Memformer

A Memory Guided Transformer for Time Series Forecasting

Yunyao Cheng, Chenjuan Guo, Bin Yang, Haomin Yu, Kai Zhao, Christian S. Jensen

February 2025

Proceedings of the VLDB Endowment, Volume 18, Issue 2

Presented by **Andreas Gottschalk Krath**

1. Introduction

1.1 Motivation

Forecasting

- Predicting the future
  - Allows preparation

1.1 Motivation

Forecasting

- Predicting the future
  - Allows preparation
- Long term forecasting?
  - Obviously more difficult than short term
  - Time constrained tasks

Long Term Forecasting

- What defines long term?

Long Term Forecasting

- What defines long term?
- Historical horizon
- Forecasting horizon

Long Term Forecasting

- What defines long term?
- Historical horizon
- Forecasting horizon
- Both exceed 96 time steps
  - Hourly time step $\longrightarrow$ 4 days

Long Term Forecasting

- What defines long term?
- Historical horizon
- Forecasting horizon
- Both exceed 96 time steps
    - Hourly time step $\longrightarrow$ 4 days

Variable Correlation

- Complex systems have many variables
    - These relate to each other
- These impact forecasting accuracy
    - Patterns in the data

Dynamic Correlations

- Are variable correlations stable over time?
  - No

Dynamic Correlations

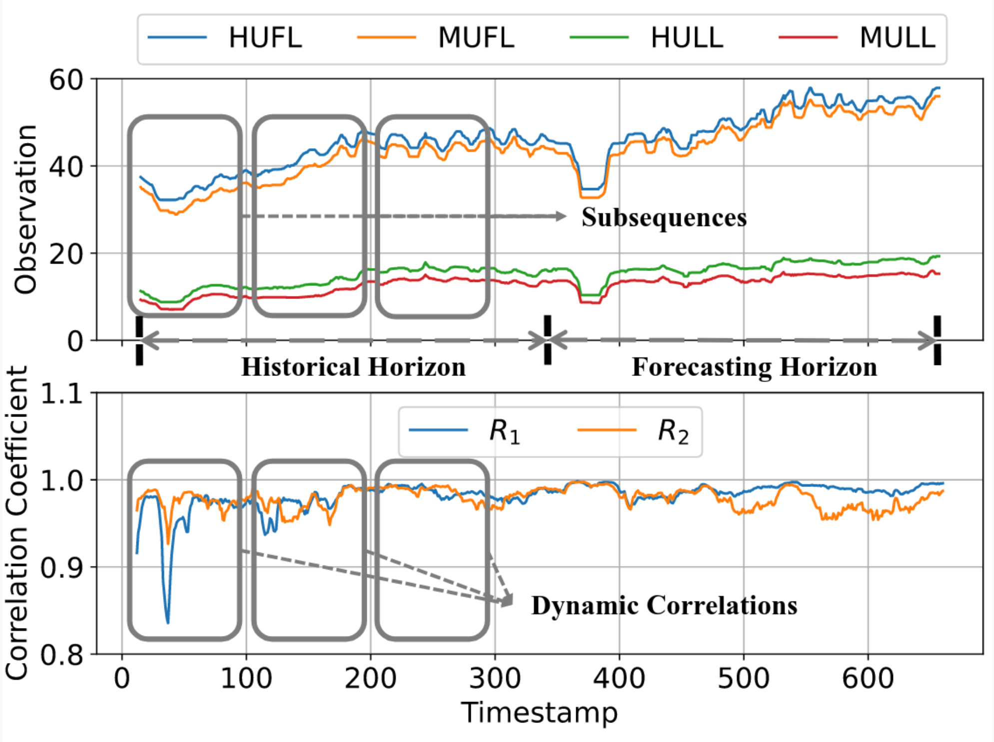
- Are variable correlations stable over time?
  - No
- Correlations are dynamic over time
  - Seasons
  - Sensor drift

Dynamic Correlations

- Are variable correlations stable over time?
  - No
- Correlations are dynamic over time
  - Seasons
  - Sensor drift
- We often consider average
  - Especially hurtful in time series
  - Predictions are bad in periods

Dynamic Correlations

- Are variable correlations stable over time?
  - No
- Correlations are dynamic over time
  - Seasons
  - Sensor drift
- We often consider average
  - Especially hurtful in time series
  - Predictions are bad in periods



(a) Dynamic correlations. The Average $R_1 = 0.995$ and $R_2 = 0.990$.

1.1 Motivation

Disrupted Correlations

- System errors
- External influence

1.1 Motivation

Disrupted Correlations

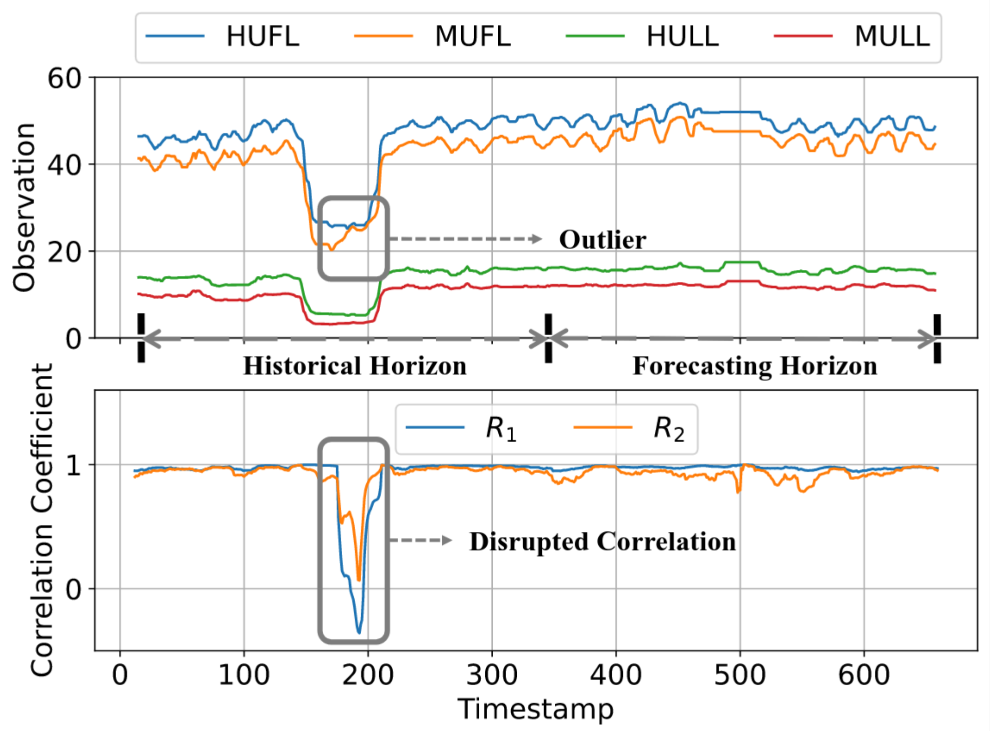
- System errors
- External influence
- What happens with outliers?
  - Affect correlation $\longrightarrow$ accuracy

Disrupted Correlations

- System errors
- External influence
- What happens with outliers?
  - Affect correlation $\longrightarrow$ accuracy
- Many models are sensitive to outliers
  - Numeric difference dominates training

Disrupted Correlations

- System errors
- External influence
- What happens with outliers?
  - Affect correlation ⟶ accuracy
- Many models are sensitive to outliers
  - Numeric difference dominates training



(b) Disrupted correlation. The Average $R_1 = 0.908$ and $R_2 = 0.963$.

Challenge 1

- Capture dynamic correlations
- Mitigate disrupted correlations
- Existing solutions struggle with the latter
  - Capture dynamic and disrupted
  - Reduces model robustness

Challenge 1

- Capture dynamic correlations
- Mitigate disrupted correlations
- Existing solutions struggle with the latter
  - Capture dynamic and disrupted
  - Reduces model robustness

Challenge 2

- Local information 🤝 global information
- Global information is *all* local information
- Local information *affects* global information
- Existing solutions struggle with combining
  - Only local
  - Only global

1.3 Contributions

Memformer

- Transformer
- Patch-wise recurrent graph learning
  - Captures dynamic correlations
- Global attention
  - Mitigates disrupted correlations
- Adresses challenge 1

Memformer

- Transformer
- Patch-wise recurrent graph learning
  - Captures dynamic correlations
- Global attention
  - Mitigates disrupted correlations
- Adresses challenge 1

Alternating Memory Enhancer

- Memory network
- Associates local and global information
- Adresses challenge 2

1.3 Contributions

Memformer

- Transformer
- Patch-wise recurrent graph learning
  - Captures dynamic correlations
- Global attention
  - Mitigates disrupted correlations
- Adresses challenge 1

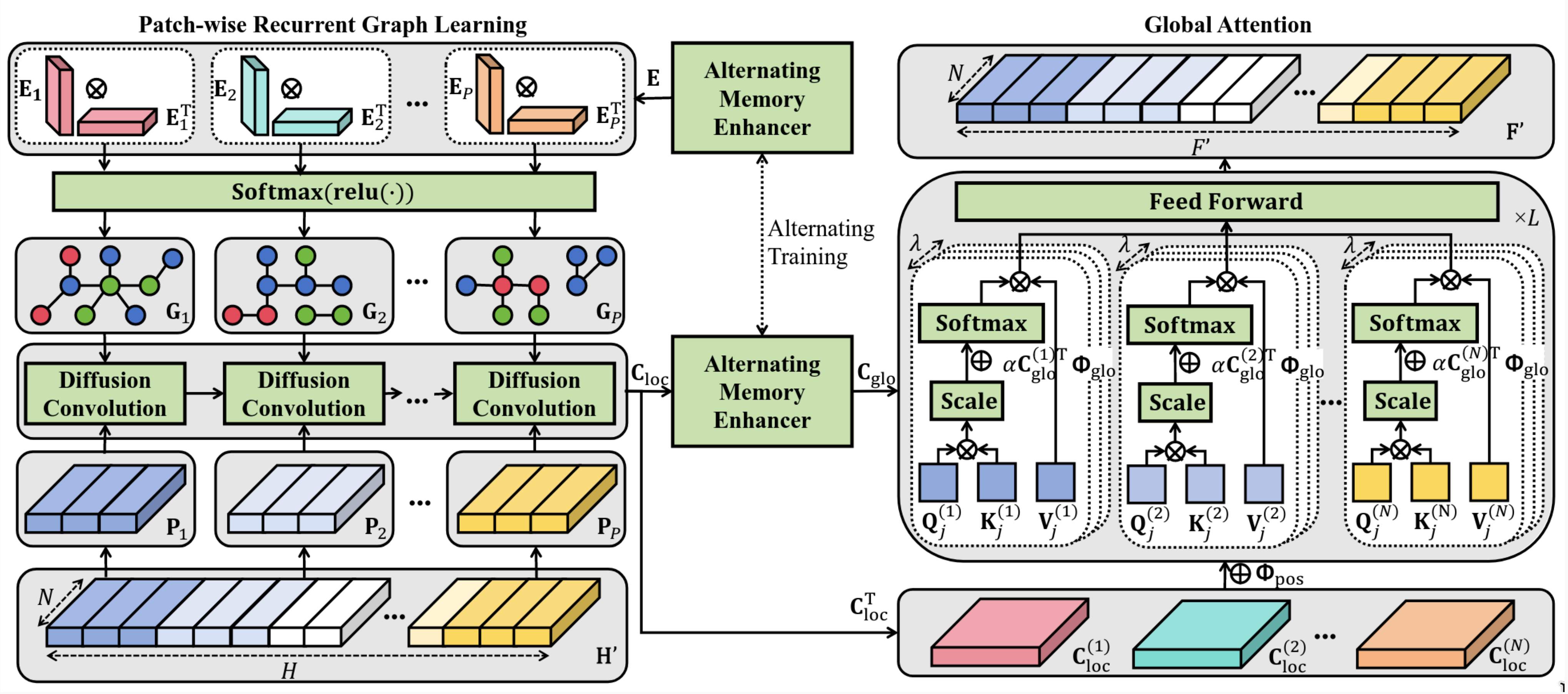
Alternating Memory Enhancer

- Memory network
- Associates local and global information
- Adresses challenge 2

Experiments

- Proof

2. Methodology

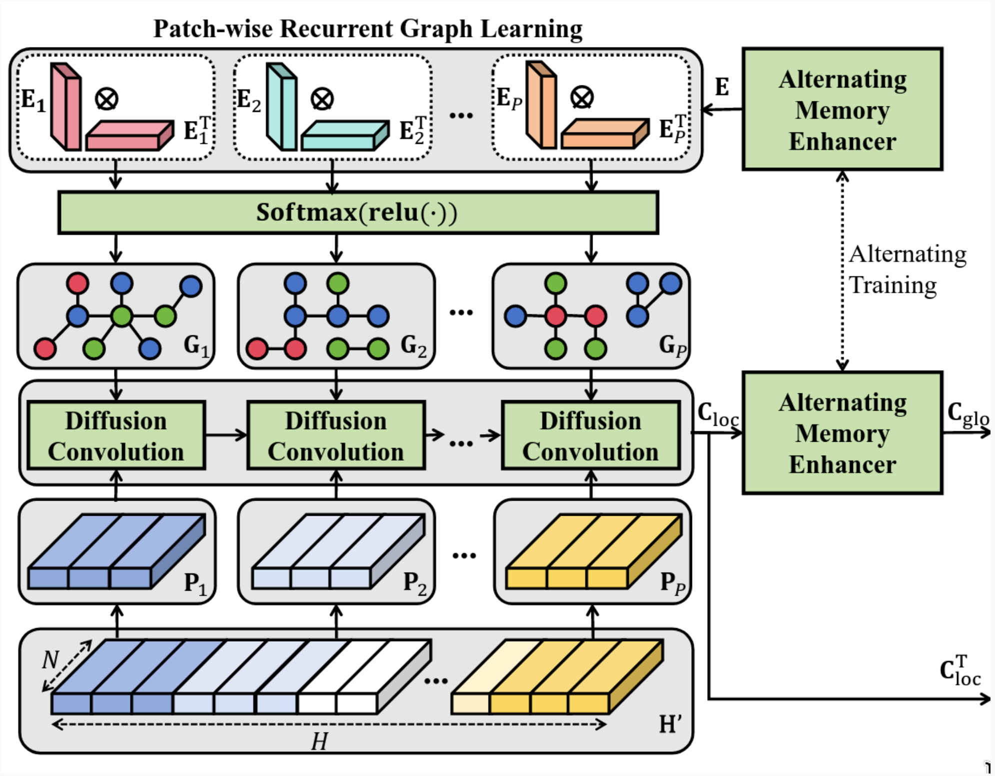

Patch-wise Recurrent Graph Learning
Global Attention
Alternating Memory Enhancer
Softmax(relu(·))
Alternating Training
Diffusion Convolution
Feed Forward
Softmax
Scale
×L
E
C_loc
C_glo
H'
F'

Architecture

Upper part → dynamic correlation

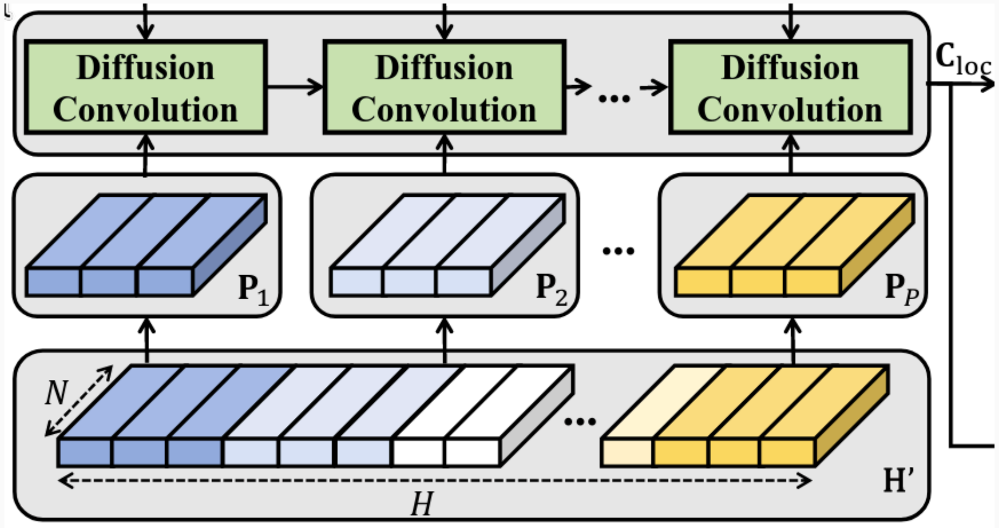
Lower part → normalized data

Output → enriched input features



Normalized Data

- Instance normalization

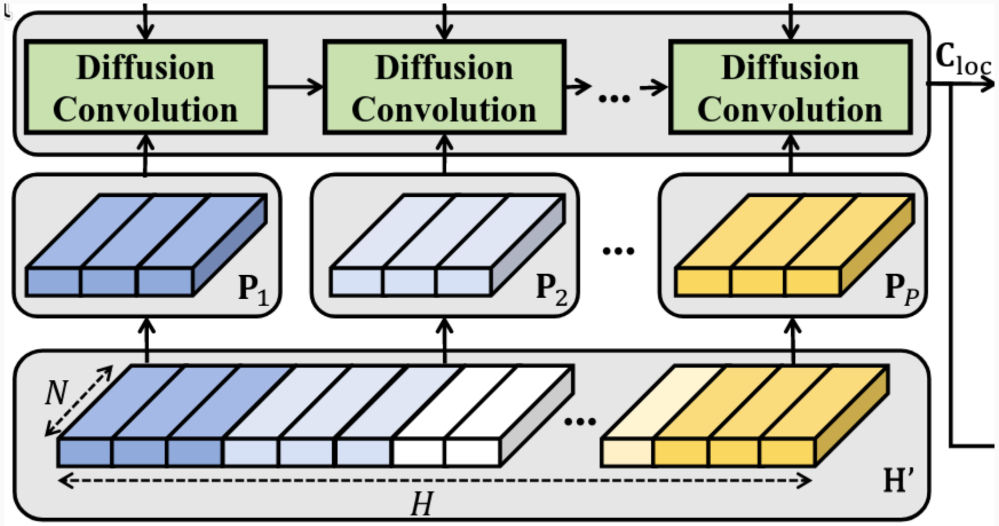


Normalized Data

- Instance normalization

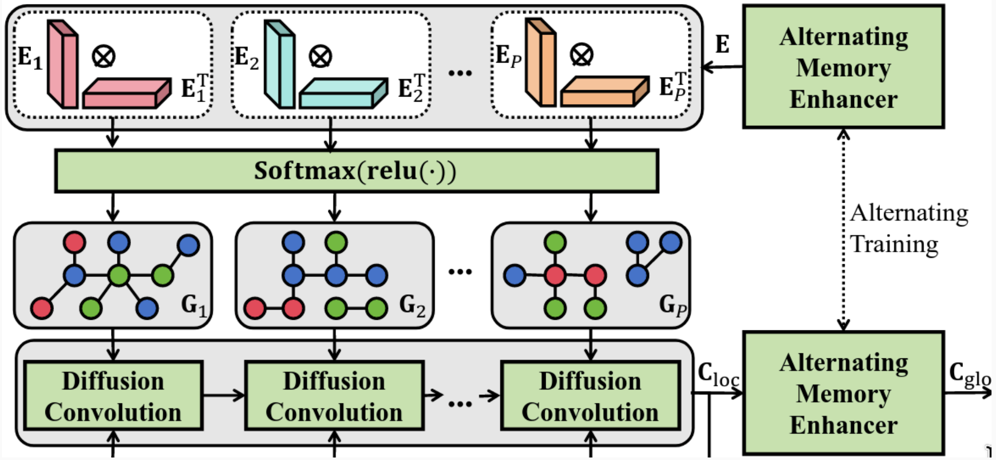
Patches

- H' is split into p patches
- Group temporally related data



AME

- Provides local information
  - These are learnable parameters
- Consistant local information for patch P_i
- Matrix product of $E_i \otimes E_i^T$
  - Similarity matrix for variables in P_i

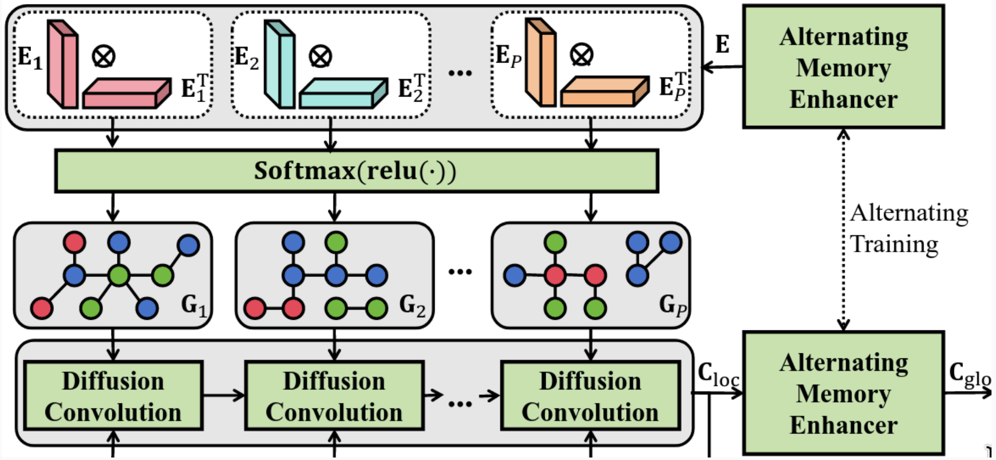


AME

- Provides local information
  - These are learnable parameters
- Consistant local information for patch P_i
- Matrix product of $E_i \otimes E_i^T$
  - Similarity matrix for variables in P_i

ReLU + Softmax

- ReLU eliminates negative values
  - Removes negative correlations
- Softmax scales into influence scores



AME

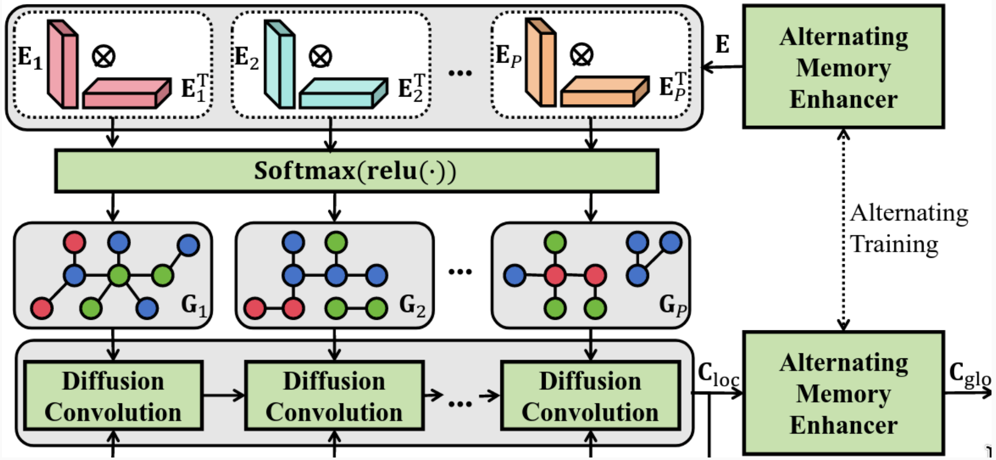
- Provides local information
  - These are learnable parameters
- Consistant local information for patch P_i
- Matrix product of $E_i \otimes E_i^T$
  - Similarity matrix for variables in P_i

ReLU + Softmax

- ReLU eliminates negative values
  - Removes negative correlations
- Softmax scales into influence scores

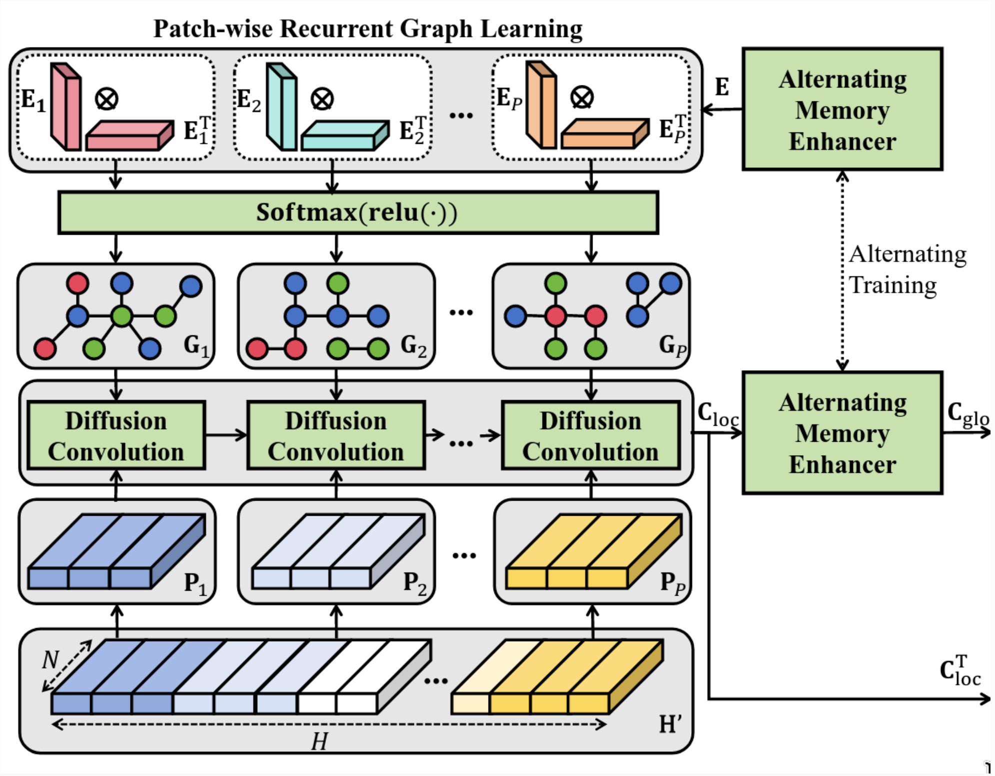
Graph

- Translates influence scores into graph
- Captures connection between variables
  - Dynamic correlations



Diffusion Convolution

- Normalized data is adjusted based on connections in graph
- Numeric values "diffuse" into neighbours
  - Not only immediate neighbours
- Spatially relates data based on connections

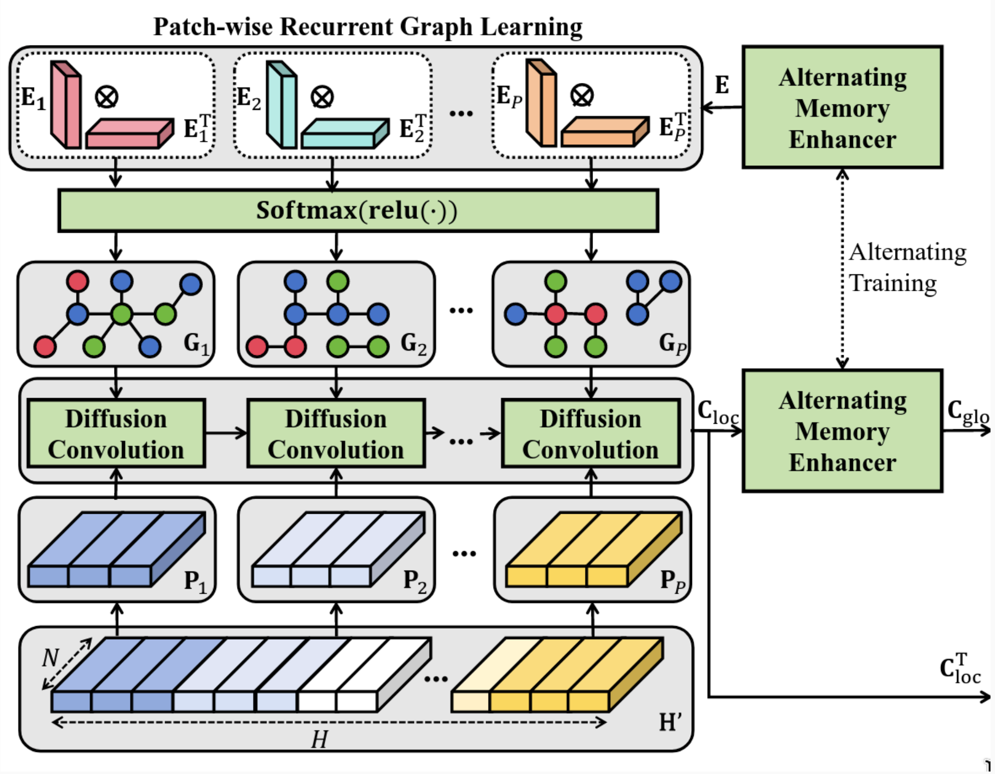


Diffusion Convolution

- Normalized data is adjusted based on connections in graph
- Numeric values "diffuse" into neighbours
  - Not only immediate neighbours
- Spatially relates data based on connections

Gated Recurrent Unit

- Forwards information from P_i to P_{i+1}
- Temporally relates data in a sequence



Diffusion Convolution

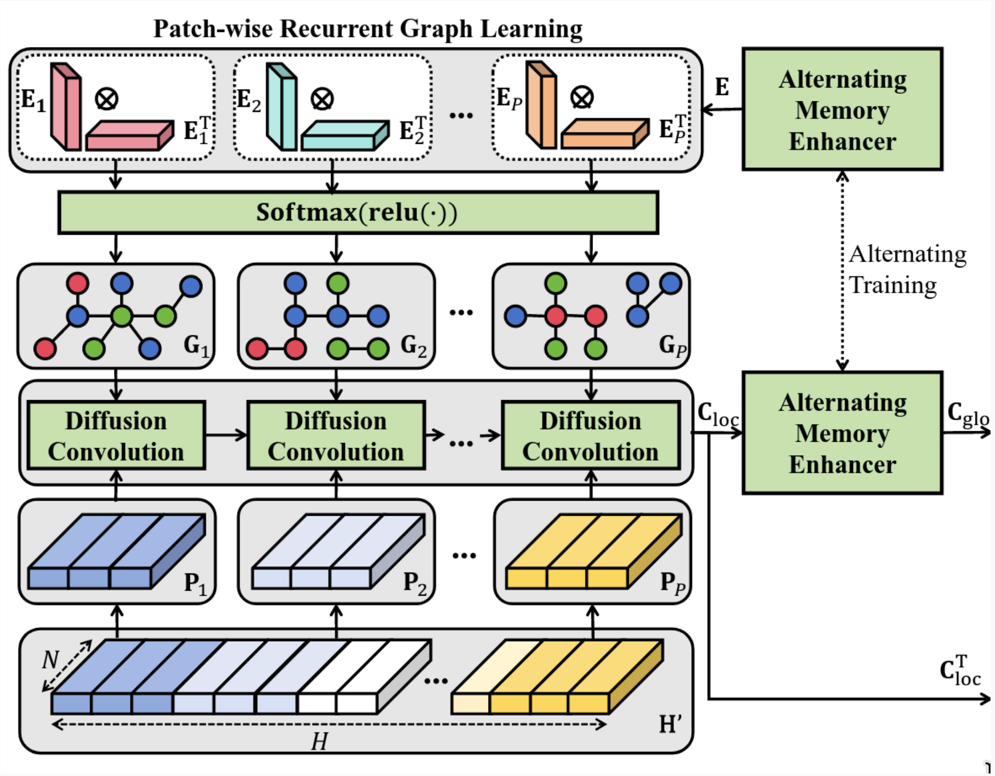
- Normalized data is adjusted based on connections in graph
- Numeric values “diffuse” into neighbours
  - Not only immediate neighbours
- Spatially relates data based on connections

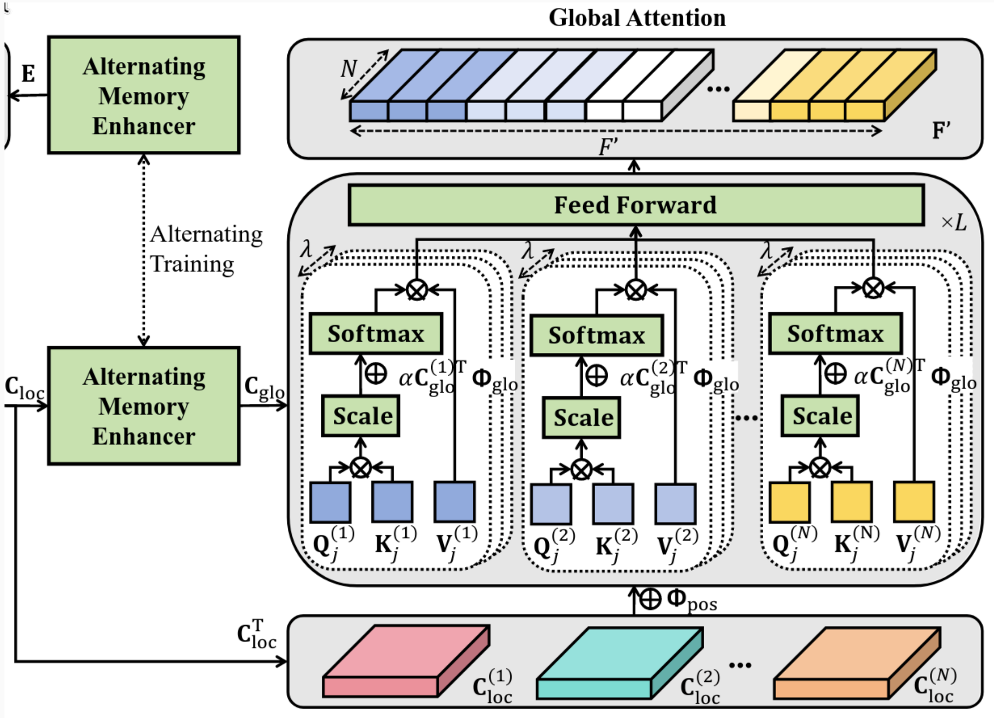
Gated Recurrent Unit

- Forwards information from P_i to P_{i+1}
- Temporally relates data in a sequence

Output

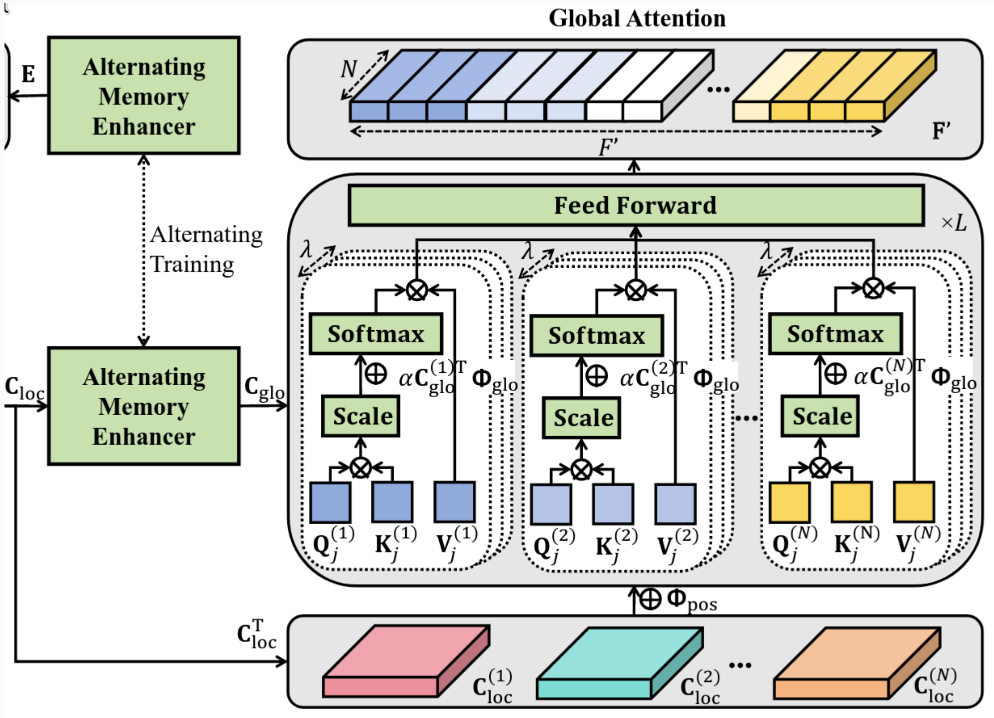
- Input features enriched with local information
- Spatial $\longrightarrow$ dynamic correlations
- Temporal $\longrightarrow$ GRU




Global Attention
E
Alternating Memory Enhancer
Alternating Training
$\mathbf{C}_{loc}$
Alternating Memory Enhancer
$\mathbf{C}_{glo}$
N
F'
$\mathbf{F}'$
Feed Forward
$\times L$
λ
Softmax
$\alpha \mathbf{C}_{glo}^{(1)\mathrm{T}}$
$\mathbf{\Phi}_{glo}$
Scale
$\mathbf{Q}_j^{(1)}$ $\mathbf{K}_j^{(1)}$ $\mathbf{V}_j^{(1)}$
$\alpha \mathbf{C}_{glo}^{(2)\mathrm{T}}$
$\mathbf{Q}_j^{(2)}$ $\mathbf{K}_j^{(2)}$ $\mathbf{V}_j^{(2)}$
$\alpha \mathbf{C}_{glo}^{(N)\mathrm{T}}$
$\mathbf{Q}_j^{(N)}$ $\mathbf{K}_j^{(N)}$ $\mathbf{V}_j^{(N)}$
$\mathbf{\Phi}_{pos}$
$\mathbf{C}_{loc}^{\mathrm{T}}$
$\mathbf{C}_{loc}^{(1)}$ $\mathbf{C}_{loc}^{(2)}$ $\mathbf{C}_{loc}^{(N)}$

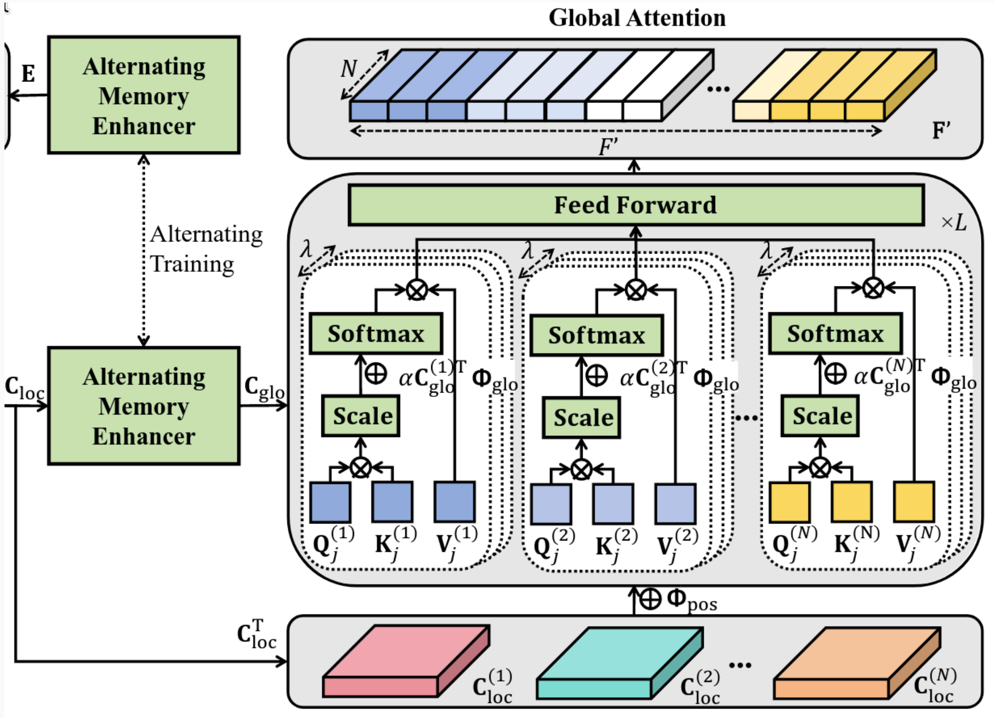
Input

- Transpose locally enriched features
  - Isolate variables
  - Diffusion earlier
- Converted to Q, K, V matrices
  - Learnable parameters



Attention

- Relatively conventional implementation
- Global information is new
- Adding global information after softmax
  - Bias probabilities

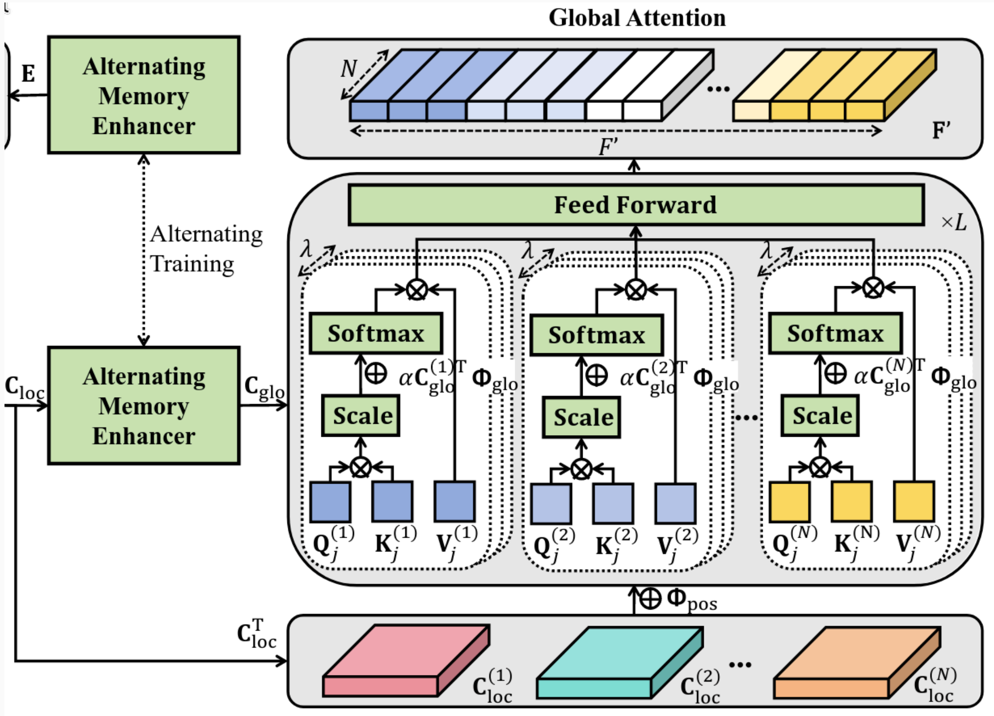


Attention

- Relatively conventional implementation
- Global information is new
- Adding global information after softmax
  - Bias probabilities

Output

- The final “representation” of data
- **F'** is not a forecast
  - Final feature representation
- Linear layer maps to forecasting horizon



3. Experiments

Datasets

- 7 in total
  - 4 are variants of the same
- 7, 21, 321, and 862 variables
- $H = 336$
- $F = [96, 192, 336, 720]$

3.1 Noteworthy Details

Datasets

- 7 in total
  - 4 are variants of the same
- 7, 21, 321, and 862 variables
- $H = 336$
- $F = [96, 192, 336, 720]$

Comparisons

- Multiple different model architectures
  - Channel independent models
  - Linear models
  - Attention models

Results

- Compare on MSE and MAE
- Bold is best, underline is second best
- Almost always best performance
  - Loses on MSE for low F in one dataset

Models		Memformer		ModernTCN		PatchTST		NLinear		DLinear		iTransformer		CARD		Crossformer		MTGNN	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Weather	96	0.151	**0.185**	0.155	0.201	0.152	0.199	0.182	0.232	0.176	0.237	0.174	0.214	0.150	0.188	**0.145**	0.211	0.342	0.385
	192	0.197	**0.231**	0.198	0.245	0.197	0.243	0.225	0.269	0.220	0.282	0.221	0.254	0.202	0.238	**0.190**	0.259	0.427	0.445
	336	**0.247**	**0.274**	0.251	0.286	0.249	0.283	0.271	0.301	0.265	0.319	0.278	0.296	0.260	0.282	0.259	0.326	0.506	0.523
	720	**0.318**	**0.326**	0.321	0.336	0.320	0.335	0.338	0.348	0.323	0.362	0.358	0.347	0.343	0.353	0.332	0.382	0.510	0.527
Traffic	96	**0.361**	**0.230**	0.368	0.253	0.367	0.251	0.410	0.279	0.410	0.282	0.395	0.268	0.419	0.269	0.511	0.292	0.516	0.308
	192	**0.381**	**0.239**	0.384	0.261	0.385	0.259	0.423	0.284	0.423	0.287	0.417	0.276	0.443	0.276	0.523	0.311	0.534	0.324
	336	**0.394**	**0.245**	0.397	0.270	0.398	0.265	0.435	0.290	0.436	0.296	0.433	0.283	0.460	0.283	0.530	0.300	0.540	0.335
	720	**0.432**	**0.267**	0.440	0.296	0.434	0.287	0.464	0.307	0.466	0.315	0.467	0.302	0.490	0.299	0.573	0.313	0.557	0.343
Electricity	96	**0.130**	**0.217**	0.131	0.228	**0.130**	0.222	0.141	0.237	0.140	0.237	0.132	0.228	0.141	0.233	0.186	0.281	0.202	0.314
	192	**0.147**	**0.232**	0.150	0.242	0.148	0.240	0.154	0.248	0.153	0.249	0.154	0.249	0.160	0.250	0.208	0.300	0.266	0.349
	336	**0.162**	**0.249**	0.171	0.265	0.167	0.261	0.171	0.265	0.169	0.267	0.172	0.267	0.173	0.263	0.323	0.369	0.328	0.373
	720	**0.199**	**0.281**	0.203	0.294	0.202	0.291	0.210	0.297	0.203	0.301	0.204	0.296	0.197	0.284	0.404	0.423	0.422	0.410
ETTh1	96	**0.362**	**0.385**	0.382	0.401	0.375	0.399	0.374	0.394	0.375	0.399	0.386	0.405	0.383	0.391	0.377	0.419	0.401	0.442
	192	**0.386**	**0.404**	0.420	0.424	0.414	0.421	0.408	0.415	0.405	0.416	0.441	0.436	0.435	0.420	0.410	0.439	0.587	0.601
	336	**0.402**	**0.421**	0.427	0.434	0.431	0.436	0.429	0.427	0.439	0.443	0.487	0.458	0.479	0.442	0.440	0.461	0.736	0.643
	720	**0.436**	**0.452**	0.450	0.461	0.449	0.466	0.440	0.453	0.472	0.490	0.503	0.491	0.471	0.461	0.519	0.524	0.916	0.750
ETTh2	96	**0.264**	**0.321**	0.276	0.342	0.274	0.336	0.277	0.338	0.289	0.353	0.297	0.349	0.281	0.330	0.770	0.529	0.735	0.643
	192	**0.314**	**0.358**	0.340	0.381	0.339	0.379	0.344	0.381	0.383	0.418	0.380	0.400	0.363	0.381	0.848	0.657	0.859	0.717
	336	**0.312**	**0.364**	0.329	0.378	0.331	0.380	0.357	0.400	0.448	0.465	0.428	0.432	0.411	0.418	0.859	0.674	1.050	0.849
	720	**0.374**	**0.410**	0.392	0.433	0.379	0.422	0.394	0.436	0.605	0.551	0.427	0.445	0.416	0.431	1.221	0.825	1.336	0.963
ETTm1	96	**0.285**	**0.336**	0.292	0.346	0.290	0.342	0.306	0.348	0.299	0.343	0.334	0.368	0.316	0.347	0.320	0.373	0.428	0.446
	192	**0.323**	**0.358**	0.332	0.368	0.332	0.369	0.349	0.375	0.335	0.365	0.377	0.391	0.363	0.370	0.372	0.411	0.551	0.505
	336	**0.365**	**0.381**	0.367	0.393	0.366	0.392	0.375	0.388	0.369	0.386	0.426	0.420	0.392	0.390	0.429	0.441	0.706	0.622
	720	**0.419**	**0.409**	0.422	0.429	0.420	0.424	0.433	0.422	0.425	0.421	0.491	0.459	0.458	0.425	0.573	0.531	0.982	0.764
ETTm2	96	**0.160**	**0.245**	0.166	0.256	0.165	0.255	0.167	0.255	0.167	0.260	0.180	0.264	0.169	0.248	0.254	0.348	0.442	0.483
	192	**0.215**	**0.285**	0.222	0.293	0.220	0.292	0.221	0.293	0.224	0.303	0.250	0.309	0.234	0.292	0.370	0.433	0.642	0.570
	336	**0.263**	**0.317**	0.276	0.327	0.278	0.329	0.274	0.327	0.281	0.342	0.311	0.348	0.294	0.339	0.511	0.527	0.726	0.658
	720	**0.350**	**0.372**	0.365	0.383	0.367	0.385	0.368	0.384	0.397	0.421	0.412	0.407	0.390	0.388	0.901	0.689	1.139	0.862

Disrupted Correlations

- Robustness
- Introduce outliers
  - Different amounts
  - Independent
  - Dependent

3.3 Challenge 1

Disrupted Correlations

- Robustness
- Introduce outliers
  - Different amounts
  - Independent
  - Dependent
- Results
  - Performed the best
  - Mitigate both types of outliers

3.3 Challenge 1

Disrupted Correlations

- Robustness
- Introduce outliers
  - Different amounts
  - Independent
  - Dependent
- Results
  - Performed the best
  - Mitigate both types of outliers

Dynamic Correlations

- Introduce dynamic correlations
  - Different amounts

Disrupted Correlations

- Robustness
- Introduce outliers
  - Different amounts
  - Independent
  - Dependent
- Results
  - Performed the best
  - Mitigate both types of outliers

Dynamic Correlations

- Introduce dynamic correlations
  - Different amounts
- Results
  - Performed the best

4. Critique

Instance normalization

- Normalize within historical horizon only
- Mitigates the issue of internal covariate shift
- Allows model to effectively grasp the intricate temporal dynamics inherent in time series

$H' = (H - \mu)/\sqrt{(\sigma^2 + c)}$, where

H is the historical horizon

μ is the mean

σ is the variance

c ensures numerical stability

4.1 Preprocessing

Instance normalization

- Normalize within historical horizon only
- Mitigates the issue of internal covariate shift
- Allows model to effectively grasp the intricate temporal dynamics inherent in time series

$H' = (H - \mu)/\sqrt{(\sigma^2 + c)}$, where

H is the historical horizon

μ is the mean

σ is the variance

c ensures numerical stability

poral dynamics inherent in time series. Instance normalization is defined as $\mathbf{H}' = (\mathbf{H} - \mu)/\sqrt{(\sigma^2 + \text{constant})}$, where $\mathbf{H}'$ denotes the preprocessed feature, μ and σ denote the mean and variance of the sample, respectively, and "constant" is a small positive real number included to ensure numerical stability.

Instance normalization

- Normalize within historical horizon only
- Mitigates the issue of internal covariate shift
- Allows model to effectively grasp the intricate temporal dynamics inherent in time series

$$H' = (H - \mu)/\sqrt{(\sigma^2 + c)}, \text{where}$$

H is the historical horizon

μ is the mean

σ is the variance

c ensures numerical stability

poral dynamics inherent in time series. Instance normalization is defined as $\mathbf{H}' = (\mathbf{H} - \mu)/\sqrt{(\sigma^2 + \text{constant})}$, where $\mathbf{H}'$ denotes the preprocessed feature, μ and σ denote the mean and variance of the sample, respectively, and "constant" is a small positive real number included to ensure numerical stability.

- Mistake in variance notation?
  - σ is conventional notation for standard deviation
  - σ^2 is conventional notation for variance

What is going on?

4.1 Preprocessing

What is going on?

- Explored code to find answer
- `data_provider/data_loader.py`
  - Only place anything related to loading data happens
  - `Dataset_ETT_hour`, `Dataset_ETT_minute`, `Dataset_Custom`, `Dataset_Pred`

What is going on?

- Explored code to find answer
- `data_provider/data_loader.py`
  - Only place anything related to loading data happens
  - `Dataset_ETT_hour`, `Dataset_ETT_minute`, `Dataset_Custom`, `Dataset_Pred`

```
from sklearn.preprocessing import StandardScaler
class ...:
    def __read_data__(self):
        self.scalar = StandardScaler()
        self.scaler.fit(train_data.values)
        data = self.scaler.transform(df_data.values)
```

What is going on?

- Explored code to find answer
- `data_provider/data_loader.py`
  - Only place anything related to loading data happens
  - `Dataset_ETT_hour`, `Dataset_ETT_minute`, `Dataset_Custom`, `Dataset_Pred`

```
from sklearn.preprocessing import StandardScaler
class ...:
    def __read_data__(self):
        self.scalar = StandardScaler()
        self.scaler.fit(train_data.values)
        data = self.scaler.transform(df_data.values)
```

- They fit on training data
- Normalize entire dataset with μ and σ from training data

What are they actually doing?

Preprocessing

$H' = (H - \mu)/\sqrt{(\sigma^2 + c)}$, where

H is the historical horizon

μ is the mean

σ is the variance

c ensures numerical stability

StandardScaler

$z = (x - \mu)/\sigma$, where

x is the sample

μ is the mean

σ is the standard deviation

What are they actually doing?

Preprocessing

$H' = (H - \mu)/\sqrt{(\sigma^2 + c)}$, where

H is the historical horizon

μ is the mean

σ is the variance

c ensures numerical stability

StandardScaler

$z = (x - \mu)/\sigma$, where

x is the sample

μ is the mean

σ is the standard deviation

- We know that $\sqrt{\sigma^2} = \sigma$

What are they actually doing?

Preprocessing

$H' = (H - \mu)/\sqrt{(\sigma^2 + c)}$, where

H is the historical horizon

μ is the mean

σ is the variance

c ensures numerical stability

StandardScaler

$z = (x - \mu)/\sigma$, where

x is the sample

μ is the mean

σ is the standard deviation

- We know that $\sqrt{\sigma^2} = \sigma$
- Essentially same formula, except constant

What are they actually doing?

Preprocessing

$$H' = (H - \mu)/\sqrt{(\sigma^2 + c)}, \text{where}$$

H is the historical horizon

μ is the mean

σ is the variance

c ensures numerical stability

StandardScaler

$$z = (x - \mu)/\sigma, \text{where}$$

x is the sample

μ is the mean

σ is the standard deviation

- We know that $\sqrt{\sigma^2} = \sigma$
- Essentially same formula, except constant
- Fit on training data, normalize entire dataset $\longrightarrow$ global normalization

What are they actually doing?

Preprocessing

$H' = (H - \mu)/\sqrt{(\sigma^2 + c)}$, where

H is the historical horizon

μ is the mean

σ is the variance

c ensures numerical stability

StandardScaler

$z = (x - \mu)/\sigma$, where

x is the sample

μ is the mean

σ is the standard deviation

- We know that $\sqrt{\sigma^2} = \sigma$
- Essentially same formula, except constant
- Fit on training data, normalize entire dataset $\longrightarrow$ global normalization
- None of the stated benefits of instance normalization
  - Mitigate internal covariate shift
  - Grasp intricate temporal dynamics in TS