SEED: An Effective Model for Highly-Skewed Streamflow Time Series Data Forecasting

Yanhong Li, Jack Xu, **David C. Anastasiu**

Santa Clara University

Problem: predicting long-term streamflow values with rain off data

$$[x_1, x_2, \ldots, x_T] \in \mathbb{R}^T \rightarrow [x_{T+1}, \ldots, x_{T+H}] \in \mathbb{R}^H,$$

x_1 to x_T : the input sequence

x_{T+1} to x_{T+H} : the output sequence

*In our research: H = 3 * 24 * 4 = 288, with majority of normal values and much fewer extreme values which cause the data skewness to one side.*

Challenges:

- Long-range dependencies.
- Rare but important extreme values; very imbalanced data.

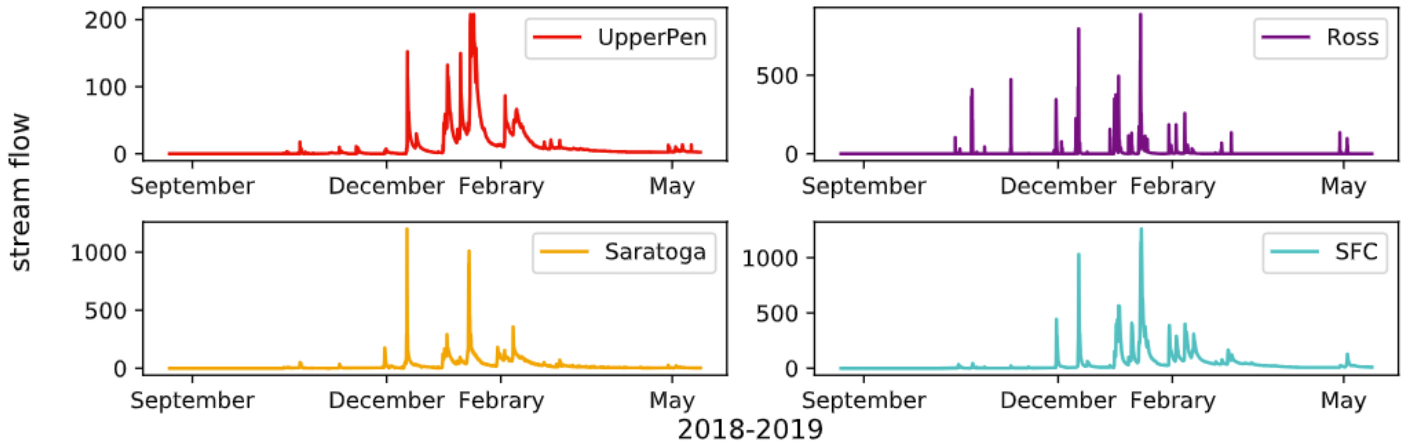
Goal:

- An end-to-end extreme-adaptive model;
- Long sequence forecasting *(predicted length = 288)*;

Dataset:

- Four groups of hydrologic datasets from Santa Clara County, CA.
- Namely Ross, Saratoga, UpperPen, and SFC, named after their respective locations.

Dataset with high skewness and kurtosis scores:



Four streams: Ross, Saratoga, UpperPen, and SFC.

Hydro year: from September to May.

High skewness and kurtosis scores indicate that there is significant deviation from a normal distribution in our data!

Statistic / Stream	Ross	Saratoga	UpperPen	SFC
mean	2.91	5.77	6.66	20.25
max	1440.00	2210.00	830.00	7200.00
min	0.00	0.00	0.00	0.00
median	0.17	1.00	3.20	1.20
variance	597.22	711.09	452.90	12108.14
skewness	19.84	19.50	13.42	18.05
kurtosis	523.16	697.78	262.18	555.18

Motivation: achieving the best overall prediction performance, without sacrificing either the quality of normal or of extreme predictions.

Root Mean Square Error ***(RMSE)***

Mean Absolute Percentage Error ***(MAPE)***

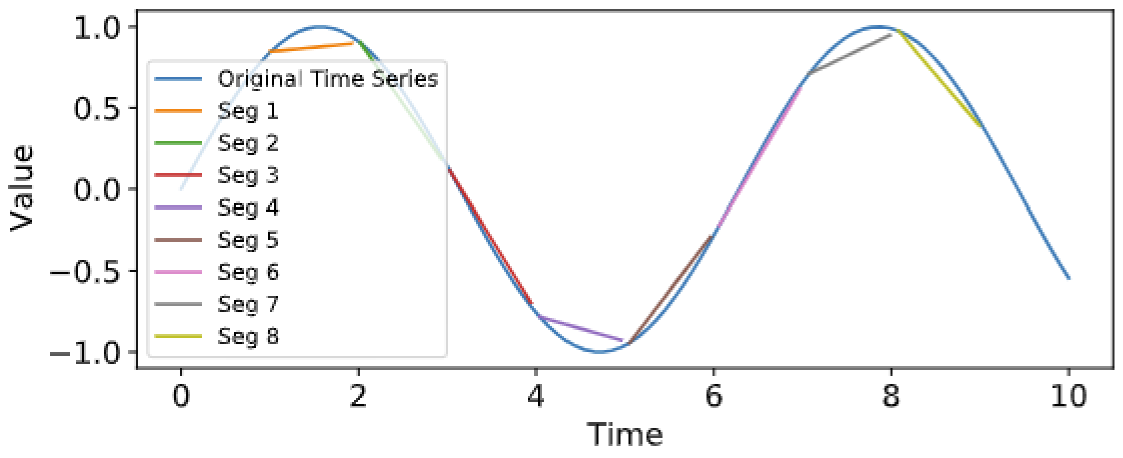
Proposed Methods:

Framework: Segment-Expandable Encoder-Decoder (SEED) model, which is the first to integrate segment representation learning with a multi-tiered encoder-decoder framework.

Importance-enhanced sampling strategy: embedded within the SEED model, allowing it to skillfully identify key features and trends in datasets.

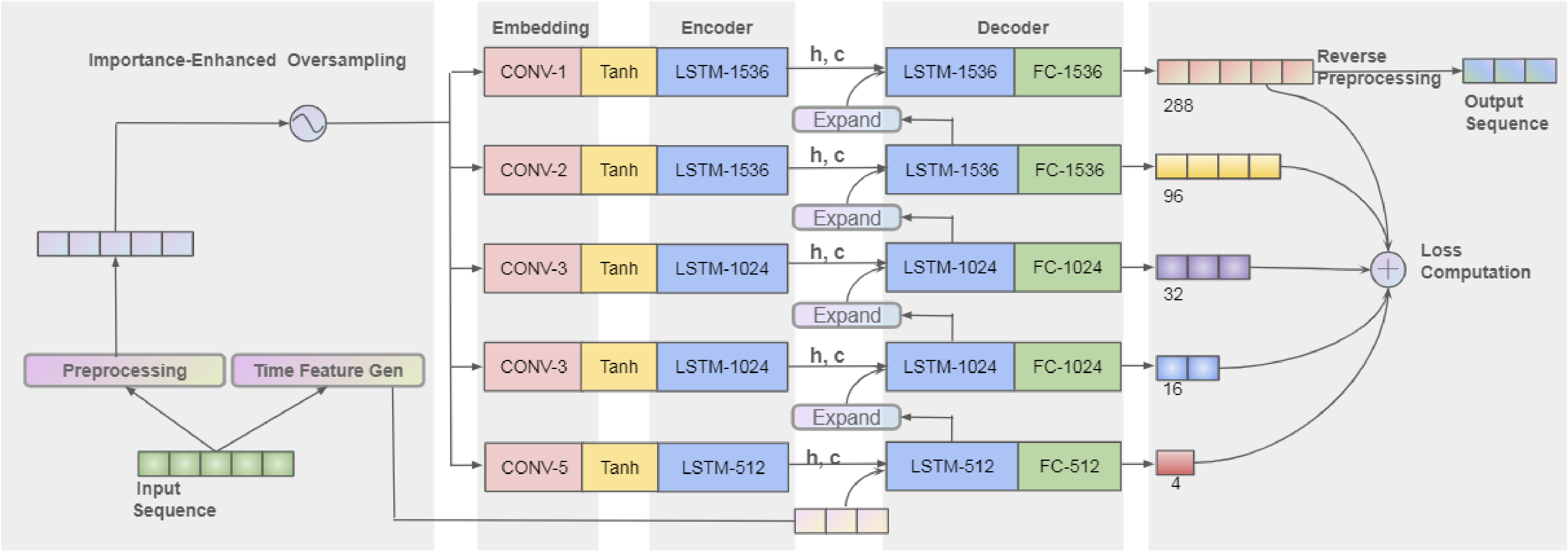
Representation Learning: A unique regularization strategy that incorporates a Kullback-Leibler divergence regularization loss term across multiple stacked layers, thereby increasing the model's robustness against anomalous events with divergent distributions.

Background: Piecewise Linear Representation (PLR)



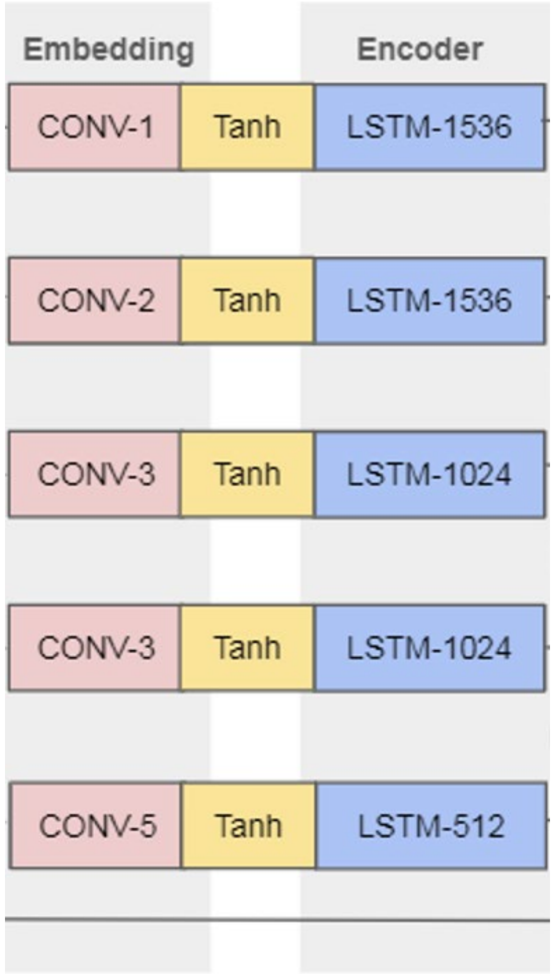
- PLR splits a series into several segments such that the maximum error of each segment does not exceed a threshold;
- **Prior work**: PLR describes the **linear** relationship of the multi-segment representation, mainly works as a preprocessing step to reduce both the space and computational cost of storing and transmitting time series.
- **Our work**: inspired by PLR, SEED learns **nonlinear segment representations** for heavily skewed long term time series.

SEED framework :



- Comprises three core components: embedding, encoder, and decoder.
- The encoder generates a unique hidden state and a cell state which serve as the initial values for the corresponding layers in the decoder.
- Each decoder layer is assigned a distinct task, as they represent the mean value distribution of different lengths of subsegments in the predicted sequence.

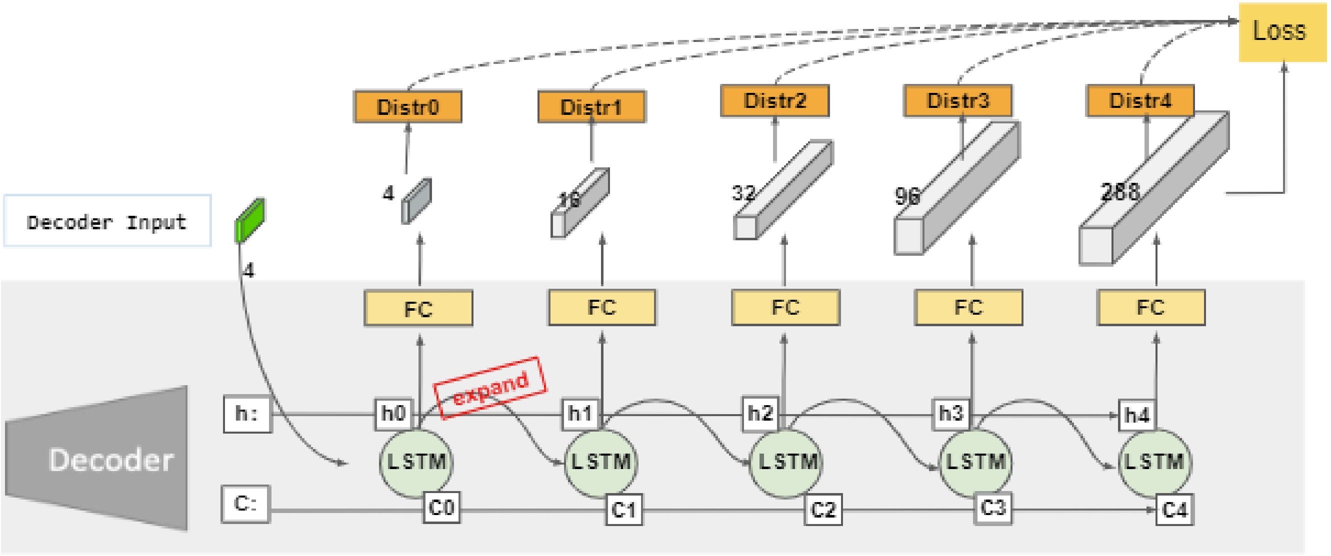
Convolutional Embedding Layers:



CNN Layers :

- different kernel sizes to extract features at different spatial scales.
- subsequent tanh activation function;
- lower level: larger kernel sizes, capturing broader patterns and global context.
- higher level: smaller kernel sizes, capturing local patterns and fine grained details.

Decoder Architecture:



- First level: the output length is 4, which is meant to predict the mean values of 4 segments, each of which contains 288/4 = 72 points in the forecasted series.
- Second level: the 4 outputs are expanded to16, each represents the mean value of 288/16 = 18 points.
- Expansion: ⟨a, b, c, d⟩ becomes ⟨a, a, a, a, b, b, b, b, , c, c, c, c, d, d, d, d⟩, in the **high-dimensional hidden space**.
- By predicting the mean value of different length sub-segments, **extreme values** are represented and spread across multiple levels in the hierarchy, leading to higher mean values in the segments containing them.

Multiple-Objective Loss Functions:

$$\mathrm{KL}(p \parallel q) = \sum p(x) \log\left(\frac{p(x)}{q(x)}\right),$$

$$\mathcal{L}_i = \mathrm{KL}\left(\mathrm{softmax}(p\_m_i), \mathrm{softmax}(g\_m_i)\right),$$

$$\mathcal{L} = RMSE(\hat{y}, y) + \lambda \times \left(\sum_{i=1}^{k} \mathcal{L}_i\right),$$

Motivations:

- Kullback-Leibler divergence loss acts as a regularization term that encourages the model to match the target distribution while balancing the sequence generation loss.
- p_m_i represents the predicted segment mean values in the *ith* layer, while g_m_i is the vector of computed ground truth mean values for the segments in the *ith* layer. .

Importance-Enhanced Oversampling Policy:

Input : Dataset with training and inference sequences
Output: Oversampled training set

Procedure Oversampling;

```
while training set size is not satisfied do
    Randomly sample a sequence including training and
      inference sections;
    if maximum value in inference section > T then
        Mark sequence as important;
        Move maximum value to the middle of the inference
          section of the sequence;
        foreach index I in the sequence with step size S do
            Sample starting at I;
            Add sampled sequence to oversampled training
              set;
        end
    end
    else
        Add sequence to oversampled training set;
    end
end
```

Steps:

- important sequences: maximum values in the inference section of the series exceed a threshold T;
- moving the maximum value to the middle of the inference section;
- multiple iterations of sampling from the beginning with a specified step size S.

Baselines:

- FEDFormer
- InFormer
- NLinear
- Dlinear
- NEC+
- EnDecoder, the common encoder-decoder model built with LSTM layers.

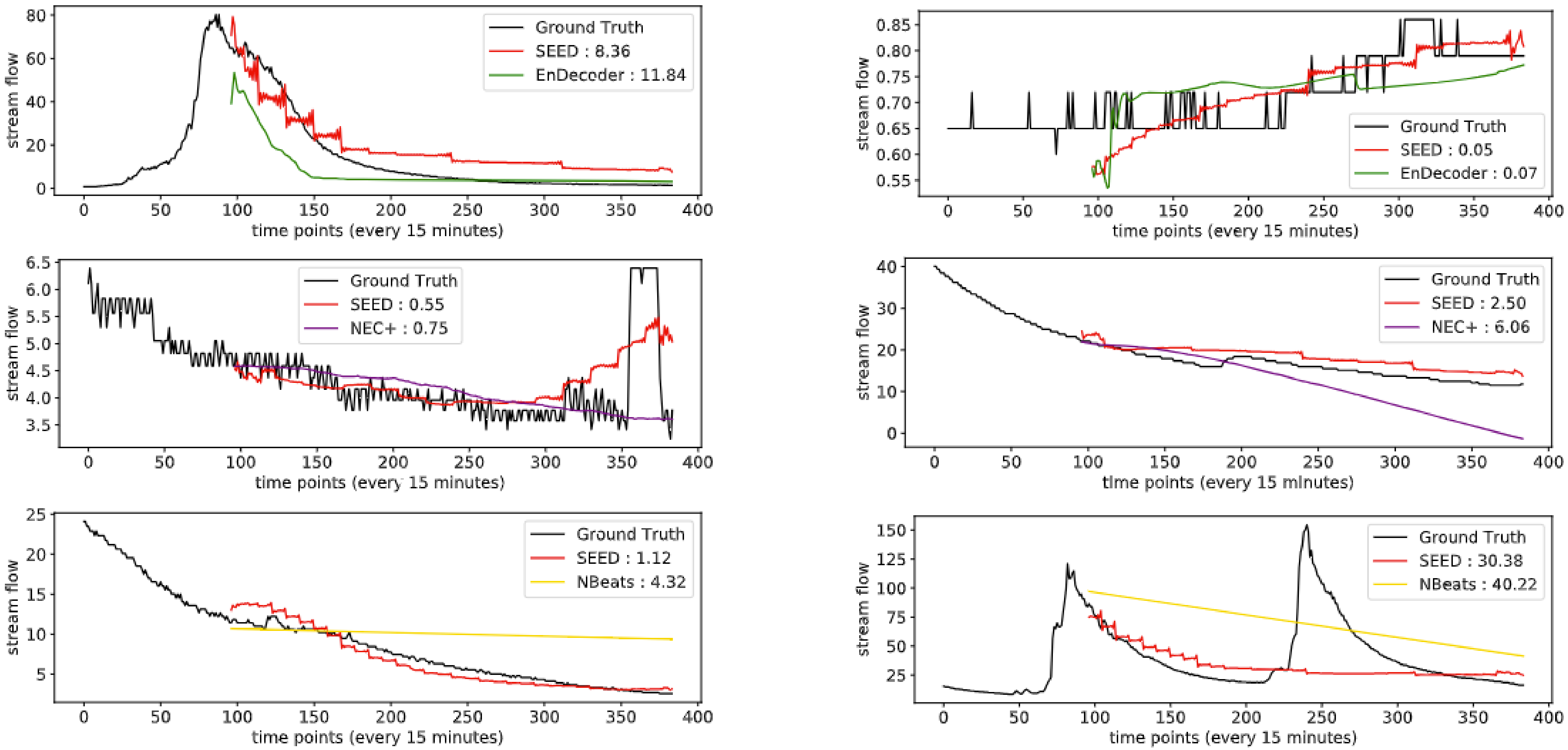
Main results:

Methods	Metric	Ross			Saratoga			Upperpen			SFC		
		All	High	Low	All	High	Low	All	High	Low	All	High	Low
FEDformer	RMSE	6.49	30.82	2.51	6.85	11.95	4.59	2.38	17.68	1.07	24.15	94.28	6.68
	MAPE	2.49	5.27	2.04	2.26	1.50	2.60	1.02	2.37	0.90	2.81	1.70	3.09
Informer	RMSE	9.14	31.00	5.56	4.89	13.64	1.01	5.33	16.26	4.40	19.00	85.40	2.46
	MAPE	5.45	5.80	5.39	0.73	1.40	0.43	4.21	2.70	4.34	0.54	0.71	0.49
Nlinear	RMSE	5.84	32.12	1.54	4.98	14.61	0.70	1.74	15.07	0.61	18.43	83.31	2.26
	MAPE	1.62	4.89	1.09	0.75	1.74	0.31	0.57	1.69	0.47	0.87	1.03	0.83
Dlinear	RMSE	6.90	30.96	2.97	4.06	7.63	2.48	3.25	14.01	2.33	23.64	79.76	9.65
	MAPE	2.79	4.03	2.58	1.31	0.85	1.51	2.04	1.68	2.07	4.02	1.04	4.76
NEC+	RMSE	9.33	38.34	4.58	1.95	5.55	0.35	1.94	13.92	0.92	16.39	76.63	1.38
	MAPE	4.53	8.33	3.91	0.21	0.30	0.17	0.80	0.84	0.80	0.55	0.61	0.54
NBeats	RMSE	5.16	30.09	1.08	3.60	9.44	1.01	1.23	13.20	0.21	31.47	95.33	15.55
	MAPE	1.25	3.17	0.94	0.70	1.21	0.47	0.25	0.78	0.20	3.24	0.88	3.83
EnDecoder	RMSE	5.58	30.81	1.45	1.93	5.69	0.26	2.95	16.33	1.80	17.46	79.04	2.11
	MAPE	1.62	3.72	1.28	0.16	0.29	0.11	1.81	2.34	1.76	0.84	0.96	0.80
SEED	RMSE	**4.23**	**29.74**	**0.05**	**1.67**	**5.14**	**0.12**	**1.07**	**12.83**	**0.07**	**14.44**	**70.04**	**0.59**
	MAPE	**0.11**	**0.53**	**0.04**	**0.09**	**0.19**	**0.05**	**0.10**	**0.57**	**0.06**	**0.20**	**0.42**	**0.14**

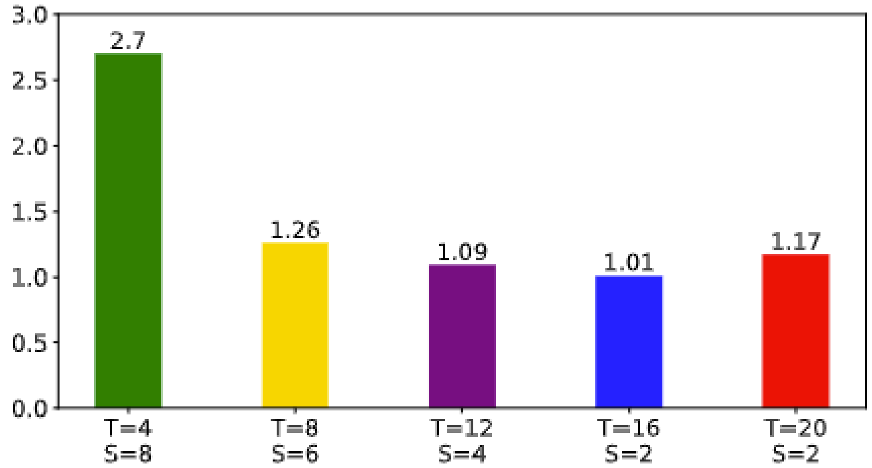
- Univariate Long-Term (h = 288) Series Forecasting Results.
- Over 1600 test points in the test set were inferenced on all datasets.
- The **best results** are in bold and the second best results are underlined.
- "All" represents the average RMSE of all test samples compared with the ground truth. "High" means larger than the mean value; "Low" includes test samples lower than the mean value.

In comparison to the three second-best models (NEC+, Nbeats and EnDecoder), SEED achieved, on average, relative RMSE reduction of 31.44%, 34.68%, and 29.67% across the datasets.

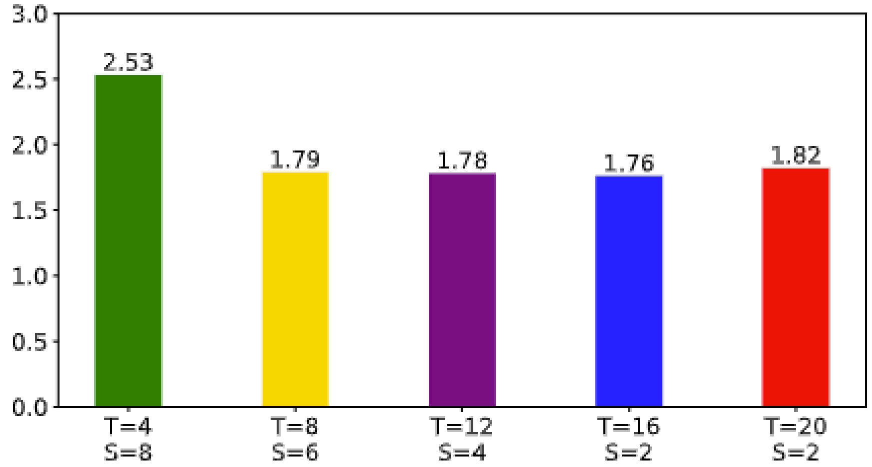
Example comparisons with the second best baselines:



Effect of the Importance-Enhanced Oversampling Policy:



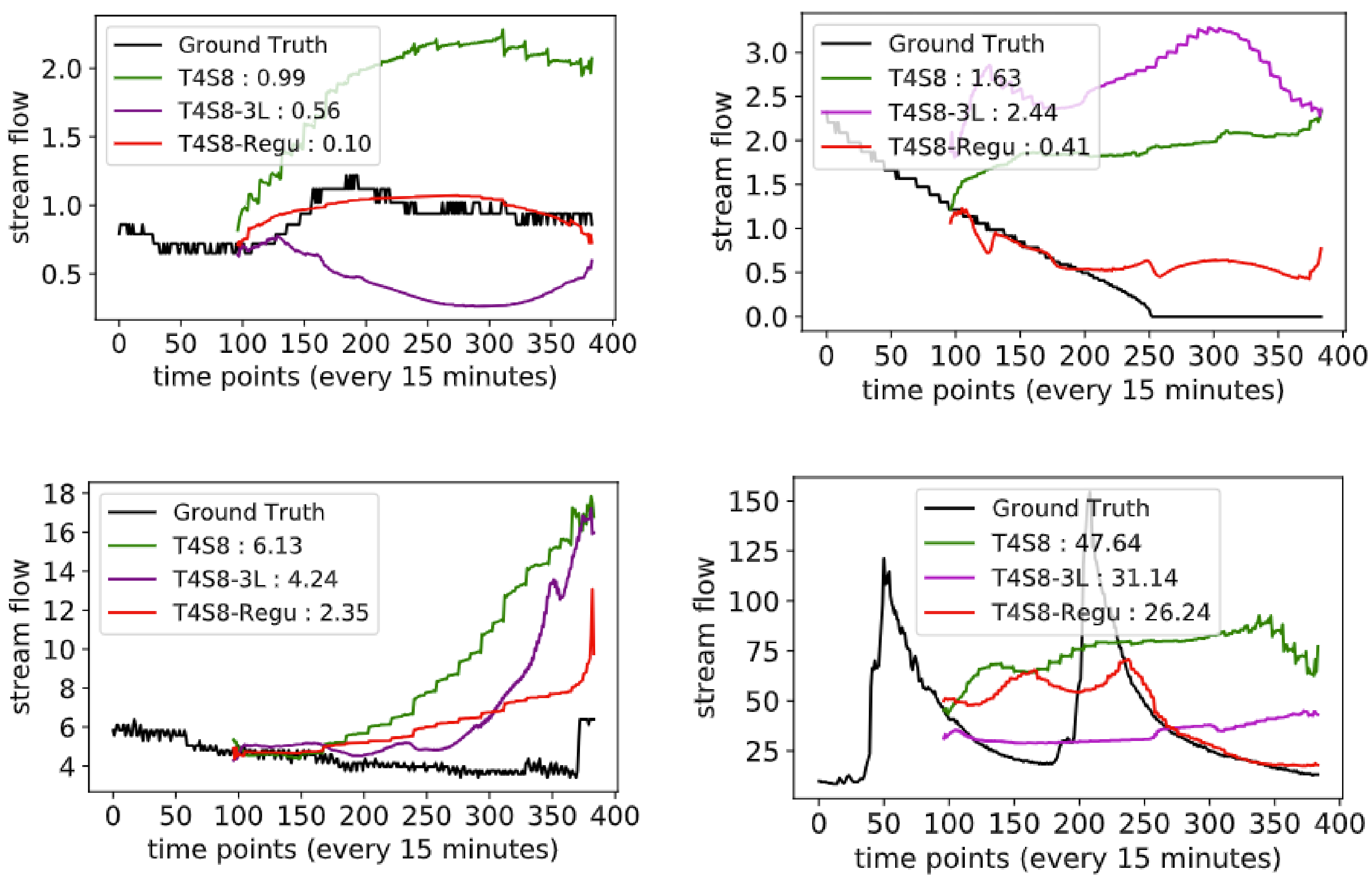
(a) RMSE of UpperPen



(b) RMSE of Saratoga

- To evaluate the impact of this policy, we increased the threshold T while simultaneously decreasing the step size S.
- Increasing the threshold T and decreasing the step size S had a positive impact on the results.
- There is an optimal threshold T value beyond which the policy's effectiveness plateaus.

Effect of the KL Regularization Terms & Segment Expanding:



We just use T=4, S=8 as an example:

- T4S8: 5-layer without regularization loss terms.
- T4S8-3L: 3-layer SEED with regularization loss terms.
- T4S8-Regu: 5-layer SEED with regularization loss terms, which gives the best result.

Q & A