<b>0.194</b>	<b>0.293</b>	0.250	0.351	<b>0.232</b>	<b>0.338</b>
Traffic	96	<b>0.395</b>	<b>0.268</b>	0.401	0.270	0.544	0.359	<b>0.499</b>	<b>0.323</b>	0.522	0.290	<b>0.515</b>	<b>0.280</b>	0.593	0.321	<b>0.576</b>	<b>0.311</b>	0.606	0.429	<b>0.709</b>	<b>0.428</b>	<b>0.587</b>	0.366	<b>0.595</b>	<b>0.337</b>	0.622	0.346	<b>0.614</b>	<b>0.340</b>	0.629	0.383	<b>0.597</b>	<b>0.372</b>
	192	<b>0.417</b>	<b>0.276</b>	0.421	0.278	0.540	0.354	<b>0.493</b>	<b>0.319</b>	0.530	0.293	<b>0.534</b>	<b>0.290</b>	0.617	0.336	<b>0.582</b>	<b>0.325</b>	0.647	0.336	<b>0.686</b>	<b>0.336</b>	0.666	0.379	<b>0.698</b>	<b>0.378</b>	0.643	0.336	<b>0.628</b>	<b>0.349</b>	0.637	0.403	<b>0.622</b>	<b>0.392</b>
	336	0.433	0.283	<b>0.427</b>	<b>0.280</b>	0.551	0.358	<b>0.510</b>	<b>0.337</b>	0.558	0.305	<b>0.550</b>	<b>0.298</b>	0.629	0.336	<b>0.615</b>	<b>0.333</b>	0.659	0.336	<b>0.686</b>	<b>0.336</b>	0.676	0.379	<b>0.686</b>	<b>0.379</b>	0.643	0.336	<b>0.628</b>	<b>0.349</b>	0.637	0.403	<b>0.622</b>	<b>0.392</b>
	720	0.467	0.302	<b>0.454</b>	<b>0.294</b>	0.586	0.375	<b>0.552</b>	<b>0.362</b>	0.589	0.328	<b>0.562</b>	<b>0.316</b>	0.640	0.350	<b>0.627</b>	<b>0.341</b>	0.689	0.424	<b>0.680</b>	<b>0.406</b>	0.626	0.382	<b>0.619</b>	<b>0.377</b>	0.667	0.365	<b>0.662</b>	<b>0.362</b>	0.658	0.404	<b>0.644</b>	<b>0.397</b>
	Avg.	0.428	0.282	<b>0.426</b>	<b>0.281</b>	0.555	0.362	<b>0.514</b>	<b>0.335</b>	0.550	0.304	<b>0.540</b>	<b>0.296</b>	0.620	0.336	<b>0.600</b>	<b>0.328</b>	0.672	0.418	<b>0.667</b>	<b>0.399</b>	0.610	0.378	<b>0.666</b>	<b>0.376</b>	0.644	0.355	<b>0.637</b>	<b>0.351</b>	0.637	0.395	<b>0.619</b>	<b>0.386</b>
Weather	96	0.177	0.218	<b>0.174</b>	<b>0.213</b>	0.177	0.216	<b>0.174</b>	<b>0.215</b>	0.158	0.230	<b>0.157</b>	<b>0.226</b>	0.172	0.220	<b>0.171</b>	<b>0.217</b>	0.200	0.256	<b>0.196</b>	<b>0.243</b>	0.217	0.260	<b>0.216</b>	<b>0.248</b>	0.189	0.238	<b>0.179</b>	<b>0.224</b>	0.274	0.336	0.239	<b>0.341</b>
	192	0.223	0.256	<b>0.220</b>	<b>0.244</b>	0.225	0.258	<b>0.221</b>	<b>0.257</b>	0.206	0.277	<b>0.200</b>	<b>0.260</b>	0.219	0.261	<b>0.210</b>	<b>0.255</b>	0.240	0.295	<b>0.237</b>	<b>0.284</b>	0.275	0.316	<b>0.276</b>	<b>0.316</b>	0.268	0.298	<b>0.263</b>	<b>0.291</b>	0.330	0.378	<b>0.323</b>	<b>0.364</b>
	336	0.280	0.299	<b>0.277</b>	<b>0.293</b>	0.285	0.301	<b>0.279</b>	<b>0.298</b>	0.272	0.335	<b>0.270</b>	<b>0.319</b>	0.280	0.306	<b>0.284</b>	<b>0.300</b>	0.285	0.332	<b>0.282</b>	<b>0.324</b>	0.339	0.380	<b>0.330</b>	<b>0.363</b>	0.351	0.338	<b>0.367</b>	<b>0.350</b>	0.389	0.410	<b>0.355</b>	<b>0.382</b>
	720	0.359	0.350	<b>0.356</b>	<b>0.347</b>	0.360	0.350	<b>0.357</b>	<b>0.349</b>	0.398	0.418	<b>0.363</b>	<b>0.399</b>	0.365	0.359	<b>0.359</b>	<b>0.354</b>	0.384	0.384	<b>0.347</b>	<b>0.372</b>	0.402	0.428	<b>0.439</b>	<b>0.405</b>	0.441	0.418	<b>0.433</b>	<b>0.402</b>	0.469	0.458	<b>0.448</b>	<b>0.448</b>
	Avg.	0.260	0.281	<b>0.257</b>	<b>0.274</b>	0.262	0.281	<b>0.258</b>	<b>0.280</b>	0.259	0.315	<b>0.248</b>	<b>0.301</b>	0.259	0.287	<b>0.256</b>	<b>0.282</b>	0.268	0.317	<b>0.266</b>	<b>0.306</b>	0.309	0.360	<b>0.302</b>	<b>0.342</b>	0.312	0.323	<b>0.310</b>	<b>0.317</b>	0.366	0.396	<b>0.364</b>	<b>0.376</b>
Exchange	96	0.092	0.214	<b>0.089</b>	<b>0.209</b>	0.086	0.204	<b>0.083</b>	<b>0.196</b>	0.071	0.377	<b>0.237</b>	<b>0.236</b>	0.105	0.234	<b>0.104</b>	<b>0.233</b>	0.078	0.200	<b>0.077</b>	<b>0.194</b>	0.160	0.290	<b>0.152</b>	<b>0.279</b>	0.144	0.265	<b>0.133</b>	<b>0.258</b>	0.172	0.302	0.154	<b>0.279</b>
	192	0.185	<b>0.308</b>	0.184	0.308	0.181	0.302	<b>0.177</b>	<b>0.296</b>	0.515	0.533	<b>0.451</b>	<b>0.495</b>	0.214	0.337	<b>0.202</b>	<b>0.328</b>	0.166	0.301	<b>0.163</b>	<b>0.295</b>	0.282	0.385	<b>0.277</b>	<b>0.378</b>	0.268	0.370	<b>0.236</b>	<b>0.350</b>	0.318	0.413	<b>0.271</b>	<b>0.378</b>
	336	0.366	0.438	<b>0.345</b>	<b>0.426</b>	<b>0.329</b>	<b>0.416</b>	<b>0.337</b>	<b>0.422</b>	1.239	0.873	<b>1.037</b>	<b>0.779</b>	0.365	0.439	<b>0.360</b>	<b>0.430</b>	0.298	0.410	<b>0.259</b>	<b>0.384</b>	0.484	0.512	<b>0.457</b>	<b>0.497</b>	1.402	0.507	<b>1.405</b>	<b>0.485</b>	1.513	0.534	<b>0.488</b>	<b>0.516</b>
	720	0.914	0.724	<b>0.895</b>	<b>0.715</b>	0.864	0.718	<b>0.857</b>	<b>0.698</b>	1.745	1.060	<b>1.606</b>	<b>1.014</b>	<b>0.926</b>	0.733	<b>0.962</b>	<b>0.759</b>	0.940	0.659	<b>0.703</b>	<b>0.643</b>	1.288	0.873	<b>1.209</b>	<b>0.844</b>	1.302	0.811	<b>1.078</b>	<b>0.733</b>	1.126	0.822	<b>1.061</b>	<b>0.799</b>
	Avg.	0.389	0.421	<b>0.379</b>	<b>0.415</b>	0.365	0.410	<b>0.364</b>	<b>0.403</b>	0.943	0.711	<b>0.833</b>	<b>0.660</b>	<b>0.403</b>	0.436	<b>0.407</b>	<b>0.438</b>	0.323	0.392	<b>0.301</b>	<b>0.373</b>	0.554	0.515	<b>0.524</b>	<b>0.499</b>	0.546	0.488	<b>0.474</b>	<b>0.457</b>	0.532	0.518	<b>0.494</b>	<b>0.493</b>
ILI	24	2.551	1.023	<b>2.552</b>	<b>0.956</b>	2.308	0.954	<b>2.257</b>	<b>0.896</b>	3.541	1.249	<b>3.000</b>	<b>1.208</b>	2.007	0.942	<b>1.753</b>	<b>0.872</b>	2.858	1.160	<b>2.771</b>	<b>1.134</b>	3.567	1.355	<b>1.477</b>	<b>1.253</b>	2.453	0.981	<b>1.708</b>	<b>0.860</b>	2.957	1.324	<b>1.577</b>	<b>1.388</b>
	36	2.237	0.964	<b>2.206</b>	<b>0.950</b>	2.333	0.926	<b>2.280</b>	<b>0.929</b>	3.615	1.251	<b>3.170</b>	<b>1.168</b>	2.752	1.005	<b>2.447</b>	<b>0.942</b>	2.689	1.110	<b>2.526</b>	<b>1.089</b>	3.553	1.331	<b>3.259</b>	<b>1.248</b>	2.959	1.060	<b>2.220</b>	<b>0.957</b>	3.899	1.274	<b>3.153</b>	<b>1.269</b>
	48	2.290	0.975	<b>2.173</b>	<b>0.935</b>	2.307	0.935	<b>2.080</b>	<b>0.890</b>	3.698	1.278	<b>3.187</b>	<b>1.180</b>	2.239	0.880	<b>2.814</b>	<b>1.151</b>	2.721	1.016	<b>2.979</b>	<b>1.196</b>	2.979	1.196	<b>2.563</b>	<b>1.176</b>	2.502	1.025	<b>2.220</b>	<b>0.923</b>	4.000	1.234	<b>2.973</b>	<b>1.159</b>
	60	2.255	0.975	<b>2.147</b>	<b>0.954</b>	2.064	0.917	<b>1.954</b>	<b>0.896</b>	4.043	1.344	<b>3.529</b>	<b>1.253</b>	2.092	0.942	<b>1.892</b>	<b>0.898</b>	2.900	1.178	<b>3.331</b>	<b>1.303</b>	3.127	1.122	<b>3.054</b>	<b>1.222</b>	2.446	1.021	<b>2.078</b>	<b>0.942</b>	3.163	1.210	<b>2.930</b>	<b>1.197</b>
	Avg.	2.333	0.984	<b>2.220</b>	<b>0.949</b>	2.253	0.933	<b>2.143</b>	<b>0.903</b>	3.724	1.281	<b>3.297</b>	<b>1.202</b>	2.346	0.963	<b>2.083</b>	<b>0.899</b>	2.815	1.150	<b>2.475</b>	<b>1.071</b>	3.123	1.173	<b>3.131</b>	<b>1.225</b>	2.631	1.024	<b>2.032</b>	<b>0.921</b>	3.927			

**Exchange:** Exchange (Wu et al., 2021) collects the panel data of daily exchange rates from 8 countries from 1990 to 2016. We follow the hyperparameter settings from author’s official codes.

**Illness:** Illness dataset contains the influenza-like illness patients in the United states in a weekly frequency.

**Deep Learning Model Architectures** We perform experiments with three different model architectures, including Sequence-to-Sequence GRU, Informer, and N-Beats. To induce models to predict future time-series in a timely manner, we set  $\alpha = 0.5$  and  $\gamma = 0.01$  for TILDE-Q. Other training metrics, including MSE and DILATE, are used as described in their original papers. All models are trained with Early Stopping and ADAM optimizer. To evaluate TILDE-Q in simple model, we utilize one layer Sequence-to-Sequence GRU model. For the training of the GRU model, we set a learning rate of  $1e-3$ , hidden size of 128, trained by maximum 1000 epochs with Early Stopping and ADAM optimizer. In addition to the basic GRU model, we conduct experiments with eight state-of-the-arts models as follows: 1) Autoformer (Wu et al., 2021), the first model that utilize frequency-domain information with autocorrelation module, 2) FEDformer (Zhou et al., 2022), an advanced version of Autoformer by introducing frequency-enhanced decomposition module, 3) nonstationary transformer (NSformer in this paper), which introduces series stationarization and de-stationary attention to resolve non-stationary characteristics of real-world time-series, 4) DLinear (Zeng et al., 2023), a simple linear model with input decomposition, tackling the importance of understanding of time-series, 5) TimesNet (Wu et al., 2023), a model to discover multiple periods and capture temporal 2D-variations with TimesBlock module, 6) Crossformer (Zhang & Yan, 2023), a first model to handle cross-dimension dependency for MTS forecasting, 7) PatchTST, a model patching time-series data for the better embedding, and 8) iTransformer (Liu et al., 2023), embed single time-series to the vector to preserving its temporal information.

## C ADDITIONAL EVALUATIONS

### C.1 DETAILED EXPERIMENT RESULTS AND ANALYSIS

At first, we observe that the model optimized with TILDE-Q outperforms the same model optimized with other objective functions in both short- and long-term forecasting tasks. However, each model has a wide range of TILDE-Q’s performance improvement caused by their design. In this section, we describe our findings related to both model design and the difficulty of the tasks. In the case of Autoformer and NSformer, we could observe that their performance improvements are signifi-

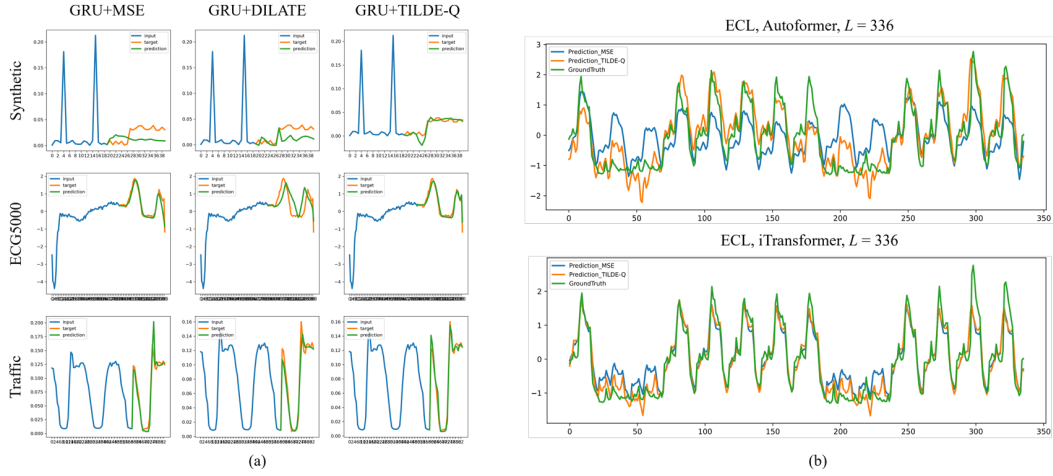


Figure 3: Qualitative results with simple sequence-to-sequence GRU model (a) and state-of-the-art model (b).

cantly higher than that of FEDformer or TimesNet. In design, Autoformer makes predictions with autocorrelation, which has a limitation in being aware of frequency-domain features than that of FEDformer or TimesNet. In the case of the NSformer, it utilizes series stationarization, which could be helpful to remove possible side effects from TILDE-Q, such as an equal gap caused by  $L_{a.shift}$ . This difference caused by design indicates two facts: 1) TILDE-Q could help the unseen feature modeling with its shape-awareness, as in Autoformer’s case, and 2) the model design influences the loss function, as NSformer helps the optimization with TILDE-Q. Crossformer is one good example with 8.85% improvement. For the Crossformer, which aims to resolve inter-domain dependency but has less attention to the temporal dependencies, TILDE-Q improves Crossformer’s temporal dependency modeling, and Crossformer’s inter-domain modeling supports TILDE-Q’s limitation for the inter-domain modeling.

In case of the dataset, Electricity or Traffic datasets makes the lowest improvements among all datasets. This is caused by task difficulty—for example, in case of the Autoformer, Electricity tasks with prediction lengths  $T' = \{96, 192, 336\}$ , Autoformer can solve the tasks without support for the shape-awareness. However, in case of the Electricity with prediction length 720, a relatively challenging task, Autoformer with MSE struggles to properly solve the task. This problem is also observable in Crossformer, which has relatively limited ability for temporal dependency modeling than the other models.

Next, we present a qualitative analysis of the results. Fig. 3 shows how the model with different training metrics forecasts with different datasets. From the figure, we have noticed that TILDE-Q allows the model to generate more robust, shape-aware forecasting, regardless of amplitude shifting, phase shifting, and uniform amplification. For example, in the case of Autoformer (Fig. 3 (b) top), TILDE-Q helps the model to handle multiple periodicity, which is not achieved with MSE (blue line). In contrast, Autoformer trained with MSE predicts only a single periodicity, indicating the limitation of MSE on shape-awareness. The strength of TILDE-Q is also observable in the iTransformer (Fig. 3 (b), bottom). Even when the model could make multiple periodicity modeling, TILDE-Q makes it more precise. In summary, TILDE-Q proves that it is model-agnostic, noise-robust, and shape-aware loss function and is far beneficial for the time-series forecasting.

## C.2 QUALITATIVE RESULTS WITH VISUALIZATION

To provide a clear comparison for MSE and TILDE-Q among different datasets and models, we list supplementary forecasting results of four representative datasets in Fig 4– 6. We provide qualitative results with six models—Autoformer (Wu et al., 2021), DLinear (Zeng et al., 2023), TimesNet (Wu et al., 2023), Crossformer (Zhang & Yan, 2023), PatchTST (Nie et al., 2023), and iTransformer (Liu

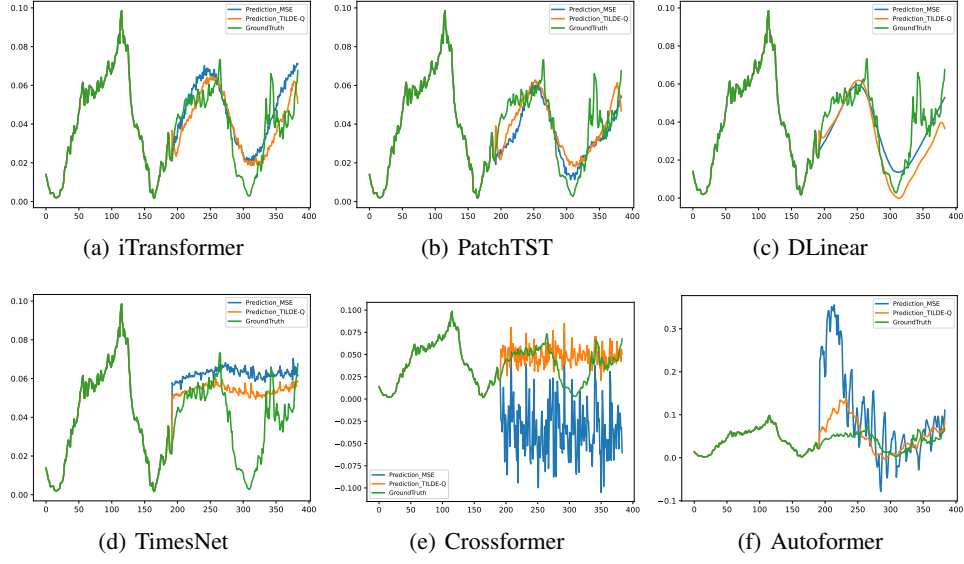


Figure 4: Qualitative Example with input-96-output-96 results on Weather dataset.

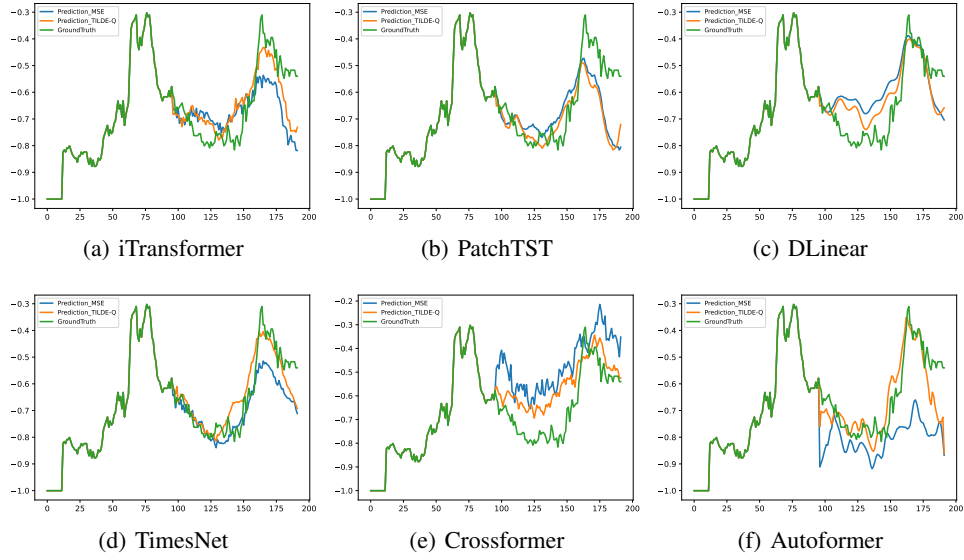


Figure 5: Qualitative Example with input-96-output-96 results on ETTm1 dataset.

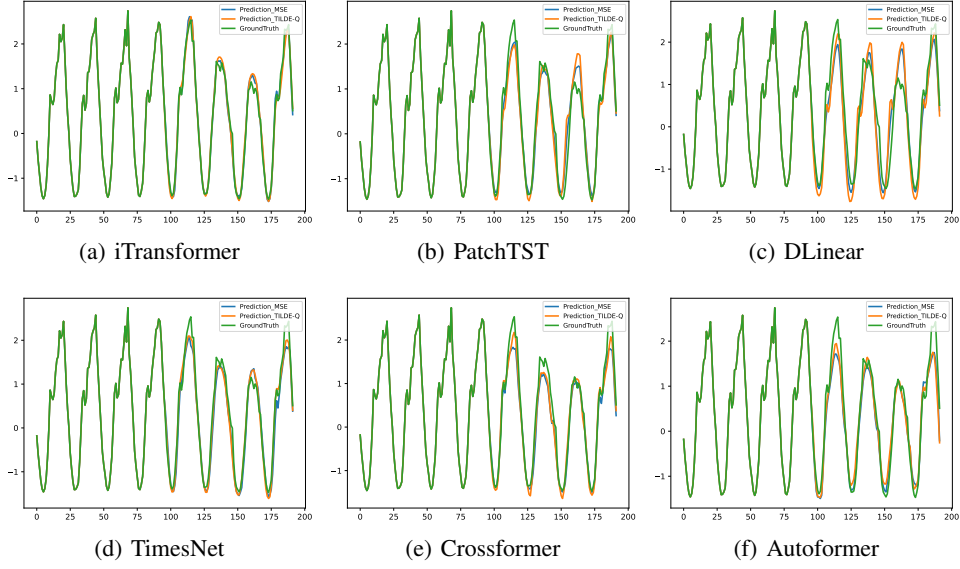


Figure 6: Qualitative Example with input-96-output-96 results on Weather dataset.

et al., 2023). Among various models and datasets, TILDE-Q shows its superior performance than MSE baseline.

Table 4: Ablation study with varying  $\alpha$  and  $\gamma$  on ETT\*, Crossformer, and 96-I-{192,720}-O settings.

Methods		TILDE-Q		$\alpha = 0.2$		$\alpha = 0.8$		$\gamma = 0.1$		$L_{a.shift}$ Only		$L_{phase}$ Only		$L_{amp}$ Only	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETT*	192	0.395	0.413	0.393	0.418	0.418	0.425	0.403	0.417	0.744	0.681	0.402	0.423	10.4	10.6
	720	0.533	0.517	0.546	0.528	0.531	0.508	0.571	0.550	0.818	0.689	0.546	0.525	2.57	2.56