TEMPORAL FUSION TRANSFORMERS FOR INTERPRETABLE MULTI-HORIZON TIME SERIES FORECASTING

2021.03.28

백지윤

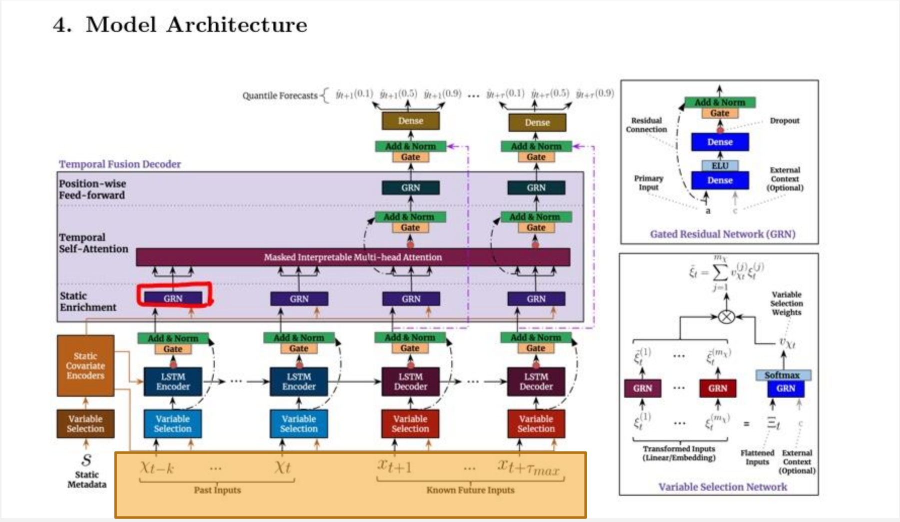
목차

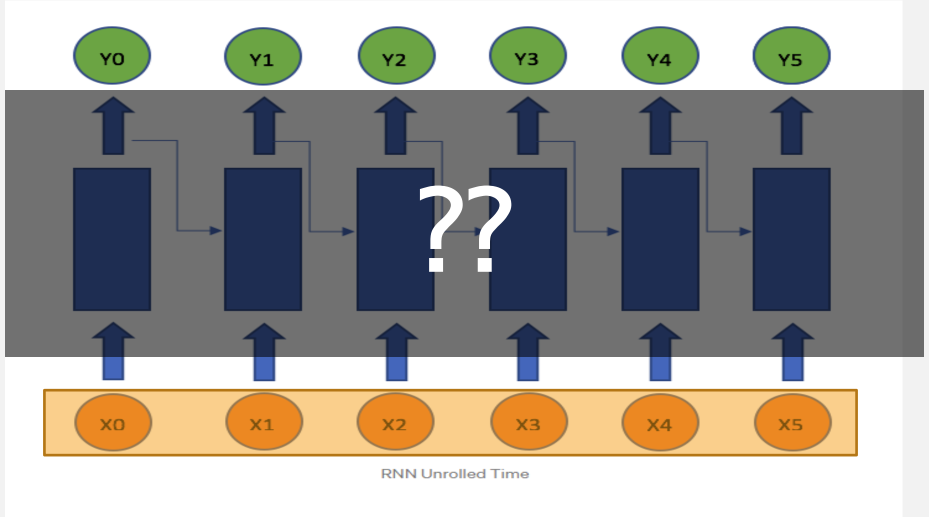
1. 연구 의의 및 목적

TFT VS RNN

ARCHITECTURE > GATING MECHANISMS
VARIABLE SELECTION NETWORKS
STATIC COVARIATE ENCODERS

ARCHITECTURE > GATING MECHANISMS ✕
VARIABLE SELECTION NETWORKS. ✕
STATIC COVARIATE ENCODERS ✕





input > Static covariates (contexts)
Observed inputs
Known inputs

input >
Observed inputs

연구 목적

- Forecasting 에 영향을 줄 수 있는 보다 유연하고 풍부한 데이터를 모두 활용할 수 있는 모델을 만들겠다

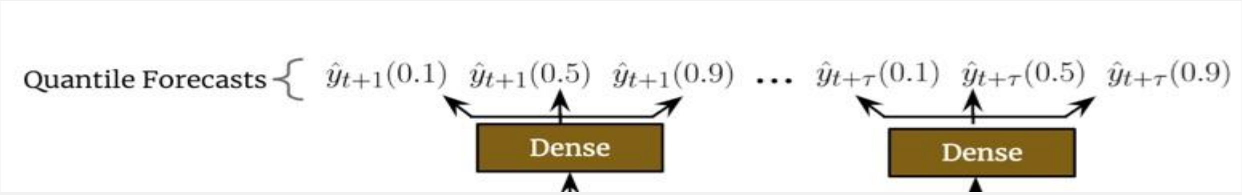
- 모델 forecasting 도중 해당 시점의 연산에서 필수적인 레이어와 features 만을 필터링하여 사용하겠다

- Multi-head attention 의 변형 방식으로 다양한 헤드를 앙상블 느낌으로 각 타임스텝의 관계성을 폭넓게 해석하겠다 (interpretability)

2. 용어 정리

용어 정리

- Horizon : 예측 범위 / Multi- Horizon : 여러 개의 예측 범위



t가 월요일이라고 한다면 화요일 예측 값 월요일 + *t* 시점의 예측 값

- Static (=time invariant) covariates : 독립 변수가 종속 변수에 미치는 효과에 영향을 줄 수 있는 변수 ex. A 수학 문제집과 수학 성적과의 관계에서 학생들의 원 수학 실력 => **메타데이터**
- Observed inputs (z), known inputs (x) ex. The way of week at time t

용어 정리

Horizon : 예측 범위 / Multi- Horizon : 여러 개의 예측 범위

Static (=time invariant) covariates : 독립 변수가 종속 변수에 미치는 효과에

영향을 줄 수 있는 변수 ex. A 수학 문제집과 수학

성적과의 관계에서 학생들의 원 수학 실력

=>메타데이터

Observed inputs (z), known inputs (x)

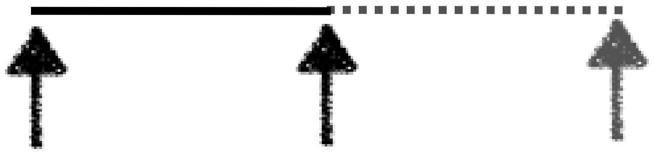
ex. The way of week at time t

$S_i \in \mathbb{R}^{m_s}$ Static covariates

$\chi_{i,t} = [z_{i,t}^T, x_{i,t}^T]$ Inputs ; (observed, known)

$y_{i,t} \in \mathbb{R}$ Outputs

$$\hat{y}_i(q,t,\tau) = f_q(\tau, y_{i,t-k:t}, z_{i,t-k:t}, x_{i,t-k:t+\tau}, s_i)$$



현재 시점 - k시점
; Window

현재 시점 (t)

t 시점 이후 예측값은?

이용할 variables 4가지

t-k : t 까지의
Outputs

t-k : t 까지의
측정 features

t-k : t + t 까지의
known features

Static
covariates
(메타 데이터)

=> features 를
이해할 수 있는
Context 로 같이
넣어줄 예정

3. 모델 구조

모델 핵심 요소 6가지

Gating Mechanisms

모델 구조

자유도 크게

자유도 작게

자유도 크게

자유도 작게

자유도 크게

자유도 작게

=> 정제된 features

Variable Selection Networks

S,X,Y 중 각 시점에 꼭 필요한 features 필터링

Static Covariate Encoders

S 메타 데이터를 features 를 이해할 수 있는 context화

Interpretable Multi-Head Attention

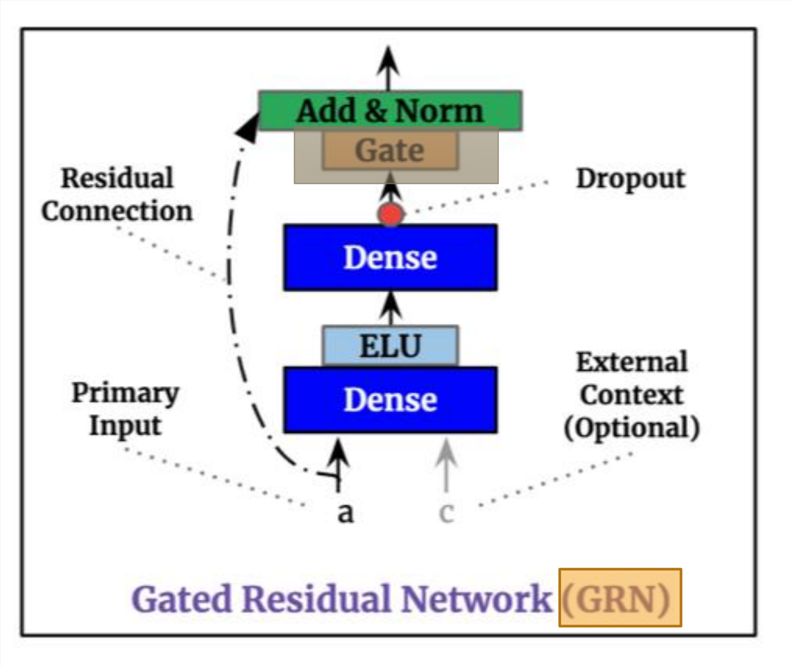
Temporal Fusion Decoder

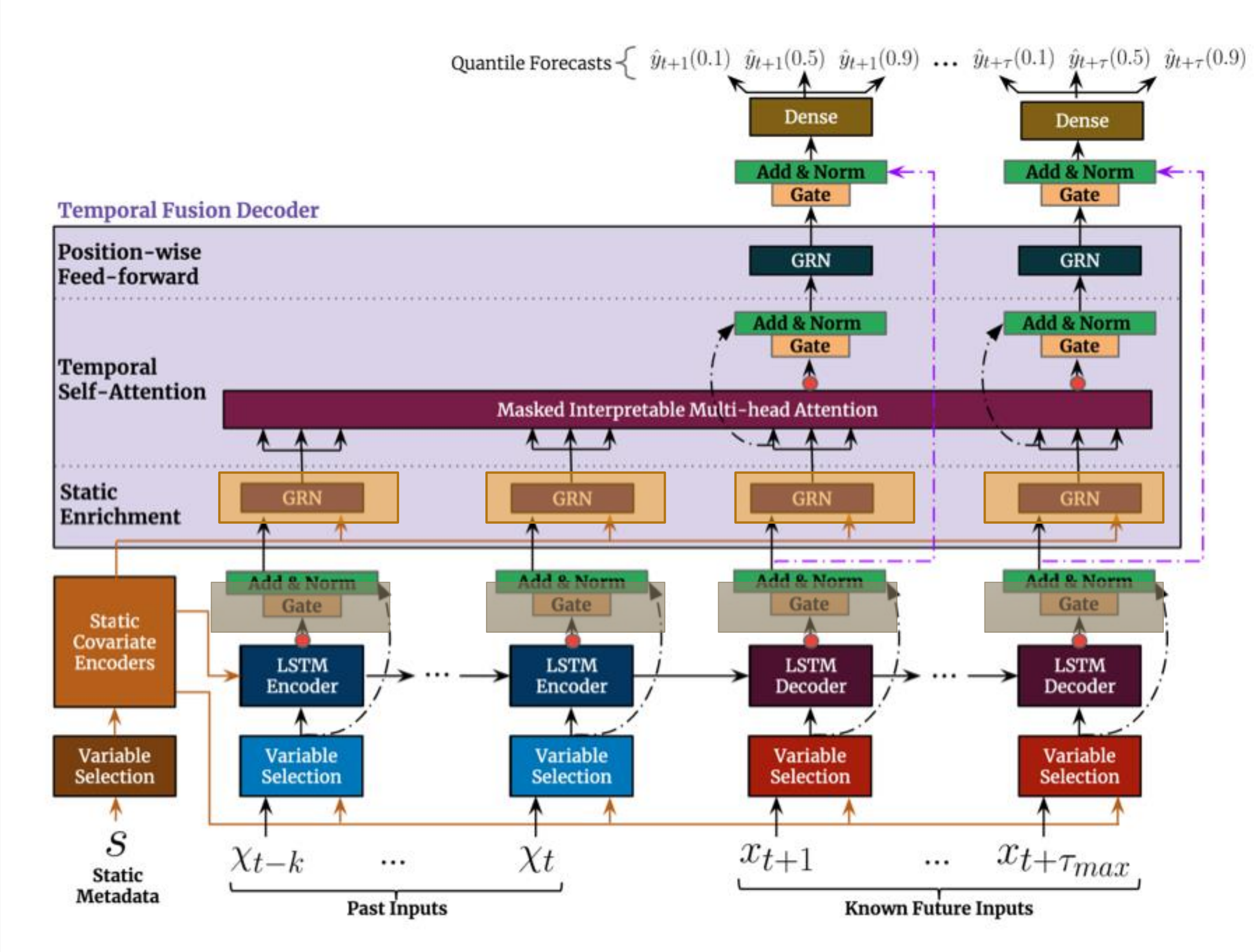
각 time step 의 장기간 상호관계 도출

Quantile Outputs

Gating Mechanisms = GRN layer

Gate 는 TFT 모델의 거의 모든 층에 사용 되는 핵심 테크닉
GRN layer 에도 gate 가 사용됨 !

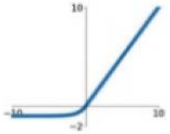




Gating Mechanisms = GRN layer

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



$$\eta_2 = \mathrm{ELU}\left(\boldsymbol{W}_{2,\omega}\, \boldsymbol{a} + \boldsymbol{W}_{3,\omega}\, \boldsymbol{c} + \boldsymbol{b}_{2,\omega}\right), \quad (4)$$

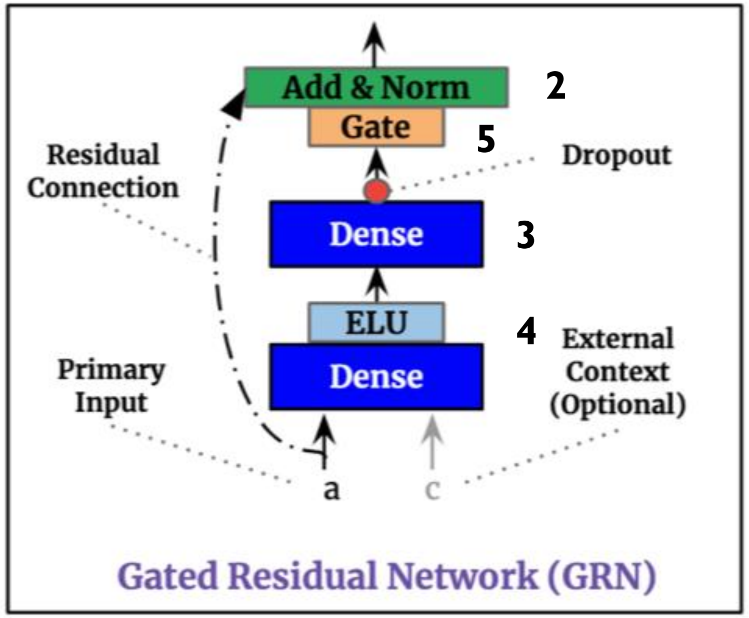
$$\eta_1 = \boldsymbol{W}_{1,\omega}\, \eta_2 + \boldsymbol{b}_{1,\omega}, \quad (3)$$

Gate

$$\mathrm{GLU}_\omega(\gamma) = \sigma(\boldsymbol{W}_{4,\omega}\, \gamma + \boldsymbol{b}_{4,\omega}) \odot (\boldsymbol{W}_{5,\omega}\, \gamma + \boldsymbol{b}_{5,\omega}), \quad (5)$$

Dropout (training)

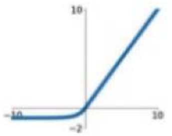
$$\mathrm{GRN}_\omega(\boldsymbol{a}, \boldsymbol{c}) = \mathrm{LayerNorm}\left(\boldsymbol{a} + \mathrm{GLU}_\omega(\eta_1)\right), \quad (2)$$



Gating Mechanisms = GRN layer

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



$$\eta_2 = \mathrm{ELU}\left(W_{2,\omega}\, a + W_{3,\omega}\, c + b_{2,\omega}\right), \quad (4)$$

$$\eta_1 = W_{1,\omega}\, \eta_2 + b_{1,\omega}, \quad (3)$$

$$\mathrm{GLU}_\omega(\gamma) = \sigma(W_{4,\omega}\, \gamma + b_{4,\omega}) \odot (W_{5,\omega}\, \gamma + b_{5,\omega}), \quad (5)$$

Gate

$$\mathrm{GRN}_\omega(a, c) = \mathrm{LayerNorm}\left(a + \mathrm{GLU}_\omega(\eta_1)\right), \quad (2)$$

(4)

>= 0 ?

W (4) + b (3)

W (3) + b (5)-2 * (5)-1 (5)

(4)

< 0 ?

W(αe^ (4) -1) + b (3)

W (3) + b (5)-2 * (5)-1 (5)

Final Output (2)

LN(a+ (5))

LN(a+ (5))

LN(a+ (5))

…

Depends on (5)-1

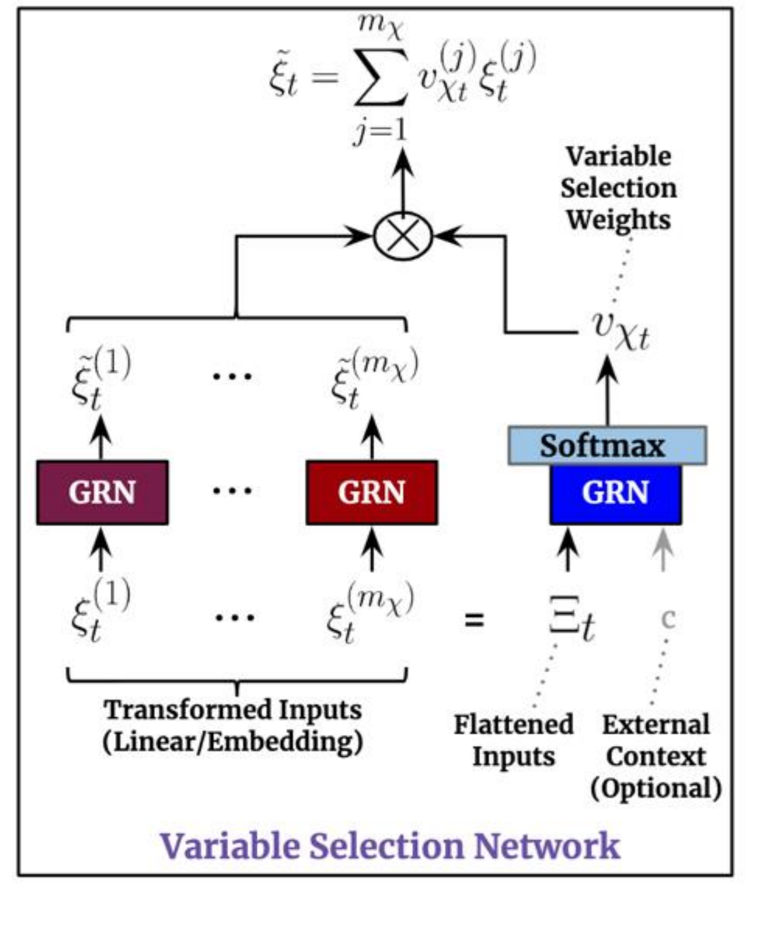
(4) 가 모델 복잡도 결정

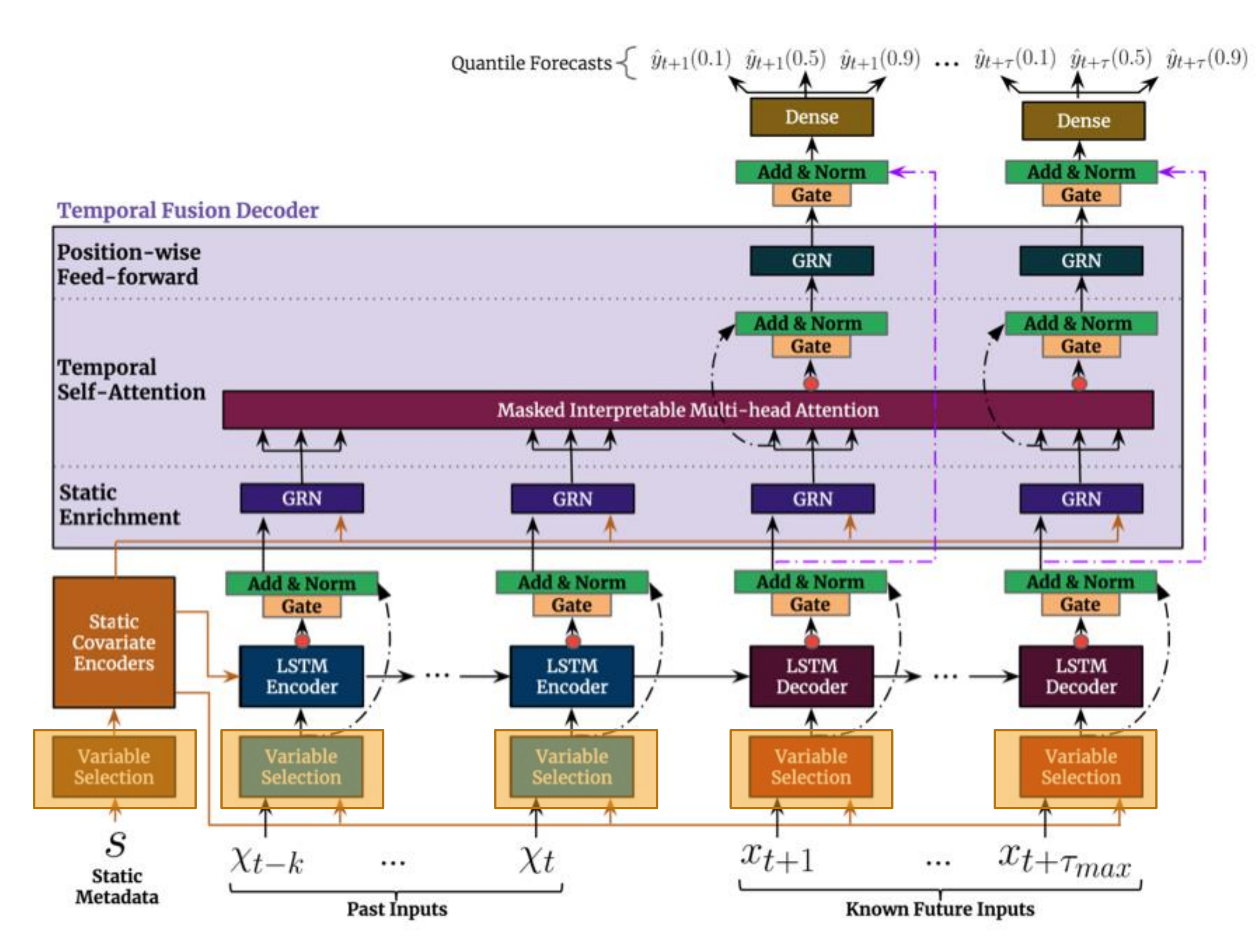
(5)-1 가 value scaling 역할

Variable Selection Networks = VSN

각 시점 input 의 여러 features 중
예측값에 확실히 관여하는 알맹이들만 남기기

VSN layer 는 GRN layer 을 포함
모든 inputs 는 VSN layer 을 거침





Variable Selection Networks = VSN

12월 아이스크림의 예측 판매량은 ?

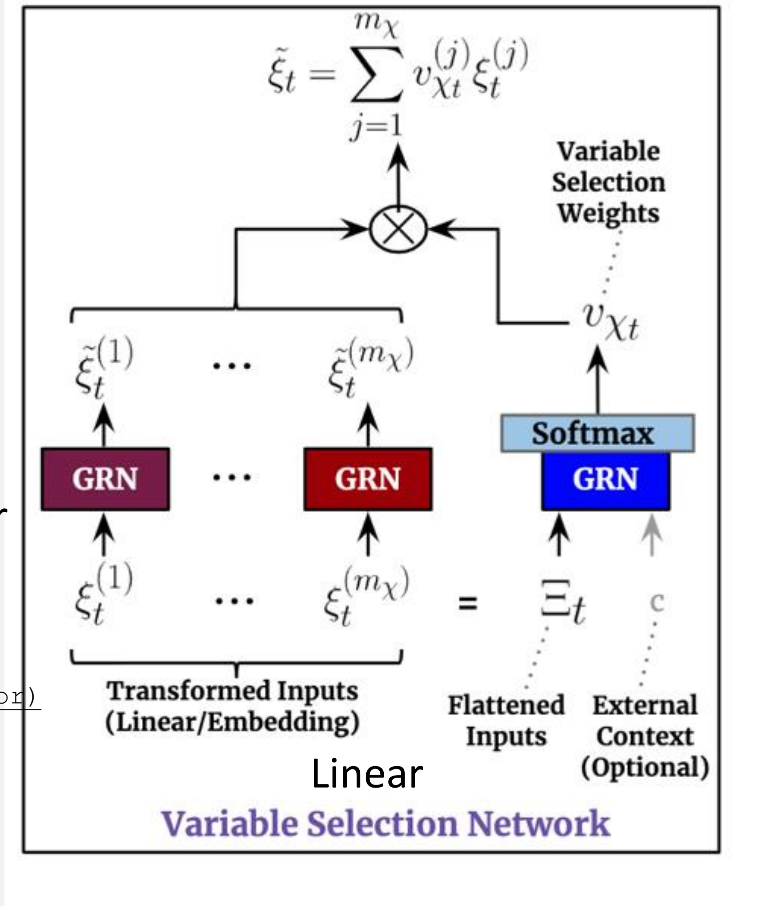
T 일 input 값

공휴일 여부	엄마는 외계인	민트초코
o	200개	100개

Categorical

O → Entity embedding (D Model vector)

Non-
Linear



Continuous

200 → Linear transformation nn.linear(1,D Model vector)

100 →

Flattened Inputs

→ Variable Selection Weights

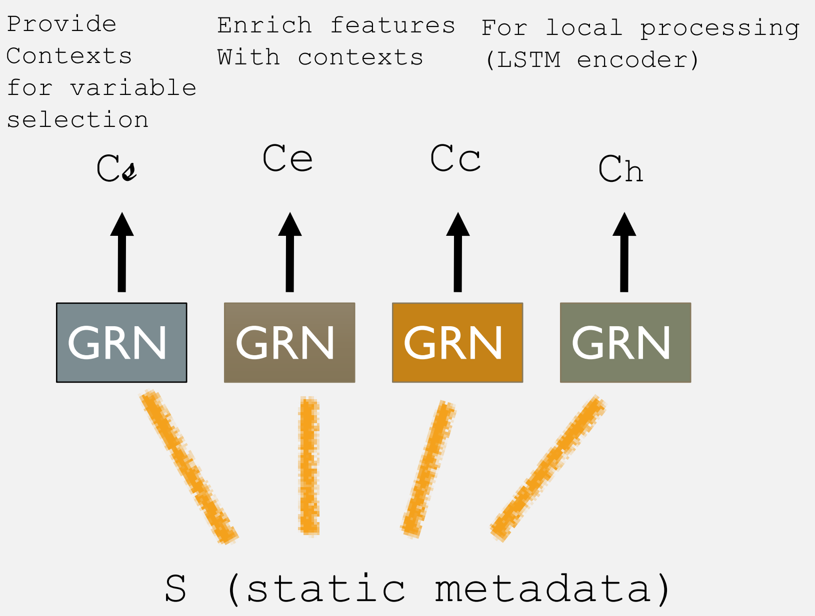
Feature 개수만큼 (j개)

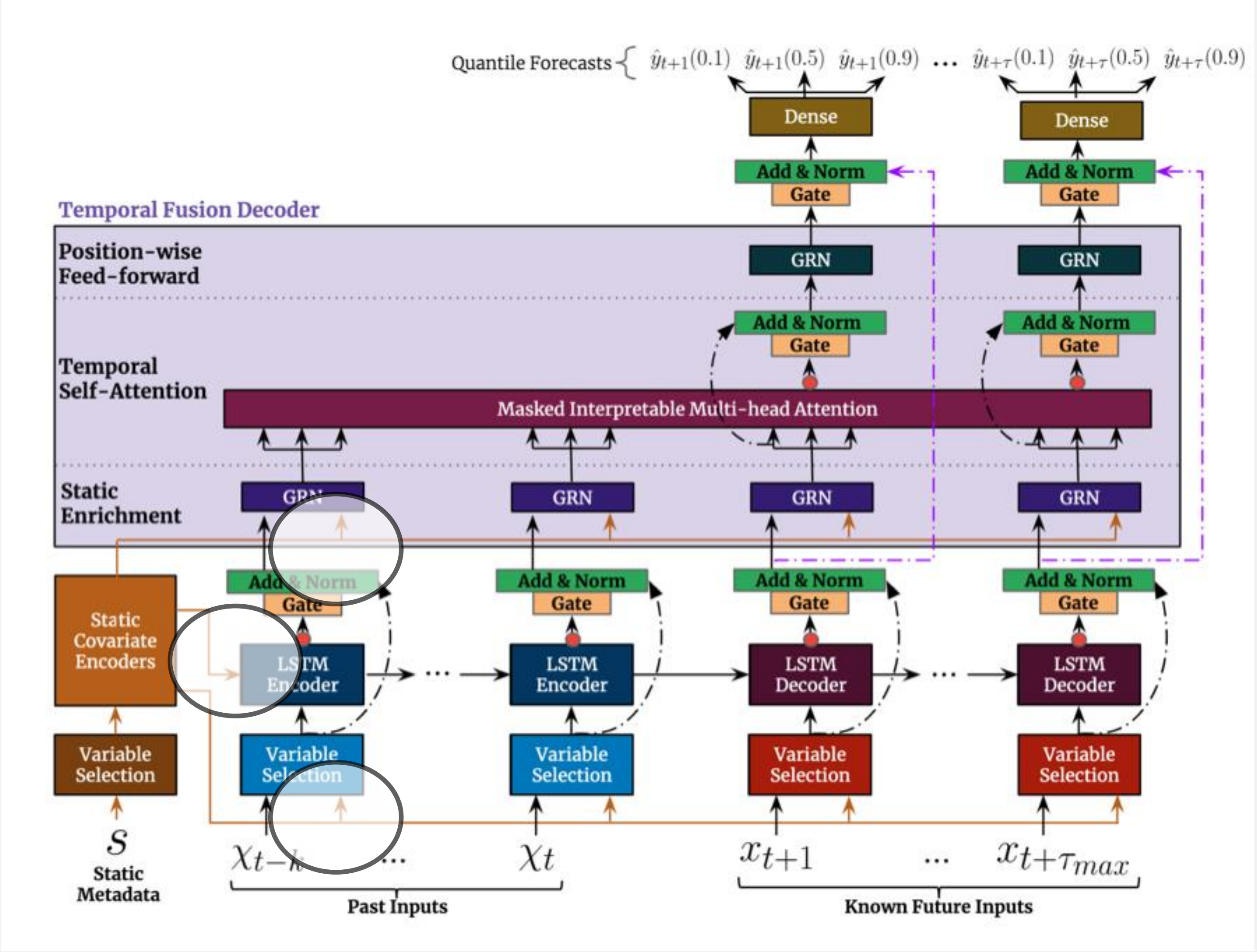
0.7	0.2	0.1

Static Covariate Encoders

S 메타 데이터를 features 을 이해할 수 있는 context 로 사용

각기 다른 4개의 GRN 을 사용하여서 쓰임이 다른 4개의 문맥 생성



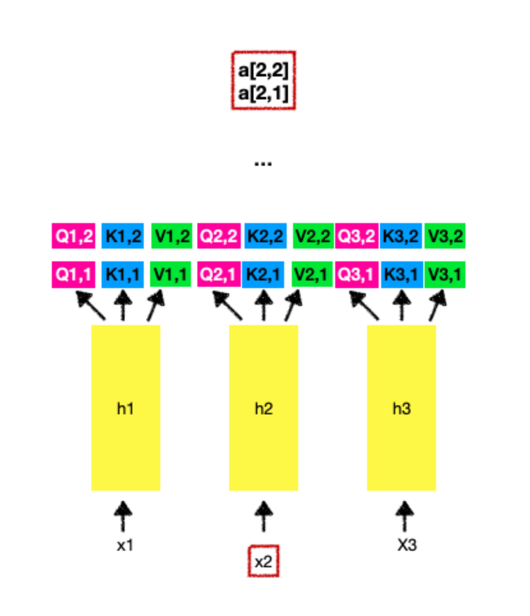


Interpretable Multi-Head Attention

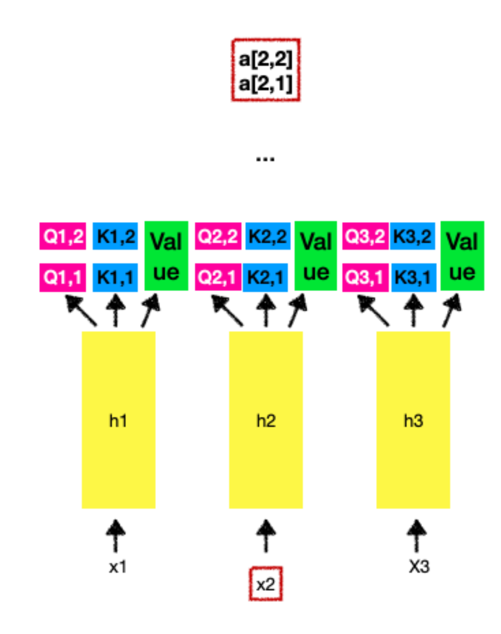
각 time step 의 장기간 상호관계 도출

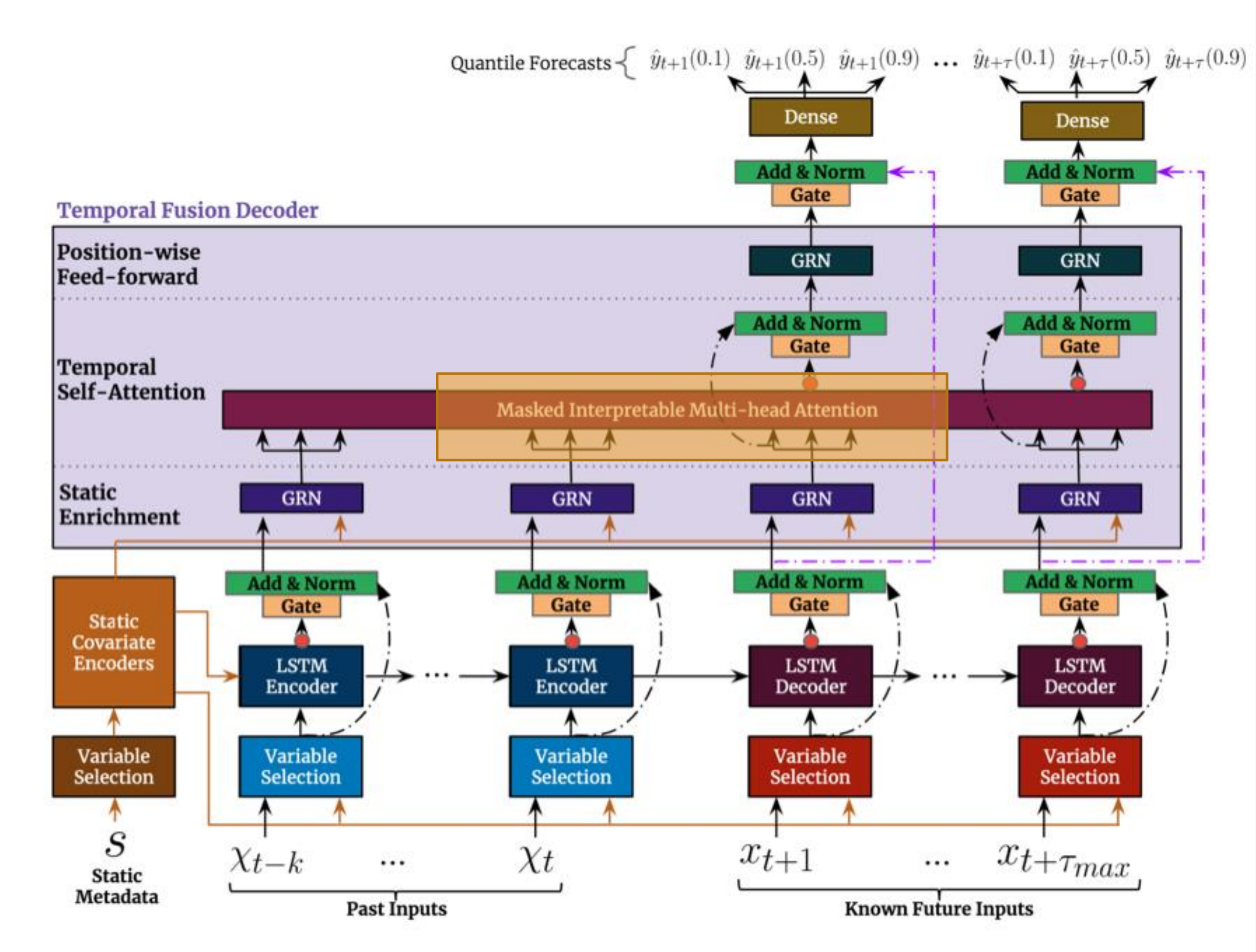
Multi-attention 아키텍처 그대로 갖고가되,
query,key,value 중 value 는 모든 head 에서 동일

원 Multi-head attention



TFT Multi-head attention





Interpretable Multi-Head Attention

각 time step 의 장기간 상호관계 도출

TFT Multi-head attention

$$\text{MultiHead}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = [\boldsymbol{H}_1, \dots, \boldsymbol{H}_{m_H}]\, \boldsymbol{W}_H, \tag{11}$$

$$\boldsymbol{H}_h = \text{Attention}(\boldsymbol{Q}\, \boldsymbol{W}_Q^{(h)}, \boldsymbol{K}\, \boldsymbol{W}_K^{(h)}, \boldsymbol{V}\, \boldsymbol{W}_V^{(h)}), \tag{12}$$

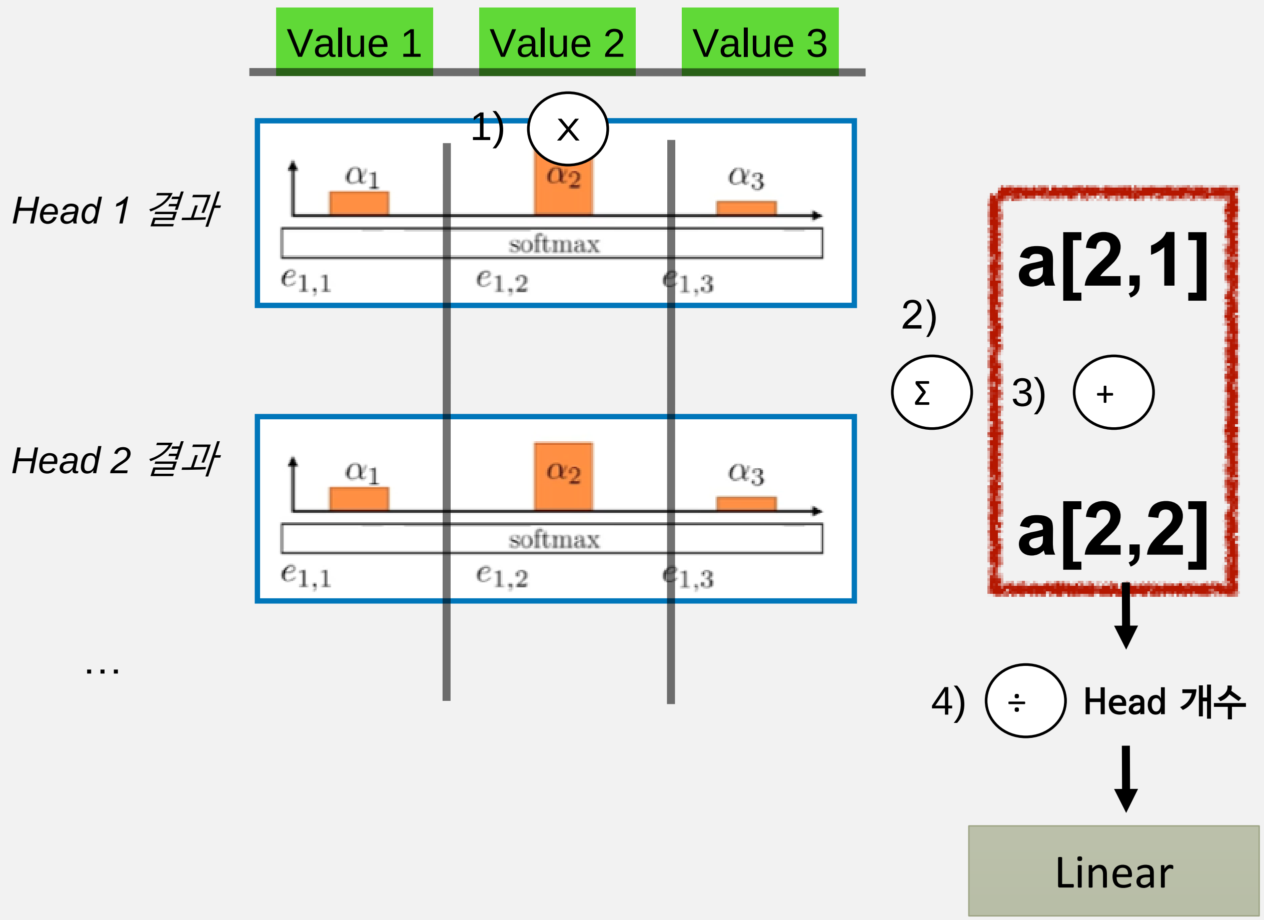
$$\text{InterpretableMultiHead}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \tilde{\boldsymbol{H}}\, \boldsymbol{W}_H, \tag{13}$$

$$\tilde{\boldsymbol{H}} = \tilde{A}(\boldsymbol{Q}, \boldsymbol{K})\, \boldsymbol{V}\, \boldsymbol{W}_V, \tag{14}$$

$$= \left\{ 1/H \sum_{h=1}^{m_H} A\left(\boldsymbol{Q}\, \boldsymbol{W}_Q^{(h)}, \boldsymbol{K}\, \boldsymbol{W}_K^{(h)}\right) \right\} \boldsymbol{V}\, \boldsymbol{W}_V, \tag{15}$$

$$= 1/H \sum_{h=1}^{m_H} \text{Attention}(\boldsymbol{Q}\, \boldsymbol{W}_Q^{(h)}, \boldsymbol{K}\, \boldsymbol{W}_K^{(h)}, \boldsymbol{V}\, \boldsymbol{W}_V), \tag{16}$$

같은 timestep 은 다른 head 에서도
동일한 value 를 갖게 함으로써 앙상블하는 방식으로 작용



ex. Timestep 2 의 어텐션 결과

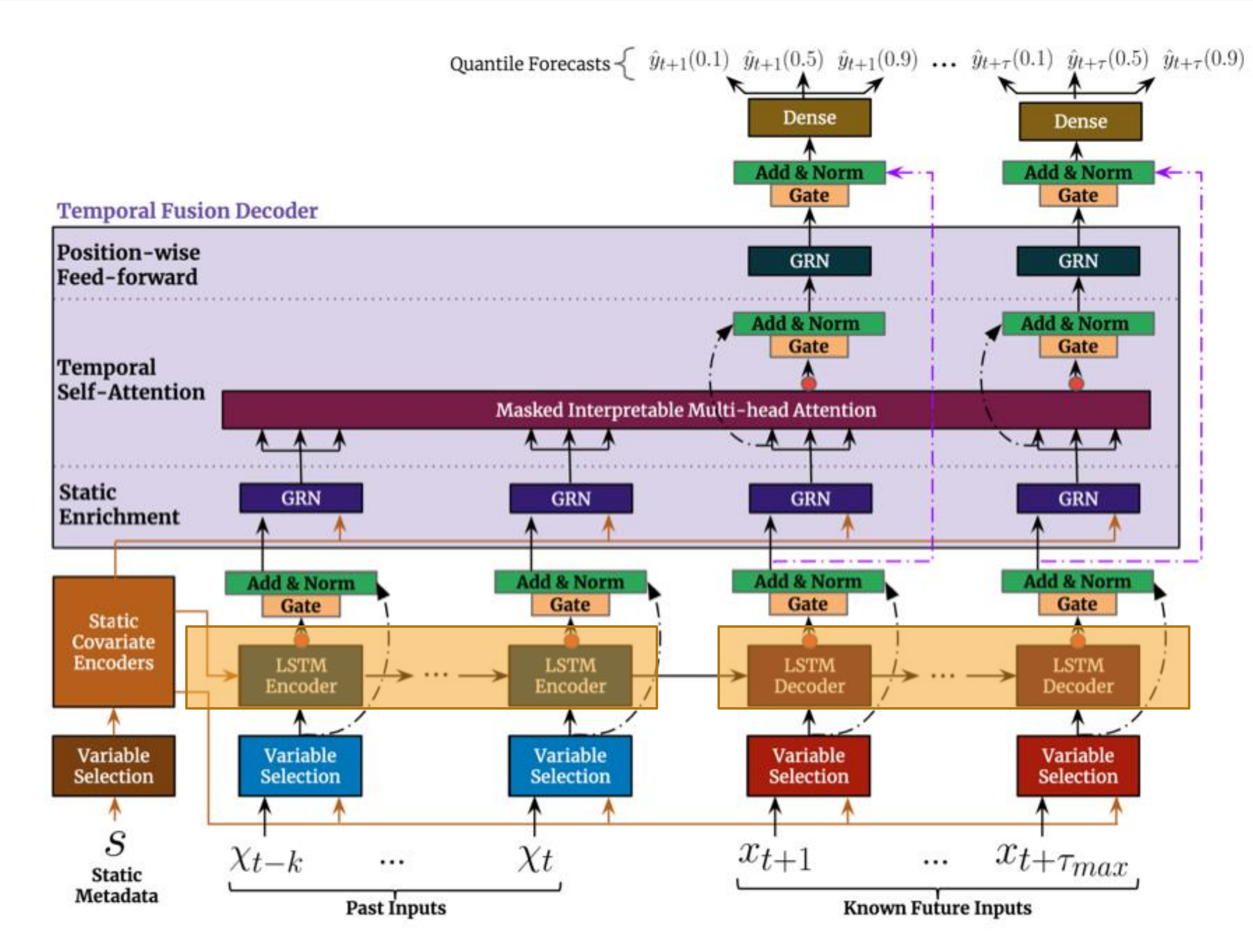
=> 앙상블 느낌 !

Temporal Fusion Decoder - (1)
Seq2Seq layer

Temporal Fusion Decoder 에 들어가는 final inputs

각 시점의 특징 추출, 각 시점 정보 추가

$$\tilde{\phi}(t,n) = \text{LayerNorm}\left(\tilde{\xi}_{t+n} + \text{GLU}_{\tilde{\phi}}(\phi(t,n))\right), \quad (17)$$

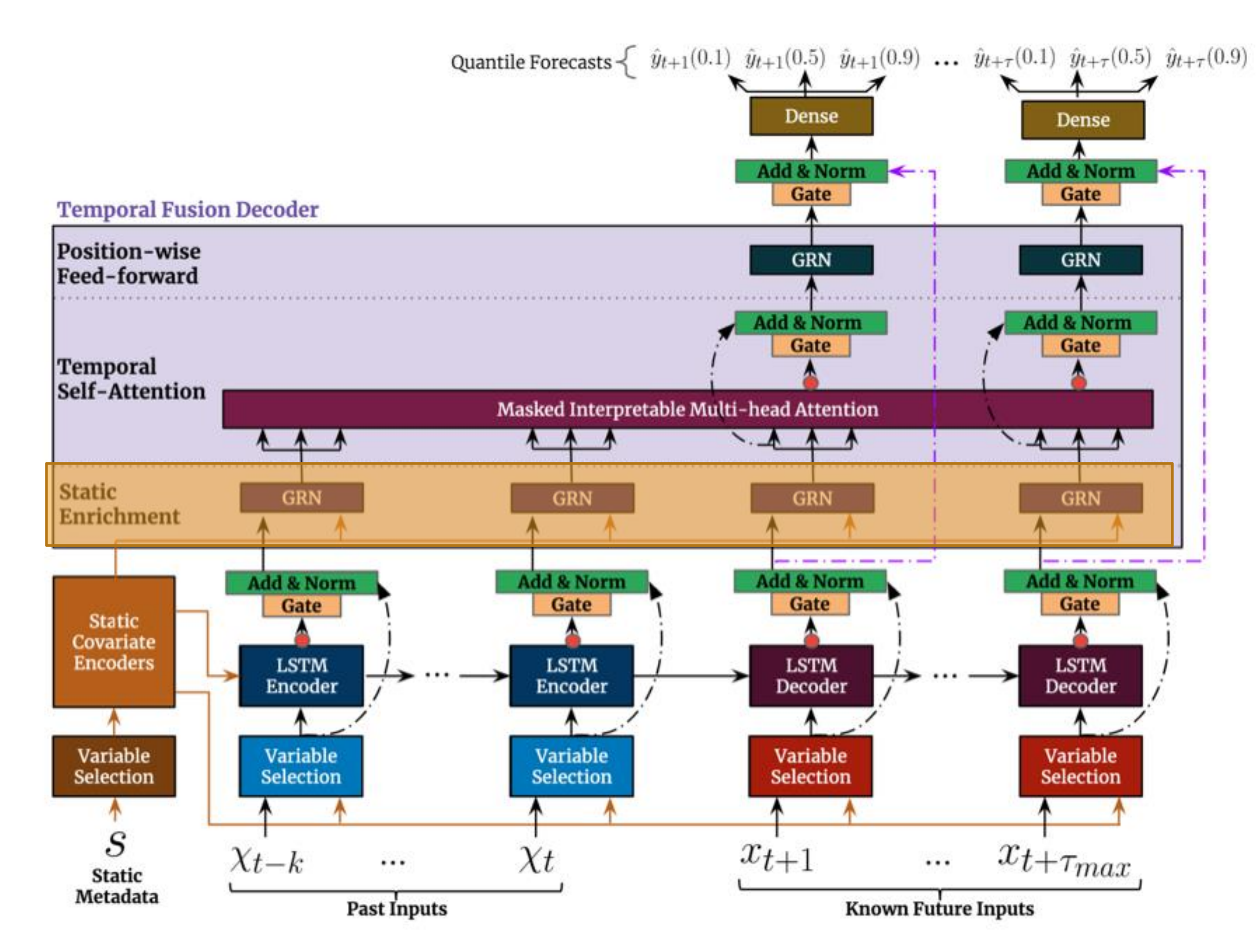


Temporal Fusion Decoder - (2)
Static Enrichment Layer

Temporal Fusion Decoder 에 들어가는 final inputs

temporal features 에 메타데이터를 활용해 풍부한 문맥 추가

$$\boldsymbol{\theta}(t,n) = \mathrm{GRN}_{\theta}\left(\tilde{\boldsymbol{\phi}}(t,n), \boldsymbol{c}_e\right), \tag{18}$$

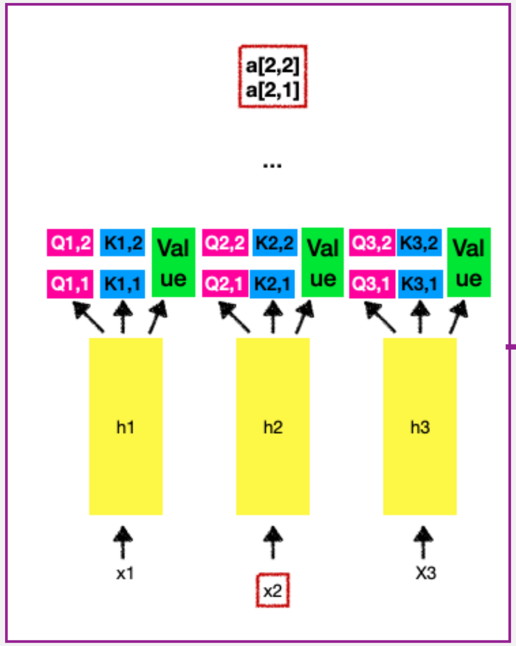


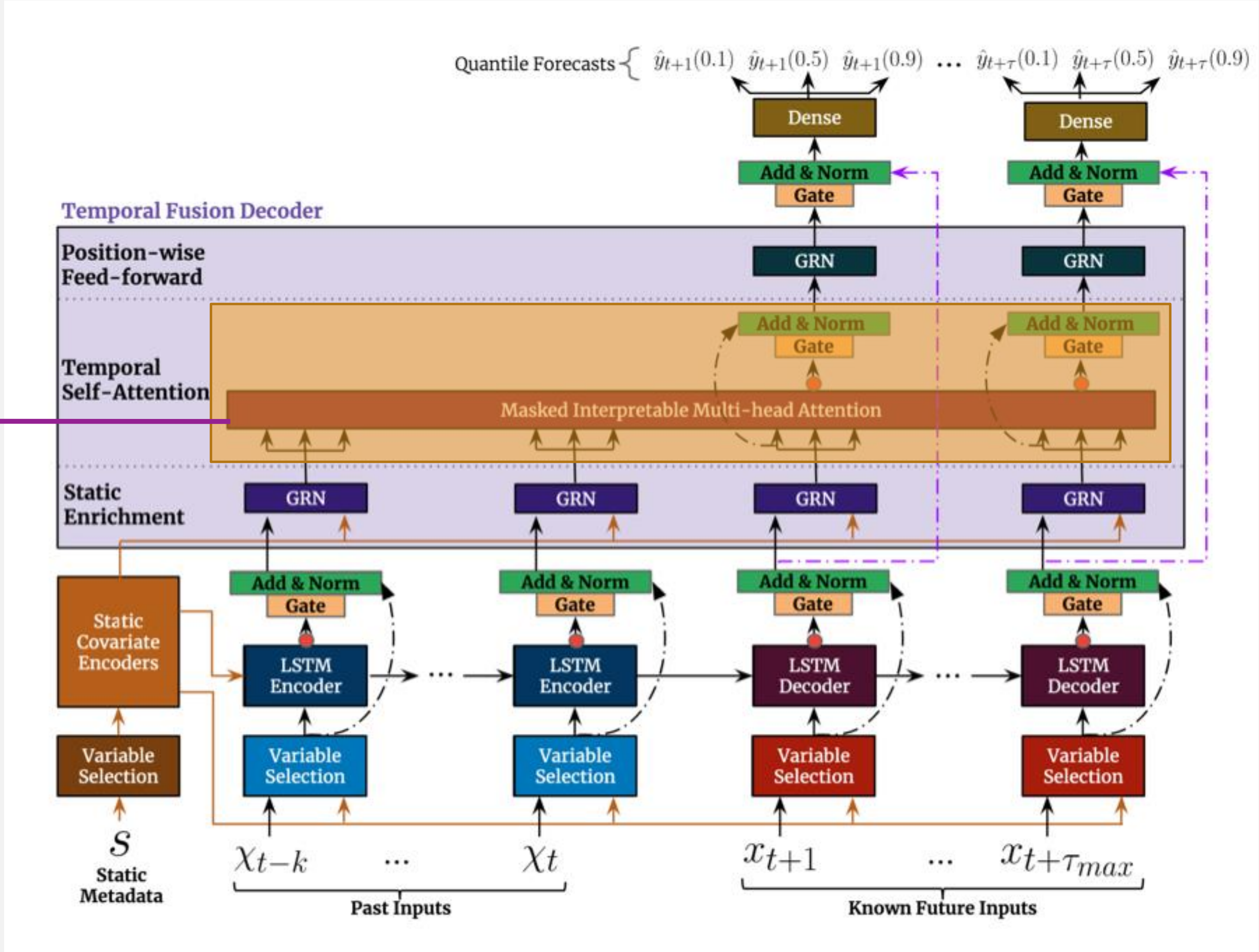
Temporal Fusion Decoder - (3)
Temporal Self-Attention Layer

Interpretable Multi-Head Attention 이 투입된 layer

각 time step 의 장기간 상호관계 도출

TFT Multi-head attention





Temporal Fusion Decoder - (3)
Temporal Self-Attention Layer

Interpretable Multi-Head Attention 이 투입된 layer

각 time step 의 장기간 상호관계 도출

$$\theta(t) = [\theta(t,-k), \ldots, \theta(t,\tau)]^T$$

Static enrichment layer 에서 나온 값들을 하나의 단일 벡터로 뭉치기

↓

TFT masked Multi-head attention : **이전 시점들과의 관계(;attention) 만을 이용하도록 하기 위해서**

↓

$$\delta(t,n) = \text{LayerNorm}(\theta(t,n) + \text{GLU}_\delta(\beta(t,n))). \quad (20)$$

n > t

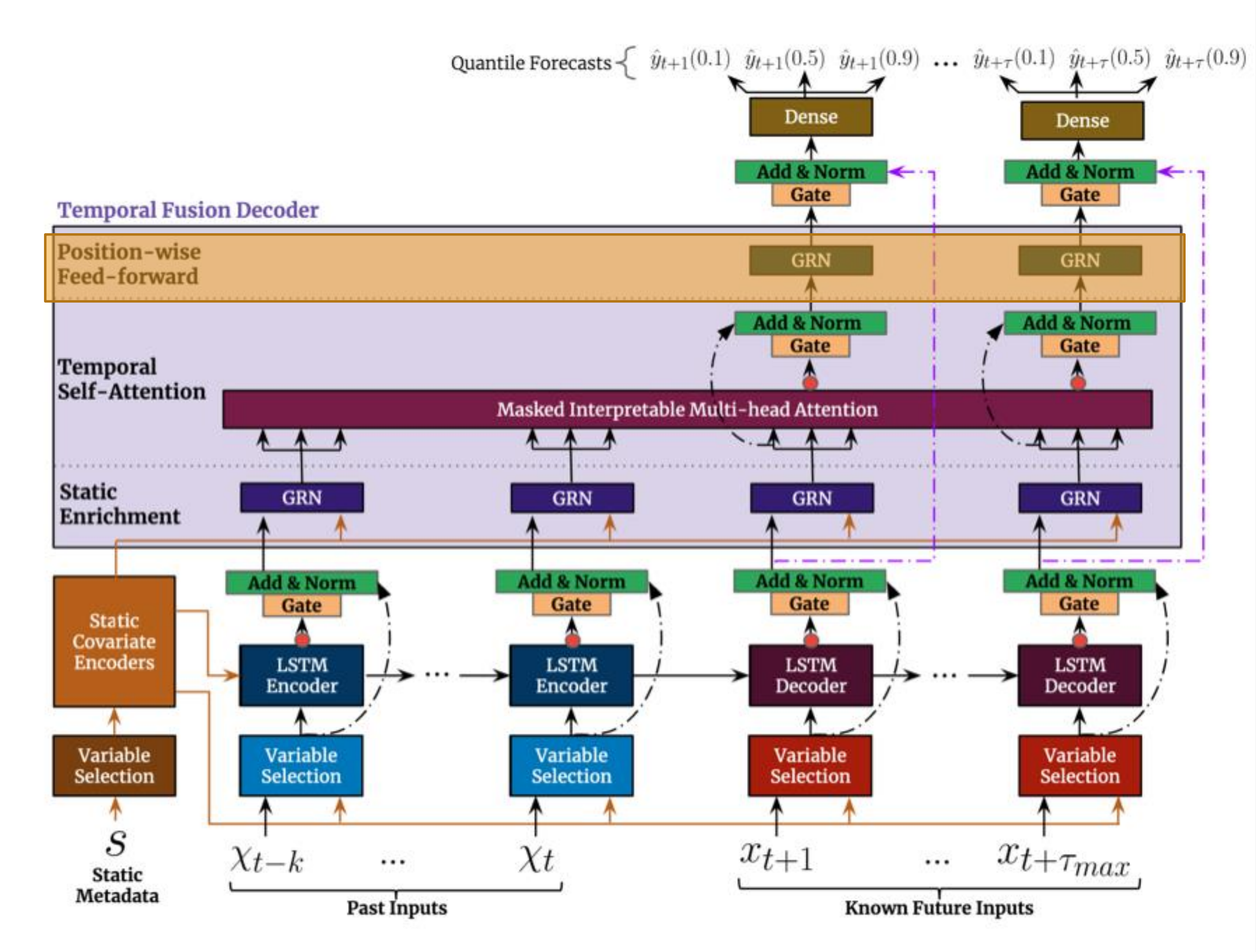
Temporal Fusion Decoder - (4)
Position-wise Feed-forward layer

n < t

non-linear 층 (GRN) 추가

$$\psi(t,n) = \text{GRN}_{\psi}\left(\boldsymbol{\delta}(t,n)\right), \tag{21}$$

$$\tilde{\psi}(t,n) = \text{LayerNorm}\left(\tilde{\phi}(t,n) + \text{GLU}_{\tilde{\psi}}(\psi(t,n))\right), \tag{22}$$

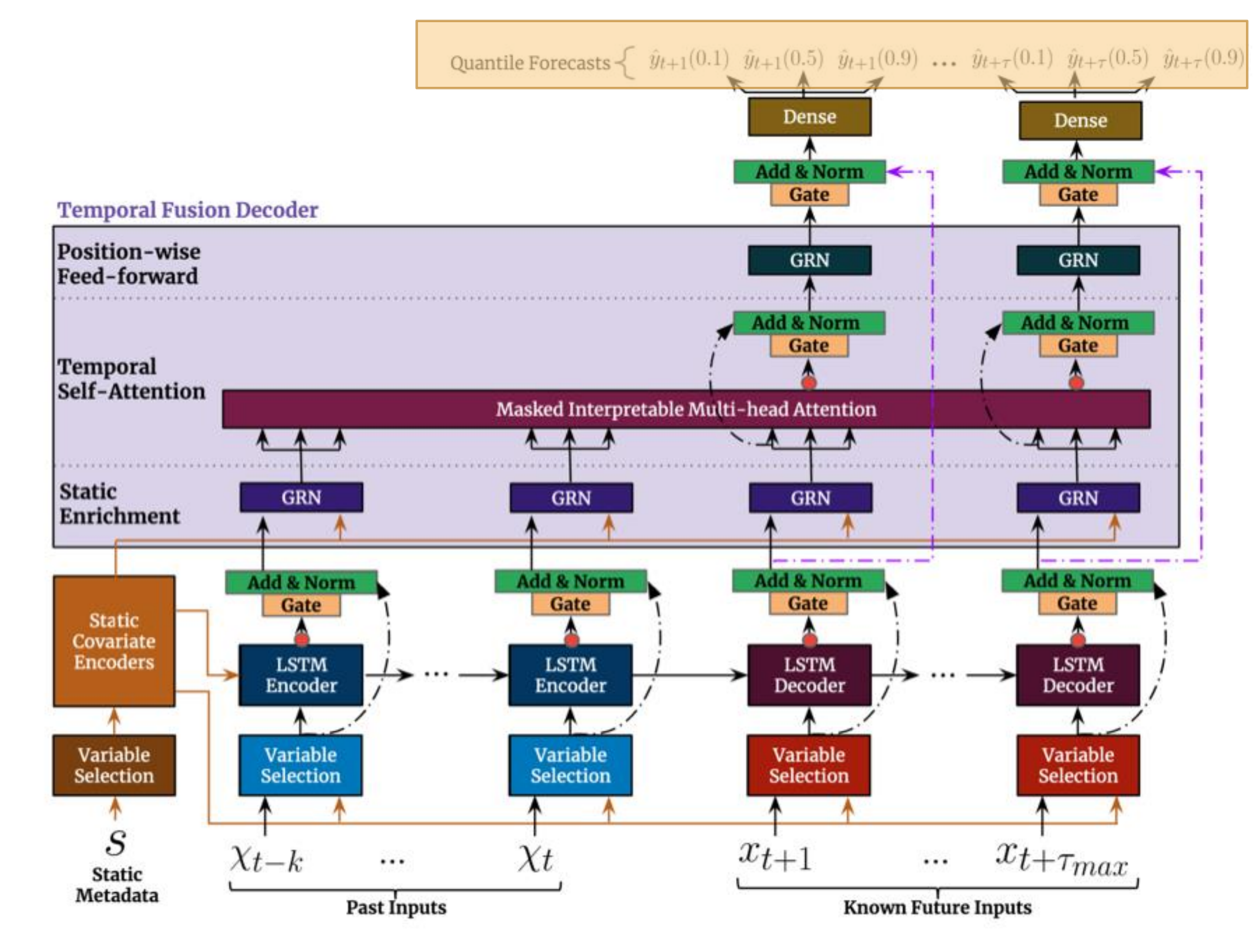


Quantile Outputs

최종 quantile 확률별 outputs 출력

$$\hat{y}(q, t, \tau) = W_q \tilde{\psi}(t, \tau) + b_q, \quad (23)$$

각 quantile 마다 각자의 linear 층으로 output 값 계산
Quantile : 해당 예측값이 나올 확률이 quantile %



4. LOSS FUNCTION

LOSS FUNCTION

Quantile Forecasts $\{ \hat{y}_{t+1}(0.1) \ \hat{y}_{t+1}(0.5) \ \hat{y}_{t+1}(0.9) \ \ldots \ \hat{y}_{t+\tau}(0.1) \ \hat{y}_{t+\tau}(0.5) \ \hat{y}_{t+\tau}(0.9)$

해당 데이터셋 内

$$\mathcal{L}(\Omega, W) = \sum_{y_t \in \Omega} \sum_{q \in Q} \sum_{\tau=1}^{\tau_{max}} \frac{QL\left(y_t, \ \hat{y}(q, t-\tau, \tau), \ q\right)}{M \tau_{max}} \tag{24}$$

{0.1,0.5,0.9}

예측할 개수

$$QL(y, \ \hat{y}, \ q) = q(y - \hat{y})_+ + (1-q)(\hat{y} - y)_+, \tag{25}$$

max(0,x)

- Quantile loss function 출처 : https://arxiv.org/pdf/1711.11053.pdf
- 조금 변형된 loss function 으로 out of sample test 도 같이 진행

$$q\text{-Risk} = \frac{2 \sum_{y_t \in \tilde{\Omega}} \sum_{\tau=1}^{\tau_{max}} QL\left(y_t, \ \hat{y}(q, t-\tau, \tau), \ q\right)}{\sum_{y_t \in \tilde{\Omega}} \sum_{\tau=1}^{\tau_{max}} |y_t|}, \tag{26}$$

where $\tilde{\Omega}$ is the domain of test samples. Full details on hyperparameter optimization and training can be found in Appendix A.

5. 데이터 셋 / 실험 결과

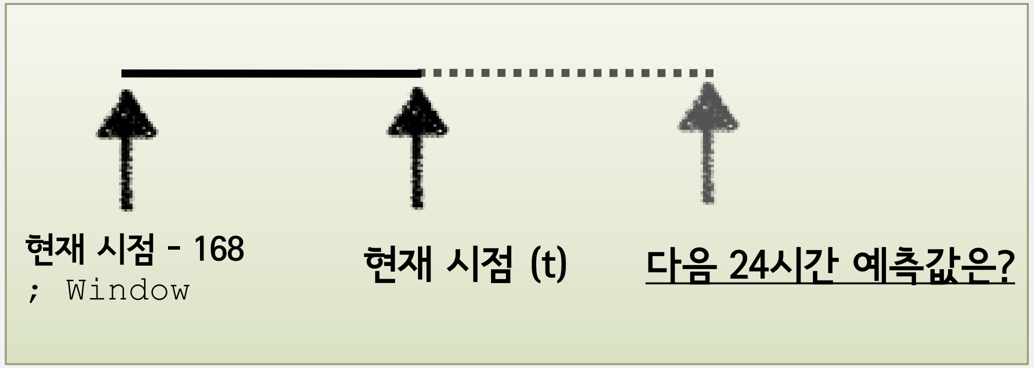
데이터 셋 1 – The UCI electricity Load Diagrams Dataset

Electricity consumption of 370 customers (168 x 370)

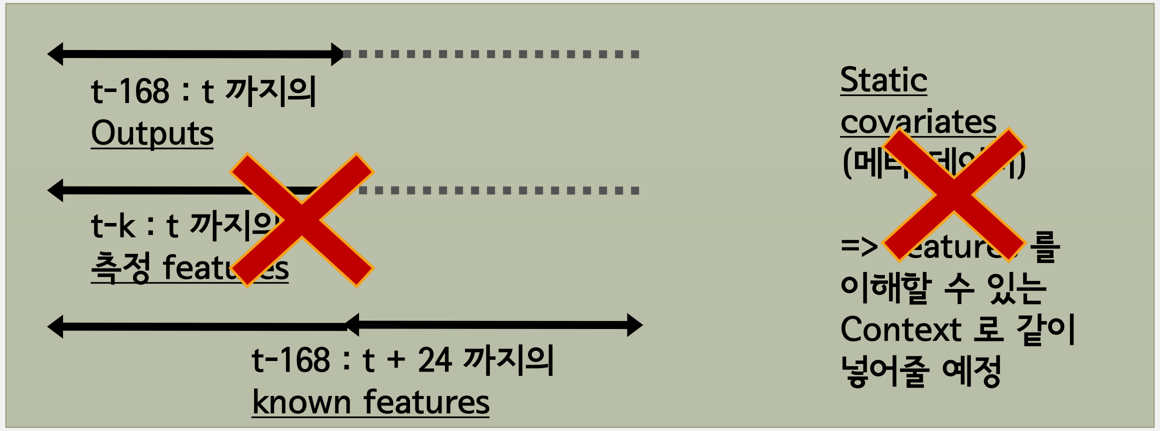
370 명 (열)

총 168 시간 (행) – window

```
_column_definition = [
    ('id', DataTypes.REAL_VALUED, InputTypes.ID),
    ('hours_from_start', DataTypes.REAL_VALUED, InputTypes.TIME),
    ('power_usage', DataTypes.REAL_VALUED, InputTypes.TARGET),
    ('hour', DataTypes.REAL_VALUED, InputTypes.KNOWN_INPUT),
    ('day_of_week', DataTypes.REAL_VALUED, InputTypes.KNOWN_INPUT),
    ('hours_from_start', DataTypes.REAL_VALUED, InputTypes.KNOWN_INPUT),
    ('categorical_id', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT),
]
```



이용할 features ~~4가지~~ 2가지

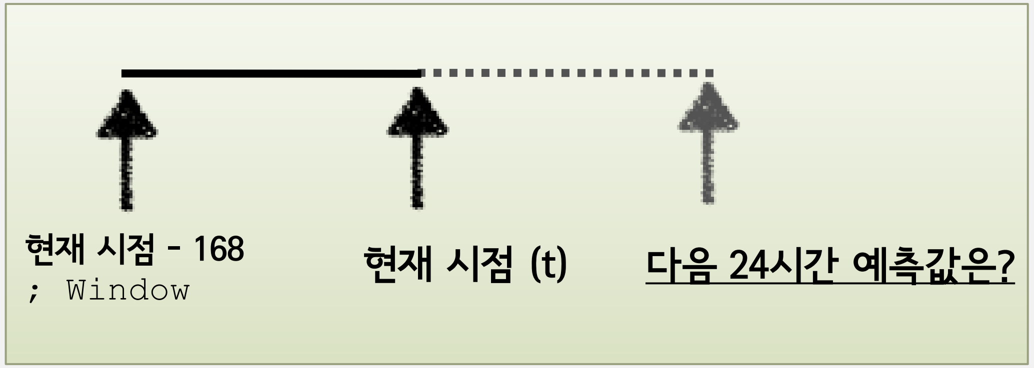


데이터 셋 2 – The UCI PEM-SF Traffic Dataset

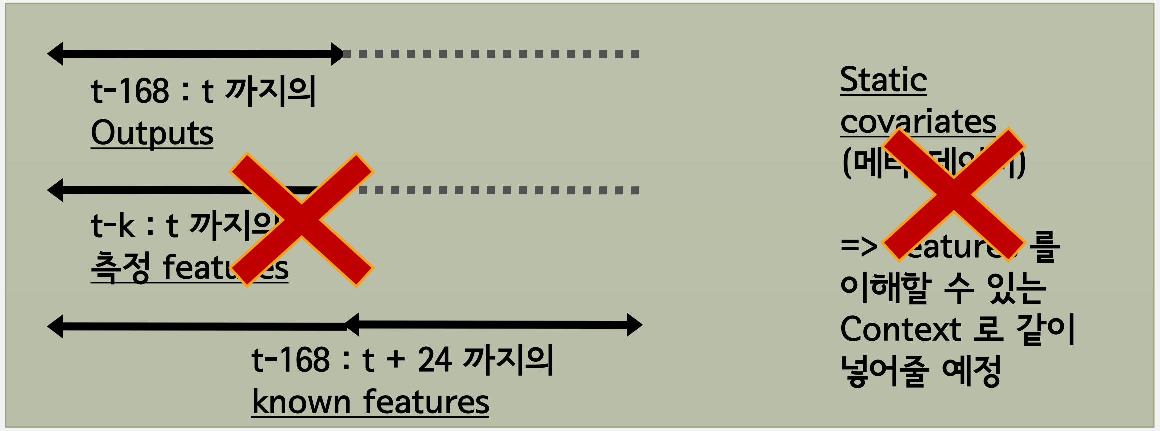
440 route freeways (열)

총 168 시간 (행) – window

```
_column_definition = [
    ('id', DataTypes.REAL_VALUED, InputTypes.ID),
    ('hours_from_start', DataTypes.REAL_VALUED, InputTypes.TIME),
    ('values', DataTypes.REAL_VALUED, InputTypes.TARGET),
    ('time_on_day', DataTypes.REAL_VALUED, InputTypes.KNOWN_INPUT),
    ('day_of_week', DataTypes.REAL_VALUED, InputTypes.KNOWN_INPUT),
    ('hours_from_start', DataTypes.REAL_VALUED, InputTypes.KNOWN_INPUT),
    ('categorical_id', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT),
]
```



이용할 features ~~4가지~~ 2가지



데이터 셋 3 – Favorita Grocery Sales Dataset

메타데이터 4개

총 8개의 dataset https://www.kaggle.com/c/favorita-grocery-sales-forecasting/rules

items

item_nbr	family	class	perishable
96995	GROCERY I	1093	0
99197	GROCERY I	1067	0
103501	CLEANING	3008	0
103520	GROCERY I	1028	0
103665	BREAD/BAKERY	2712	1
105574	GROCERY I	1045	0
105575	GROCERY I	1045	0
105576	GROCERY I	1045	0
105577	GROCERY I	1045	0
105693	GROCERY I	1034	0

stores

store_nbr	city	state	type	cluster
1	Quito	Pichincha	D	13
2	Quito	Pichincha	D	13
3	Quito	Pichincha	D	8
4	Quito	Pichincha	D	9
5	Santo Domingo	Santo Domingo de los Tsachilas	D	4
6	Quito	Pichincha	D	13
7	Quito	Pichincha	D	8
8	Quito	Pichincha	D	8
9	Quito	Pichincha	B	6
10	Quito	Pichincha	C	15

oil

date	dcoilwtico
2013-01-01	
2013-01-02	93.14
2013-01-03	92.97
2013-01-04	93.12
2013-01-07	93.2
2013-01-08	93.21
2013-01-09	93.08
2013-01-10	93.81
2013-01-11	93.6
2013-01-14	94.27
2013-01-15	93.26

holidays_events

date	type	locale	locale_name	description	transferred
2012-03-02	Holiday	Local	Manta	Fundacion de Manta	FALSE
2012-04-01	Holiday	Regional	Cotopaxi	Provincializacion de Cotopaxi	FALSE
2012-04-12	Holiday	Local	Cuenca	Fundacion de Cuenca	FALSE
2012-04-14	Holiday	Local	Libertad	Cantonizacion de Libertad	FALSE
2012-04-21	Holiday	Local	Riobamba	Cantonizacion de Riobamba	FALSE
2012-05-12	Holiday	Local	Puyo	Cantonizacion del Puyo	FALSE
2012-06-23	Holiday	Local	Guaranda	Cantonizacion de Guaranda	FALSE
2012-06-25	Holiday	Regional	Imbabura	Provincializacion de Imbabura	FALSE
2012-06-25	Holiday	Local	Latacunga	Cantonizacion de Latacunga	FALSE

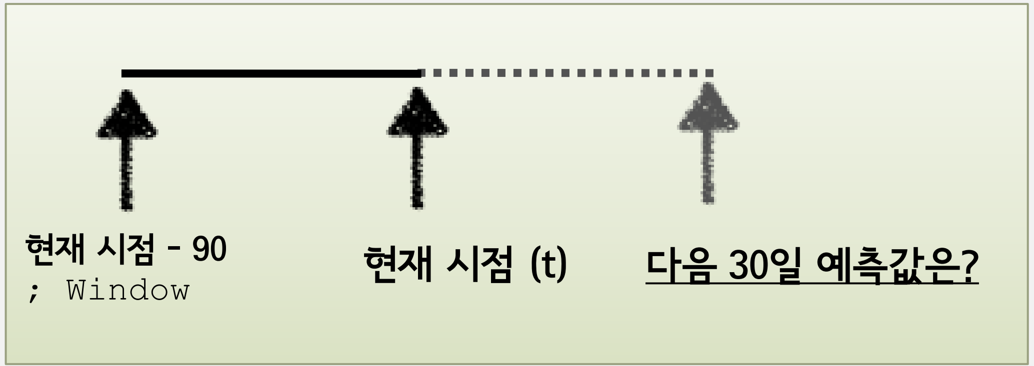
데이터 셋 3 – Favorita Grocery Sales Dataset

Training, Test dataset

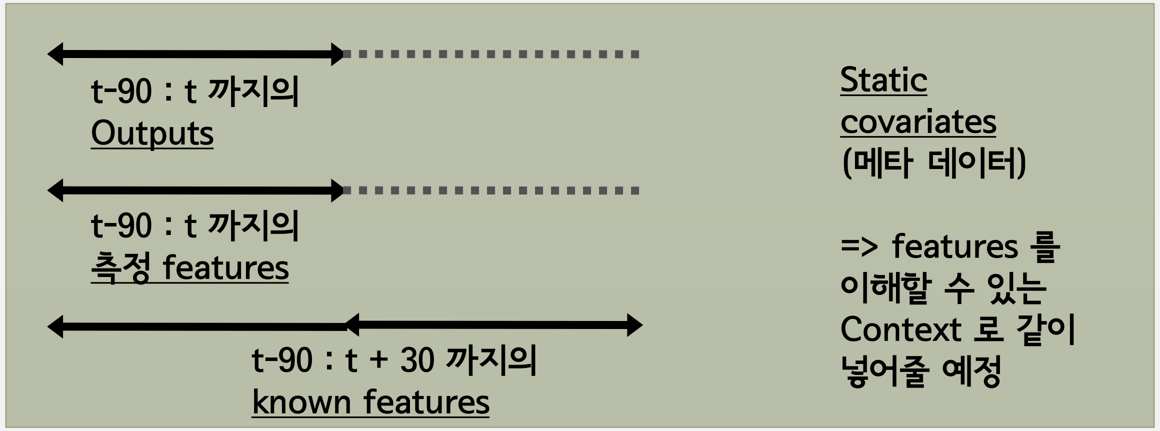
```
_column_definition = [
    ('traj_id', DataTypes.REAL_VALUED, InputTypes.ID),
    ('date', DataTypes.DATE, InputTypes.TIME),
    ('log_sales', DataTypes.REAL_VALUED, InputTypes.TARGET),
    ('onpromotion', DataTypes.CATEGORICAL, InputTypes.KNOWN_INPUT),
    ('transactions', DataTypes.REAL_VALUED, InputTypes.OBSERVED_INPUT),
    ('oil', DataTypes.REAL_VALUED, InputTypes.OBSERVED_INPUT),
    ('day_of_week', DataTypes.CATEGORICAL, InputTypes.KNOWN_INPUT),
    ('day_of_month', DataTypes.REAL_VALUED, InputTypes.KNOWN_INPUT),
    ('month', DataTypes.REAL_VALUED, InputTypes.KNOWN_INPUT),
    ('national_hol', DataTypes.CATEGORICAL, InputTypes.KNOWN_INPUT),
    ('regional_hol', DataTypes.CATEGORICAL, InputTypes.KNOWN_INPUT),
    ('local_hol', DataTypes.CATEGORICAL, InputTypes.KNOWN_INPUT),
    ('open', DataTypes.REAL_VALUED, InputTypes.KNOWN_INPUT),
    ('item_nbr', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT),
    ('store_nbr', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT),
    ('city', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT),
    ('state', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT),
    ('type', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT),
    ('cluster', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT),
    ('family', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT),
    ('class', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT),
    ('perishable', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT)
]
```

총 90 일 (행) – window

…



이용할 features 4가지



데이터 셋 4 – The OMI realized library

시간 간격 불규칙적

Training Set

31 Stock indices (열)

총 252 일 (행) – window

```
_column_definition = [
    ('Symbol', DataTypes.CATEGORICAL, InputTypes.ID),
    ('date', DataTypes.DATE, InputTypes.TIME),
    ('log_vol', DataTypes.REAL_VALUED, InputTypes.TARGET),
    ('open_to_close', DataTypes.REAL_VALUED, InputTypes.OBSERVED_INPUT),
    ('days_from_start', DataTypes.REAL_VALUED, InputTypes.KNOWN_INPUT),
    ('day_of_week', DataTypes.CATEGORICAL, InputTypes.KNOWN_INPUT),
    ('day_of_month', DataTypes.CATEGORICAL, InputTypes.KNOWN_INPUT),
    ('week_of_year', DataTypes.CATEGORICAL, InputTypes.KNOWN_INPUT),
    ('month', DataTypes.CATEGORICAL, InputTypes.KNOWN_INPUT),
    ('Region', DataTypes.CATEGORICAL, InputTypes.STATIC_INPUT),
]
```

...

현재 시점 - 252 ; Window

현재 시점 (t)

다음 5일 예측값은?

features ~~4가지~~ 2가지

t-252 : t 까지의 Outputs

t-k : t 까지의 측정 features

t-252 : t + 5 까지의 known features

~~Static covariates (메타데이터)~~

~~=> features 를~~ 이해할 수 있는 Context 로 같이 넣어줄 예정

실험 결과

- 하이퍼파라미터 정보 (random search 를 해가면서 발견 – validation loss 기준)

Table 1: Information on dataset and optimal TFT configuration.

	Electricity	Traffic	Retail	Vol.
Dataset Details				
Target Type	$\mathbb{R}$	[0, 1]	$\mathbb{R}$	$\mathbb{R}$
Number of Entities	370	440	130k	41
Number of Samples	500k	500k	500k	$\sim$100k
Network Parameters				
k	168	168	90	252
τ_{max}	24	24	30	5
Dropout Rate	0.1	0.3	0.1	0.3
State Size	160	320	240	160
Number of Heads	4	4	4	1
Training Parameters				
Minibatch Size	64	128	128	64
Learning Rate	0.001	0.001	0.001	0.01
Max Gradient Norm	0.01	100	100	0.01

GRN 을 이용하여 효과적으로 연산 비용을 줄임.
(V100) train – 6시간, validate – 8분 소요

실험 결과

Table 2: P50 and P90 quantile losses on a range of real-world datasets. Percentages in brackets reflect the increase in quantile loss versus TFT (lower q-Risk better), with TFT outperforming competing methods across all experiments, improving on the next best alternative method (underlined) between 3% and 26%.

	ARIMA	ETS	TRMF	DeepAR	DSSM
Electricity	0.154 (*+180%*)	0.102 (*+85%*)	0.084 (*+53%*)	0.075 (*+36%*)	0.083 (*+51%*)
Traffic	0.223 (*+135%*)	0.236 (*+148%*)	0.186 (*+96%*)	0.161 (*+69%*)	0.167 (*+76%*)
	ConvTrans	**Seq2Seq**	**MQRNN**	**TFT**	
Electricity	0.059 (*+7%*)	0.067 (*+22%*)	0.077 (*+40%*)	**0.055***	
Traffic	0.122 (*+28%*)	0.105 (*+11%*)	0.117 (*+23%*)	**0.095***	

(a) P50 losses on simpler univariate datasets.

	ARIMA	ETS	TRMF	DeepAR	DSSM
Electricity	0.102 (*+278%*)	0.077 (*+185%*)	-	0.040 (*+48%*)	0.056 (*+107%*)
Traffic	0.137 (*+94%*)	0.148 (*+110%*)	-	0.099 (*+40%*)	0.113 (*+60%*)
	ConvTrans	**Seq2Seq**	**MQRNN**	**TFT**	
Electricity	0.034 (*+26%*)	0.036 (*+33%*)	0.036 (*+33%*)	**0.027***	
Traffic	0.081 (*+15%*)	0.075 (*+6%*)	0.082 (*+16%*)	**0.070***	

(b) P90 losses on simpler univariate datasets.

	DeepAR	CovTrans	Seq2Seq	MQRNN	TFT
Vol.	0.050 (*+28%*)	0.047 (*+20%*)	0.042 (*+7%*)	0.042 (*+7%*)	**0.039***
Retail	0.574 (*+62%*)	0.429 (*+21%*)	0.411 (*+16%*)	0.379 (*+7%*)	**0.354***

(c) P50 losses on datasets with rich static or observed inputs.

	DeepAR	CovTrans	Seq2Seq	MQRNN	TFT
Vol.	0.024 (*+21%*)	0.024 (*+22%*)	0.021 (*+8%*)	0.021 (*+9%*)	**0.020***
Retail	0.230 (*+56%*)	0.192 (*+30%*)	0.157 (*+7%*)	0.152 (*+3%*)	**0.147***

(d) P90 losses on datasets with rich static or observed inputs.

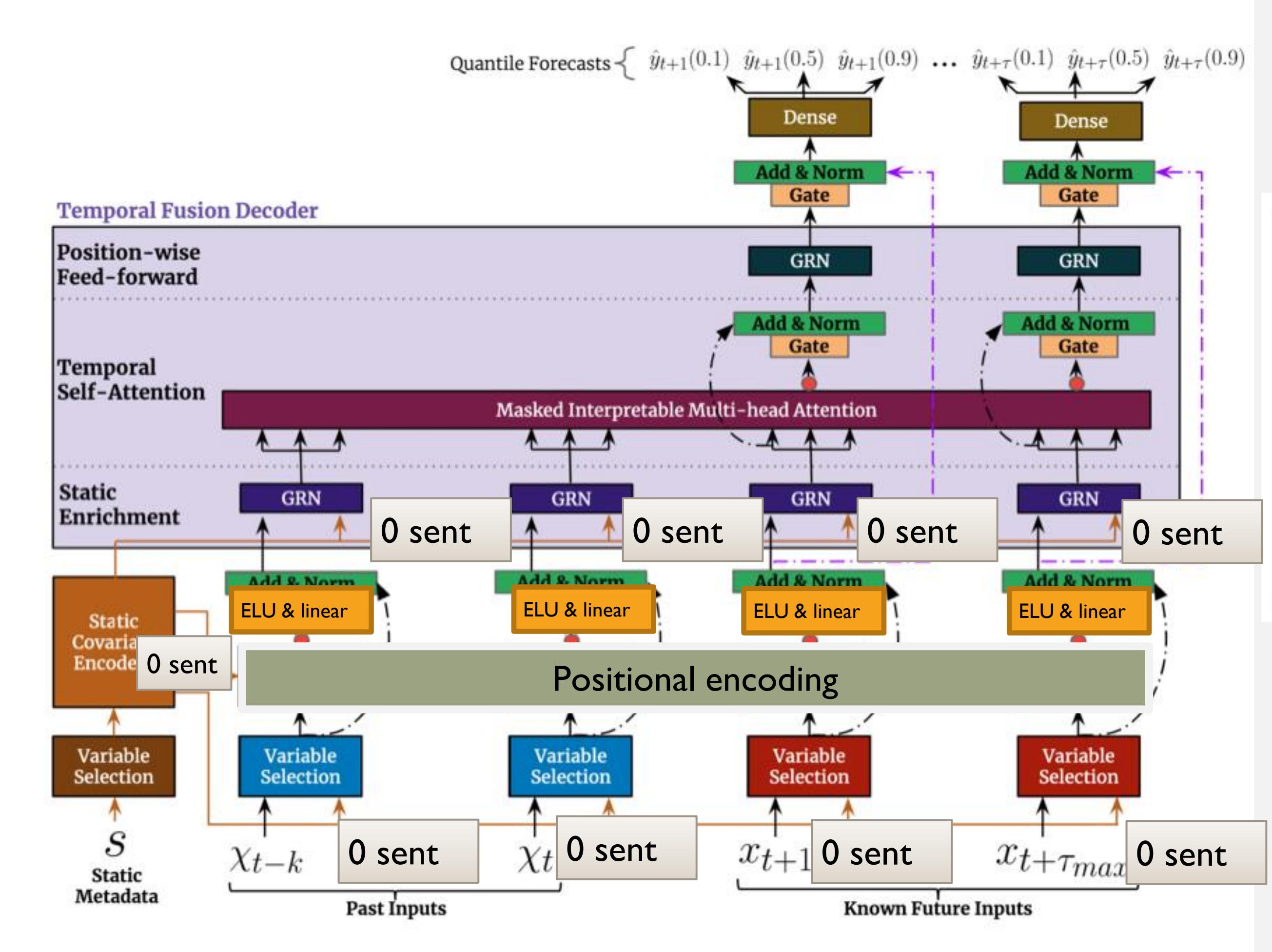
는 direct methods cf. iterative methods

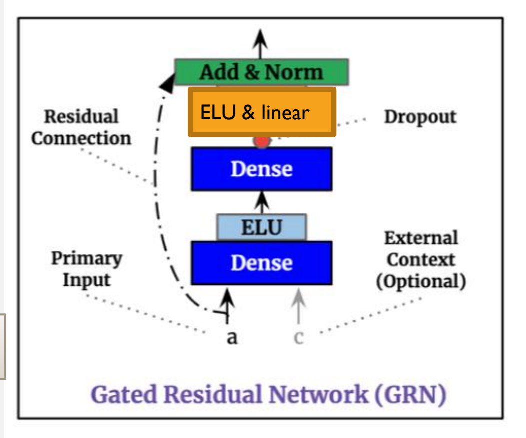
TFT 는 타 모델들에 비해 평균적으로 7% 낮은 P50 loss 와 9% 낮은 P90 loss 를 보여줌

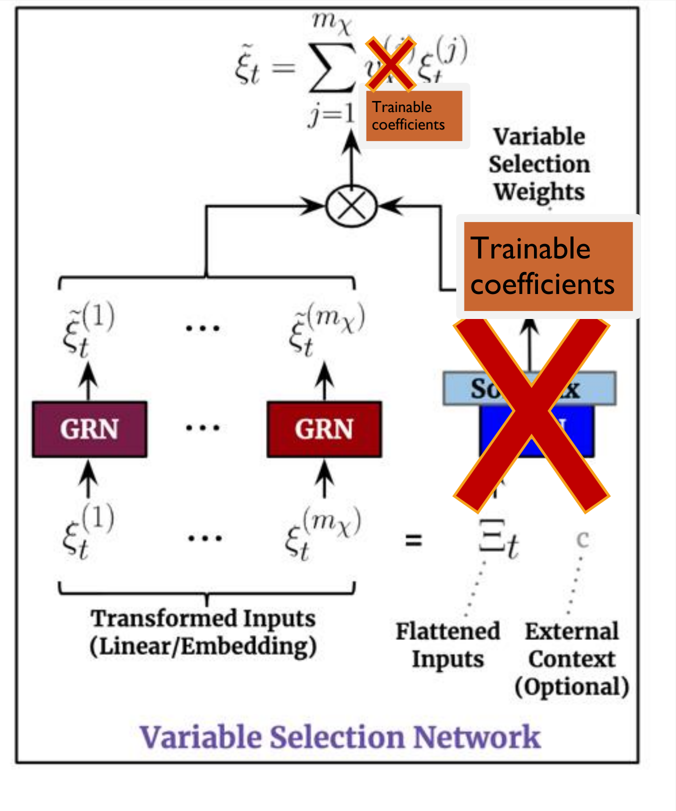
Iterative 방법을 사용한 핵심 모델 ConvTrans 의 경우 observed Input 등 다양하고 복잡한 데이터에서는 성능이 떨어짐

⇒ 즉 iterative methods 는 고정적인 input 값을 취해야한다는 한계를 넘지 못하였음을 보여줌

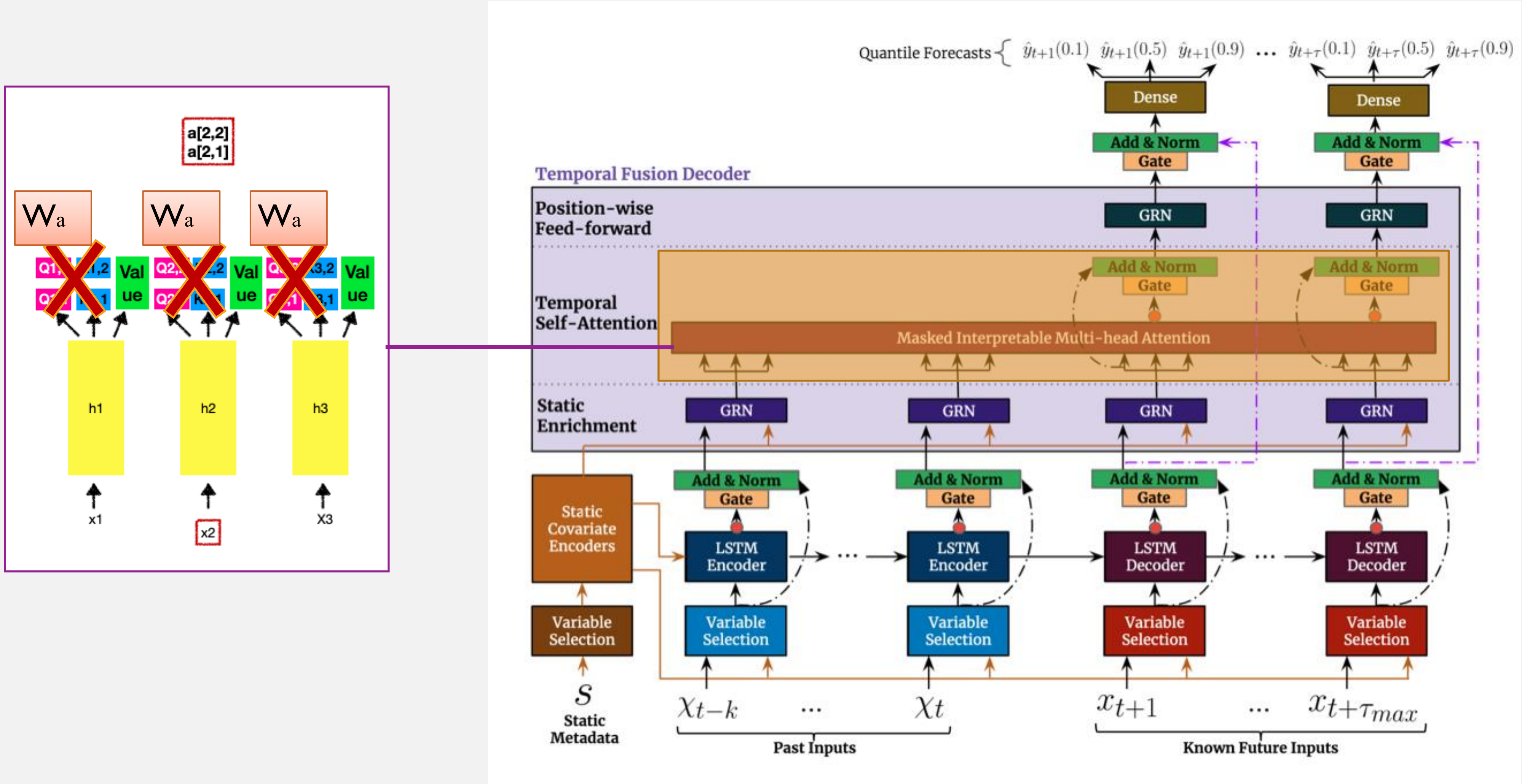
ABLATION ANALYSIS







ABLATION ANALYSIS



ABLATION RESULTS

- Capturing temporal relationships, local processing ☆☆ : 비활성화 시켰더니 P90 loss 평균 6% 증가

- Local processing : 비활성화 시켰더니 traffic,retail,volatility 는 모두 악영향, electricity 는 오히려 P50 loss 높게 나옴 : Electricity data 의 경우 daily 단위로 seasonality 가 발견되기 때문에 direct attention to previous days > adjacent time steps
- Static covariate encoder , variable selection : 비활성화 시켰더니 P90 loss 평균 4.1% 증가, electricity 에 제일 영향을 많이 미친 것으로 파악
- Gating layer : 비활성화 시켰더니 P90 loss 평균 1.9% 증가, 노이즈가 많은 volatility 에 제일 영향을 많이 미친 것으로 파악