

경북대학교 공학석사학위논문

# 시계열 데이터 극값 예측성능 향상을 위한 인코딩 모듈 연구

대학원 인공지능학부

김 민 기

2021년 12월

경북대학교 대학원

# 시계열 데이터 극값 예측성능 향상을 위한 인코딩 모듈 연구

이 논문을 공학석사 학위논문으로 제출함

대학원 인공지능학부

김민기

지도교수 김재일

김민기의 공학석사 학위논문을 인준함

2021년 12월

위원장: \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

경북대학교 대학원위원회

# 목차

I	서론	3
II	관련연구	7
2.1	신경망 기반 해수면온도 예측모델 . . . . .	7
III	배경	9
3.1	시계열 데이터의 특징 . . . . .	9
3.2	수용 장 블록(Receptive Field Block) . . . . .	10
3.3	순환 신경망(Recurrent Neural Network) . . . . .	11
3.4	장단기 메모리 유닛(Long Short-Term Memory Unit) . . . . .	12
3.5	게이트 순환 유닛(Gated Recurrent Unit) . . . . .	14
3.6	트랜스포머(Transformer) . . . . .	15
IV	제안모델	18
4.1	모델 구조 . . . . .	18
V	실험	20
5.1	데이터 셋 . . . . .	20
5.1.1	거문도 지역 부표관측 데이터 셋 . . . . .	20
5.2	훈련 시퀀스와 정답 시퀀스 구성 방법 . . . . .	20
5.3	평가 지표(Evaluation Metric) . . . . .	21
5.3.1	평균 제곱 오차(Mean Squared Error) . . . . .	21
5.3.2	평균 절대 오차(Mean Absolute Error) . . . . .	22
5.3.3	$R^2$ . . . . .	22
5.3.4	lead time에 따른 예측 성능 . . . . .	23
VI	실험 결과	25

6.1 양적 지표를 활용한 결과 분석 . . . . .	25
6.2 질적 지표를 활용한 결과 분석 . . . . .	26
<b>VII 결론</b>	<b>34</b>
<b>참고 문헌</b>	<b>35</b>

## 표 목록

표 1 lead time에 따른 모델 별 평균 제곱 오차, 평균 절대 오차, $R^2$ 의 결과 . . . . .	26
--------------------------------------------------------------------	----

그림 목록

그림 1 [31]에서 발췌해온 수용 장 블록의 그림이다. 수용 장 블록 에서는 크기가 다른 커널(kernel)을 가지는 컨볼루션 층과 크기가 다른 dilated convolution을 사용한다.	10
그림 2 [36]에서 발췌해온 장단기 메모리 유닛의 그림이다. $i$ , $f$ , $o$ 는 각각 입력게이트, 망각게이트, 아웃풋게이트이다. $c$ , $\tilde{c}$ 는 각각 메 모리 셀과 새로운 메모리 내용을 의미한다.	12
그림 3 [36]에서 발췌해온 게이트 순환 유닛의 그림이다. $r$ , $z$ 는 각 각 리셋 게이트와 업데이트 게이트를 의미하고 $h$ , $\tilde{h}$ 는 각각 은닉 상태와 후보 은닉 상태를 의미한다.	14
그림 4 [20]에서 제안된 트랜스포머 구조를 수정하여 인코더 부분만 을 추출한 후 완전연결층을 연결하여 실험에 사용하였다.	17
그림 5 본 연구에서 제안하는 수용 장 블록 기반 인코딩 모듈	18
그림 6 모델훈련을 위한 입력과 정답의 이동 창(sliding window)	21
그림 7 트랜스포머 모델의 lead time 1의 예측 성능 그래프	23
그림 8 트랜스포머 모델의 lead time 7의 예측 성능 그래프	24
그림 9 월, 일에 대한 정보를 One-Hot encoding하여 출력값과 합친 경우의 흐름도	25
그림 10 본 연구에서 제안하는 LPRFBM을 사용한 경우의 흐름도	25
그림 11 lead time에 따른 LSTM모델의 평균 제곱 오차	26
그림 12 lead time에 따른 LSTM모델의 평균 절대 오차	27
그림 13 lead time에 따른 LSTM모델의 $R^2$	27
그림 14 lead time에 따른 LSTM모델의 평균 제곱 오차	28
그림 15 lead time에 따른 LSTM모델의 평균 절대 오차	28
그림 16 lead time에 따른 LSTM모델의 $R^2$	29

그림 17 lead time에 따른 LSTM모델의 평균 제곱 오차 . . . . .	29
그림 18 lead time에 따른 LSTM모델의 평균 절대 오차 . . . . .	30
그림 19 lead time에 따른 LSTM모델의 $R^2$ . . . . .	30
그림 20 LSTM(Vanila)모델의 lead time 1일 때의 실험결과 . . . . .	31
그림 21 LSTM(RFB)모델의 lead time 1일 때의 실험결과 . . . . .	31
그림 22 GRU(Vanila)모델의 lead time 1일 때의 실험결과 . . . . .	32
그림 23 GRU(RFB)모델의 lead time 1일 때의 실험결과 . . . . .	32
그림 24 Transformer(Vanila)모델의 lead time 1일 때의 실험결과 . . . . .	33
그림 25 Transformer(RFB)모델의 lead time 1일 때의 실험결과 . . . . .	33

# 시계열 데이터 극값 예측성능 향상을 위한 인코딩 모듈 연구

Mingi Kim

경북대학교 대학원 인공지능학과  
(지도교수 김재일)

## Abstract

해수면 온도는 수치적 기후 예측모델과 해양 예측 모델의 가장 중요한 변수로 전세계 해양의 열 분배, 해양-대기 순환을 결정한다. 해수면 온도 변화는 전세계적인 기후 변화와 해양 시스템 간의 상호작용으로 인해 발생한다. 최근 다양한 외력요인들이 한반도의 해수면 온도를 높이고 있으며 이로 인해 연안 어로, 양식업 등 수산업에 막대한 피해를 끼치고 있다. 이러한 해수면 온도의 고수온 현상을 사전에 예측하여 대비한다면 고수온 현상으로 인해 발생되는 피해를 줄일 수 있을 것이다. 본 연구에서는 해수면 온도의 고수온과 같은 극값(Extreme Value)을 예측하는데 뛰어난 성능을 보이는 수용 장 블록(Receptive Field Block) 기반의 선형사영 인코딩 모듈 LPRFBM(Linear Projected Receptive Field Block Module)을 제안한다. LPRFBM은 시계열 데이터를 시계열 변수 축으로 선형사영시켜 시간적 요소 뿐 아니라 시계열 데이터의 변수까지도 고려할 수 있다. LPRFBM은 어느 시계열 예측모델의 인코딩에서도 적용할 수 있기 때문에 확장성이 높고 시간 정보를 부여하는 작업을 따로 해주지 않아도 되기 때문에 편리하게 사용할 수 있다. 여러 실험의 양적(Quantitative), 질적(Qualtitative)지표를 바탕으로 보았을 때, LPRFBM은 시계열 데이터의 극값 예측성능을 향상시킬

수 있다고 볼 수 있다.

## I 서론

해수면 온도(sea surface temperature, SST)는 해수면 온도를 예측하는 수치적 기후 예측모델(numerical weather prediction)과 해양 예측 모델(ocean forecasting models)의 가장 중요한 변수로써 전세계 해양의 열 분배, 해양-대기 순환 현상을 결정한다. 또한 해양의 상층부에서 장, 단기적인 해양순환을 일으키며 물리적인 영향 및 생지화학 변화에 대한 영향을 끼친다. 수치적 기후예측모델과 해양예측모델은 근해작업(offshore operations), 해상운송 (maritime transport), 해상안전(maritime safety), 긴급응답(emergency response), 군사작전(military operations), 근해동력개발과 관리(offshore power development and management), 생태계평가(ecosystem assesment), 해양오염조절(marine pollution control), 파도모델(wave and surf models), 어업지원과 관광(fisheries support and tourism) 등의 영역에서 점점 더 많이 사용되고 있다[1].

해수면 온도 변화는 전세계적인 기후 변화와 해양 시스템 간의 상호작용으로 인해 발생한다. 특히 북서태평양 해역은 태평양으로부터 고위도로 북상하는 쿠로시오 난류의 영향을 크게 받으며 엘니뇨, 라니냐 등의 장기간 다양한 규모의 해양현상이 난류의 강도 및 흐름을 변화시킨다. 국립수산과학원이 정한 우리나라에서의 이상 고수온은 절대적 수치  $28^{\circ}\text{C}$ 를 기준으로 정의한다. 해수면 온도 예측 시 수온  $28^{\circ}\text{C}$ 에 도달이 예상되는 해역에는 고수온 주의보가 발령되며,  $28^{\circ}\text{C}$ 이상의 수온이 3일 이상 지속되는 해역에는 고수온 경보가 발령된다. 또한 평년(최근 10년간) 대비  $2^{\circ}\text{C}$ 와  $3^{\circ}\text{C}$ 이상의 수온이 상승할 경우 각각 고수온 주의보와 경보를 발령한다. 한반도 해역에서의 대기 폭염 현상, 양자강 저염수의 황해로의 대량 유입, 대기의 고기압 전선 분포 등의 다양한 외력 요인이 한반도 해수면 온도 분포와 열류량에 영향을 미치며 대한민국 해역에 고수온 현상을

자주 발생시키고 있다. 한반도 남해 연안은 얕은 수심과 복잡한 연안구조를 가지고 있으며 연안 어로, 양식업 등 수산업이 활발하게 이루어지는 해역이다. 하지만 특정 기간동안 이상 고수온 현상이 발생하여 생태학적 변화를 일으켜 광범위한 피해를 끼친다. [2].

일반적인 데이터는 데이터들간의 관계를 동일독립분포(identical independent distribution)에서 추출되는 것으로 가정한다. 하지만 시계열데이터는 긴기간 떨어진 데이터에 대해서도 영향을 받기 때문에 데이터들간의 관계가 독립적이지 않다. 따라서 시계열 데이터는 다른 데이터들과 다르게 여겨졌으며 독립적인 연구분야를 형성하며 발전하였다. 시계열 예측(time series forecasting)은 유통에서의 상업적인 결정[3], 금융[4], 교통[5], 에너지[6] 등의 산업분야(industrial area)와 기후모델링(climate modelling)[7], 생물학[8], 약학[9] 등의 학문적인 분야(academical area)에서 많이 쓰이고 있다.

시계열을 예측하는 모델은 크게 네가지(전통적인 통계모델(classical statistics model), 머신러닝 모델(machine learning model), 신경망을 기반으로 하는 모델(neural net based model), 베이지안 모델(bayesian model))로 분류할 수 있으며 각각의 연구분야를 형성하며 발전하였다. 전통적인 통계모델 [10], [11], [12], [13]은 확률변수에 대한 정규성과 시계열의 정상성 가정에 기반하여 자기상관관계, 주기성, 계절성 등의 시간적 요소를 학습한다. 전통적인 통계모델은 확률변수에 대한 가정을 바탕으로 모델링 하기 때문에 모델을 구성하고 있는 모수에 대해 예측구간을 생성하는 등의 통계적인 해석이 가능하다. 머신러닝 모델은 시계열데이터가 아닌 데이터를 모델링하는데 자주 사용되는 트리기반모델[14], 서포트벡터머신[15], 비모수모델[16], [17]등이 해당되며 전통적인 통계모델과 비교해서 높은 시간, 공간 복잡도를 가진다. 신경망을 기반으로 하는 모델은 순환신경망(recurrent neural nets)을 기반으로

하는 모델[18],[19]과 어텐션을 사용하는 트랜스포머 기반모델[20]로 나눌 수 있으며, 시간적인 특징을 효과적으로 학습할 수 있기 때문에 최근 가장 많이 사용되는 모델 중 하나이다. 전통적인 통계모델은 확률변수에 대한 확률분포를 가정하기 때문에 설명력이 높은 장점이 있지만, 다른모델들 보다 상대적으로 예측성능이 저조하다. 그리고 신경망을 기반으로 하는 모델은 전통적인 통계모델보다 예측성능이 뛰어나지만, 복잡한 비선형 결정경계로 인해 설명력이 부족하다. 베이지안 모델[21], [22], [23], [24]은 전통적인 통계모델의 장점과 신경망 기반 모델의 장점을 취한 모델로 예측성능이 뛰어나면서 설명력이 높은 장점이 있다.

시계열 예측 모델을 사용하는 산업분야와 학문분야에서 극값(extreme value or high variance segments)이 일어날 상황을 잘 예측한다면 예상치 못한 상황에 대한 피해를 줄이고, 그 상황에 대해 적절히 대처할 수 있다. 따라서 극값이 일어날 상황에 대해 예측하는 연구가 활발히 진행되어 왔다. 극값에 대한 연구는 통계학의 응용분야 중 하나로 극값이론(Extreme Value Theory)이라는 이름으로 연구되고 있다. 신경망 기반 예측모델과 관련된 극값 발생 상황예측에 대한 연구는 부트스트랩(boot strap)기법을 사용하여 불확실성(uncertainty)을 측정한 연구[5], [25]가 주를 이룬다.

해수면 온도는 시간적 요소를 띠며 나타나기 때문에 시계열 예측 모델을 활용하여 해수면 온도를 예측하려는 연구 [26], [27], [2], [28], [29], [30]가 계속되어 왔다. 하지만 대부분의 연구들은 고수온 현상을 예측하는 것을 연구대상에 고려하지 않았다. 그리고 해수면 온도의 고수온 현상을 예측하는 것을 연구의 주된 목적으로 하는 연구는 드물었다. 따라서 본 연구에서는 해수면 온도의 고수온 현상을 예측하기 위해 [31]에서 제안된 Receptive Field Block을 활용한 인코딩 모듈을 제안한다. 본 연구에서 제안한 인코딩 모듈은 시계열 데이터를 인코딩 하는 효과적인 방법으로 어느 시계열 예측모델에도 포함시킬 수 있어

확장성이 뛰어나다. 또한 해수면 온도의 고수온 현상과 같은 극값을 효과적으로 예측할 수 있기 때문에 극값 예측을 위한 다른 연구에서도 사용될 수 있을 것으로 생각한다.

본 연구는 다음의 순서로 진행된다. 2장에서는 해수면 온도를 예측하는 신경망 기반의 모델에 대한 관련연구를 소개한다. 3장에서는 비교실험에 쓰일 순환신경망 기반 신경망 모델과 트랜스포머 기반 신경망 모델에 대해 간략하게 소개한다. 그리고 본 연구에서 제안하는 인코딩 모듈의 바탕이 되는 Receptive Field Block에 대한 이론적 배경에 대해 소개한다. 4장에서는 실험에 쓰일 데이터셋들과 모델들 그리고 평가방법에 대해 소개한다. 5장에서는 해수면 온도의 고수온 현상을 예측하기 위해 진행한 실험의 실험결과를 소개한다. 6장에서는 본 연구의 실험결과와 요지를 정리하며 끝맺는다.

## II 관련연구

이번 장은 기계학습과 신경망 기반의 해수면 온도 예측모델 소개하며 본 연구에 대한 동기를 간단히 설명한다.

### 2.1 신경망 기반 해수면온도 예측모델

신경망(Neural Nets)은 이미지 분류, 자연어 처리, 강화학습 분야에서 뛰어난 성능을 기록하며 많은 인기를 얻고 있다. 또한 데이터의 특징(feature)을 수작업(manual feature engineering)으로 만들지 않고도 데이터의 복잡한 표현(representation)을 학습할 수 있기 때문에 점점 더 많이 사용되고 있다[32]. 신경망이 많은 분야에서 뛰어난 성능을 기록하고 있기 때문에 해수면 온도를 예측하는 모델에도 점점 더 많이 사용되고 있다. 해수면 온도를 예측하는 모델은 인공위성에서 얻어진 해수면 온도 지도를 입력 데이터로 하는 연구과 부표(buoy)에서 얻어진 변수들을 입력 데이터로 하는 연구로 나눌 수 있다.

먼저 인공위성에서 얻어진 해수면 온도 지도를 입력데이터로 하는 연구는 다음과 같다. [26]에서는 신경망 기반의 시계열 예측모델을 만들 때 널리 사용되는 LSTM(Long Short Term Memory)모델에 완전연결층(Fully Connected Layer) 결합하여 해수면의 온도를 예측하는 모델로 사용하였다. [27]에서는 LSTM모델과 완전연결층을 통과시킨 후 컨볼루션층(Convolution layer)을 통과시켜 시간적(temporal)인 정보과 공간적(spatial)인 정보를 모두 학습하고자 하였다. [2]에서는 LSTM모델과 ConvLSTM(Convolutional Long Short Term Memory)모델을 사용하여 해수면의 온도를 예측한 성능을 제시하였다. 또한 고수온 예측에 대한 결과를 실제 인공위성 지도와 비교하여 시각적으로 모델의 결과를 비교해 볼 수 있도록 하였다. 여기서 ConvLSTM모델은 LSTM층 내부에 컨볼루션 층을

삽입하여 시간적인 정보와 공간적인 정보를 모두 학습할 수 있는 모델이다.

다음으로 부표에서 얻어진 변수들을 입력데이터로 하는 연구는 다음과 같다. [28]에서는 장기간 의존(long term dependancy)을 효과적으로 학습할 수 있지만 과적합(over fitting)에 취약한 LSTM모델에서 도출된 예측결과와 강력한 예측 성능(prediction capability)을 기록하면서 과적합에 강건한 Adaboost(adaptive boosting)모델에서 도출된 예측결과를 결합하는 모델을 제안하였다. [29]에서는 LSTM모델보다 더 적은 연산량을 통해 더 빠른 시간내에 학습할 수 있는 GRU(Gated Recurrent Units)모델을 여러 층 쌓는 모델을 제안하였다. [30]에서는 해수면 온도를 예측하는 대부분의 모델들은 다범위 특징(multiscale features)에 대한 영향을 무시하고 있다는 점을 지적하며 다범위 특징을 효과적으로 학습할 수 있는 M-LCNN(Multi Long short-term memory Convolution Neural Network)모델을 제안하였다. M-LCNN 모델에서는 입력 해수면 온도 데이터를 웨이블릿 변환(wavelet transform)을 사용하여 다른 시간범위(timescales)로 분해한다. 그렇게 변환된 데이터를 LSTM층과 컨볼루션층을 통과시켜 해수면 온도를 예측하고자 하였다.

이상의 관련연구에서 볼 수 있듯이 해수면 온도를 예측하는 연구들은 기존의 신경망 기반 예측모델을 해수면 온도를 예측하는 임무(task)에 사용해본 것에 의미가 있는 연구들이었고, 고수온 예측을 연구의 주된 목적으로 하지 않는 경우가 대부분이었다. 해수면 온도의 고수온 현상을 예측하는 연구가 부족하다는 사실은 본 연구를 진행하는 직접적인 동기가 되었다.

### III 배경

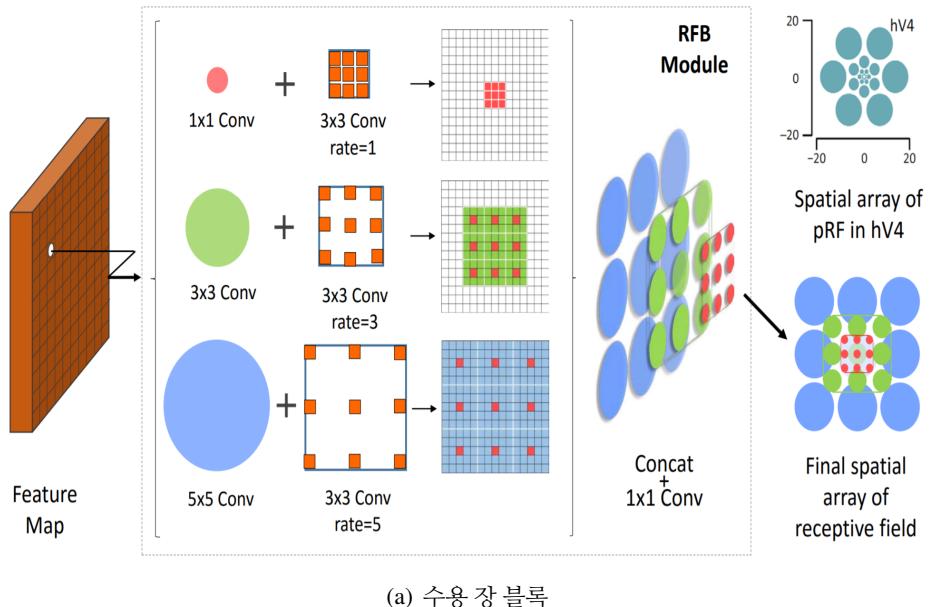
이번 장은 시계열 데이터의 특징과 본 연구에서 제안한 모델의 기반이 되는 수용 장 블록(Receptive Field Block)에 대한 배경 그리고 비교실험에 사용될 장단기 메모리 유닛(LSTM Unit), 게이트 순환 유닛(GRU Unit), 트랜스포머(Transformer)에 대한 배경을 설명한다.

#### 3.1 시계열 데이터의 특징

시계열 데이터는 시스템적인(systematic) 부분과 비시스템적인(non-systematic) 부분으로 구성된다. 시스템적인 부분은 예측모델에 기술될 수 있는 일관성과 반복성을 가진다. 그리고 비시스템적인 부분은 임의의 현상(random phenomenon)으로 예측모델에 의해 기술될 수 없는 특징을 가진다.

시스템적인 부분은 평균(level), 추세(trend), 계절성(seasonality)으로 나눌 수 있다. 평균은 시계열 데이터의 평균적인 패턴으로 생각할 수 있고, 추세는 시계열 데이터의 요소들이 조화된 패턴으로 생각할 수 있다. 그리고 계절성은 짧은 기간의 순환적인 변화가 긴기간 나타나는 패턴으로 생각할 수 있다. 이 시스템적인 부분은 분해기법[33]으로 추세와 계절성요소로 분해 할 수 있으며 계절성 요소가 가지는 효과를 없애면 추세를 더욱 명확히 파악할 수 있다. 비시스템적인 부분은 예측모델에 의해 기술될 수 없는 임의의 현상으로 확률변수에 의해 정규분포와 같은 분포로 모델링 된다.

시계열 데이터는 앞서 기술한 평균, 추세, 계절성으로 나눌 수 있는 시스템적인 부분과 임의의 현상으로 표현되는 비시스템적인 부분들이 가법적(additive)이거나 승법적(multiplicative)인 형태로 나타나게 된다. 시계열 데이터를 분석하고 예측모델을 만드는 주된 목적은 이 요소들로부터 다양한



(a) 수용 장 블록

**그림 1:** [31]에서 발췌해온 수용 장 블록의 그림이다. 수용 장 블록에서는 크기가 다른 커널(kernel)을 가지는 컨볼루션 층과 크기가 다른 dilated convolution을 사용한다.

패턴을 학습하는 것에 있다.

### 3.2 수용 장 블록(Receptive Field Block)

수용 장 블록(Receptive Field Block)은 [31]에서 제안되었다. 수용 장 블록은 인간의 시각 시스템이 다양한 크기의 이심률을 가지는 다양한 크기의 수용 장들로 구성된다는 신경과학 사실에 기반하여 착안되었다. 구체적으로, 수용 장 블록에서는 인간의 시각 시스템이 가지는 이심률을 재현하기 위해 크기가 다른 dilated convolution를 사용하였고, 다양한 크기의 수용장을 구현하기 위해 컨볼루션 층에 크기가 다른 커널(kernel)들을 사용하였다. 수용 장 블록을 시각적으로 나타낸 그림은 다음과 같다.

### 3.3 순환 신경망(Recurrent Neural Network)

순환 신경망(Recurrent Neural Network)는 전통적으로 쓰이던 전방향 신경망(feedforward neural network)에 대한 확장으로 길이가 변하는 입력 시퀀스를 처리할 수 있다. 순환 신경망은 길이가 변하는 입력 시퀀스를 각 시간마다 활성화되는 순환 은닉 상태로 모델링 하게 된다. 시퀀스  $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_T)$ 가 주어졌을 때, 순환 신경망은 순환 은닉 상태(recurrent hidden state)를 다음과 같이 갱신한다.

$$\mathbf{h}_t = \begin{cases} 0 & t = 0 \\ \phi(\mathbf{h}_{t-1}, \mathbf{x}_t) & \text{otherwise,} \end{cases}$$

여기서  $\phi$ 는 비선형 함수를 말한다. 그리고 순환 신경망의 갱신은 다음과 같은 식으로 구현된다.

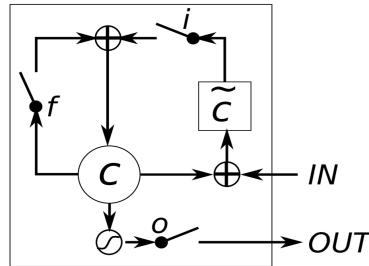
$$\mathbf{h}_t = g(W\mathbf{x}_t + U\mathbf{h}_{t-1}),$$

여기서  $g$ 는 logistic sigmoid, hyperbolic tangent과 같은 부드럽고 유계(bounded)인 함수를 말한다. 이전의 관측치가 주어졌을 때, 현재 시간에 대한 확률 분포는 다음과 같이 나타낼 수 있다.

$$p(x_t | x_1, \dots, x_{t-1}) = g(h_t).$$

입력 시퀀스가 주어졌을 때 입력 시퀀스에 대한 결합 분포는 다음과 같이 나타낼 수 있다.

$$p(x_1, \dots, x_T) = g(h_1)g(h_2) \cdots g(h_T).$$



(a) 장단기 메모리 유닛

**그림 2:** [36]에서 발췌해온 장단기 메모리 유닛의 그림이다.  $i$ ,  $f$ ,  $o$ 는 각각 입력 게이트, 망각게이트, 아웃풋게이트이다.  $c$ ,  $\tilde{c}$ 는 각각 메모리 셀과 새로운 메모리 내용을 의미한다.

하지만 기울기 소실 또는 증폭(gradient vanish or explode) 때문에 장기 종속성 (long-term dependencies)을 순환 신경망에서 학습시키기 어렵다는 사실이 발견되었다[34]. 기울기 소실 또는 증폭은 기울기 크기의 변화(variations in gradient magnitudes)와 단기 종속성을 발생시켜 장기 종속성의 효과를 숨기기 때문에 기울기 기반 최적화를 어렵게 만든다.

기울기 소실 또는 증폭의 영향을 완화시키기 위해 두 가지 접근 방법이 제시되었다. 첫 번째는 확률적 경사 하강법과 같은 알고리즘보다 더 나은 학습 알고리즘을 사용하는 접근 방법이고, 두 번째는 원소별 곱셈에 의해 수행되는 아핀 변환과 같은 더 복잡한 활성화 함수를 사용하는 방법이다. 두번째 방법을 시도한 연구가 후술할 장단기 메모리 유닛[18]과 게이트 순환 유닛[35]에 해당한다.

### 3.4 장단기 메모리 유닛(Long Short-Term Memory Unit)

장단기 메모리는 [18]에서 제안되었다. 입력 신호를 가중합 한 후 비선형 함수를 적용하는 기존의 순환 유닛과 다르게 각  $j$ 번째 장단기 메모리 유닛은

시간  $t$ 에 대한 메모리  $c_t^j$ 를 유지한다. 출력 은닉 상태  $h_t^j$ 는 다음과 같이 나타낼 수 있다.

$$h_t^j = o_t^j \tanh(c_t^j),$$

여기서  $o_t^j$ 는 메모리 내용의 출력양을 모듈화 하는 아웃풋 게이트(output gate)이다. 아웃풋 게이트는 다음과 같이 계산된다.

$$o_t^j = \sigma(W_o \mathbf{x}_t + U_o \mathbf{h}_{t-1} + V_o \mathbf{c}_t),$$

여기서  $\sigma$ 는 sigmoid 함수이고,  $V_o$ 는 대각행렬이다. 메모리 셀(memory cell)  $c_t^j$ 는 기존의 메모리를 부분적으로 망각하고 새로운 메모리 셀 내용(new memory cell content)을 더하며 다음과 같이 계산된다.

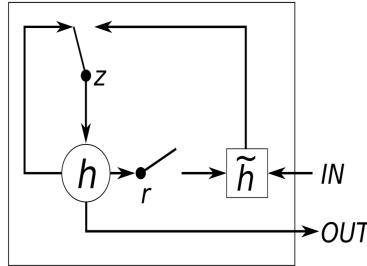
$$c_t^j = f_i^j c_{t-1}^j + i_t^j \tilde{c}_t^j,$$

여기서 새로운 메모리 셀 내용은 다음과 같다.

$$\tilde{c}_t^j = \tanh(W_c \mathbf{x}_t + U_c \mathbf{h}_{t-1})^j.$$

메모리 셀에서 부분적으로 망각하는 정도를 결정하는  $f_t^j$ 와 새로운 메모리 셀 내용을 얼마나 더할지의 정도를 결정하는  $i_t^j$ 는 다음과 같이 계산된다.

$$\begin{aligned} f_t^j &= \sigma(W_f \mathbf{x}_t + U_f \mathbf{h}_{t-1} + V_f \mathbf{c}_{t-1})^j \\ i_t^j &= \sigma(W_i \mathbf{x}_t + U_i \mathbf{h}_{t-1} + V_i \mathbf{c}_{t-1})^j, \end{aligned}$$



(a) 게이트 순환 유닛

**그림 3:** [36]에서 발췌해온 게이트 순환 유닛의 그림이다.  $r$ ,  $z$ 는 각각 리셋 게이트와 업데이트 게이트를 의미하고  $h$ ,  $\tilde{h}$ 는 각각 은닉 상태와 후보 은닉 상태를 의미한다.

여기서  $V_f$ 와  $V_i$ 는 대각행렬이다.

전통적인 순환 유닛과 다르게 장단기 메모리는 여러 게이트들을 통해 존재하는 메모리의 어떤 부분을 망각하고 어떤 부분을 남길지를 학습하게 된다. 결론적으로 이전 시간의 중요한 정보들만 남길 수 있게 되어 장기 종속성을 효과적으로 학습할 수 있게 된다.

### 3.5 게이트 순환 유닛(Gated Recurrent Unit)

게이트 순환 유닛은 [19]에서 제안되었다. 게이트 순환 유닛은 다른 시간 범위에 대한 종속성을 적응적으로 학습할 수 있도록 만든 순환 유닛이다. 입력 신호를 게이팅 유닛을 통과시켜 정보의 흐름을 조절하는 것은 장단기 메모리와 같지만, 분리된 메모리 셀이 없는 것이 특징이다. 게이트 순환 유닛의 시간  $t$ 에 대한 은닉 상태  $h_t^j$ 는 이전 은닉 상태  $h_{t-1}^j$ 와 후보 은닉 상태  $\tilde{h}_t^j$  사이의 선형 보간을 통해 계산되는데 계산식은 다음과 같다.

$$h_t^j = (1 - z_i^t)h_{t-1}^j + z_t^j \tilde{h}_t^j,$$

여기서  $z_t^j$ 는 업데이트 게이트(update gate)로 은닉상태들의 망각정도를 결정한다.  $z_t^j$ 는 다음과 같이 계산된다.

$$z_t^j = \sigma(W_z \mathbf{x}_t + U_z \mathbf{h}_{t-1})^j.$$

게이트 순환 유닛에서는 이전 은닉 상태  $h_{t-1}^j$ 와 후보 은닉 상태  $h_t^{\sim j}$  사이의 선형 보간을 통해 계산되는 은닉 상태를 얼마나 내보낼 것인지를 결정하는 유닛이 없다는 점에서 장단기 메모리 유닛과 차이점이 있다. 후보 은닉 상태는 다음식을 통해 계산된다.

$$\tilde{h}_t^j = \tanh(W \mathbf{x}_t + U(\mathbf{r}_t \odot \mathbf{h}_{t-1}))^j,$$

여기서  $r_t^j$ 는 리셋 게이트(reset gate)로 리셋 게이트의 값이 없으면 이전 상태에 대한 정보를 전달하지 않겠다는 것을 의미하며 다음과 같이 계산된다.

$$r_t^j = \sigma(W_r \mathbf{x}_t + U_r \mathbf{h}_{t-1})^j.$$

### 3.6 트랜스포머(Transformer)

트랜스포머는 [20]에서 제안되었다. 신경망을 기반으로 하는 자연어 처리 연구에서는 순환 신경망을 이용한 시퀀스 투 시퀀스 (seq2seq)모델이 많이 사용되었다[37], [19]. 순환 신경망을 이용한 시퀀스 투 시퀀스 모델은 인코더(encoder)-디코더(decoder) 형태로 구성되어있다. 인코더는 입력 시퀀스를 정해진 길이의 벡터 표현(fixed length vector representation)으로 압축하고, 디코더는 이 벡터 표현을 통해 출력시퀀스를 생성한다. 하지만 시퀀스 투 시퀀스 구조는 입력 시퀀스를 정해진 길이의 벡터 표현으로 압축할 때 정보가

손실된다는 단점이 있었다. 또한 이전 시간의 정보를 망각하는 장기 종속성 문제도 나타났다. 이러한 문제를 해결하기 위해 출력 시퀀스가 입력 시퀀스 중 어디에 더 연관이 있는지를 학습하는 모델이 제안되었다[38]. 이 방법[38]은 입력 시퀀스를 압축할 때 생기는 정보의 손실이나 장기 종속성 문제를 일부 해결하였다. 하지만 순환 신경망의 순차적 계산(sequential computation)방식을 대체하지는 못했고 결과적으로 높은 연산량이 요구되었다. 트랜스포머는 어텐션 메카니즘(attention mechanism)을 활용해 출력 시퀀스가 입력 시퀀스 중 어디에 더 연관이 있는지를 효과적으로 학습했을 뿐 아니라, 어텐션 구조만을 사용하여 기존 순환 신경망의 순차적인 계산을 완전히 대체할 수 있다는 것을 보여주었다.

트랜스포머에서는 다중얼굴 주의(Multi-Head Attention)가 사용되었는데, 다중얼굴 주의는 여러 개의 조정된 점곱 주의(Scaled Dot-Product Attention)로 구성된다. 조정된 점곱 주의는 쿼리(Q)와 키(K)간의 유사도를 점곱(Dot-Product)연산으로 구한 후 키에 해당 하는 밸류(V)를 곱해서 계산한다. 다중얼굴 주의는 계산된 조정된 점곱 주의의 정보를 취합하여 다양한 표현부공간(representation subspaces)으로부터 정보를 효과적으로 학습하게 된다. 다중얼굴 주의와 조정된 점곱 주의의 계산식은 다음과 같다.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

$$\text{MultiHeadAttention}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O$$

$$\text{,where } \text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

본 연구의 실험에 사용된 트랜스포머 구조는 [20]에서 제안된 모델을 시계열 데이터의 특성에 맞게 인코더 부분만을 추출한 후 완전연결층을 연결하여 사용하였다. 실험에 사용된 트랜스포머의 구조는 다음과 같다.

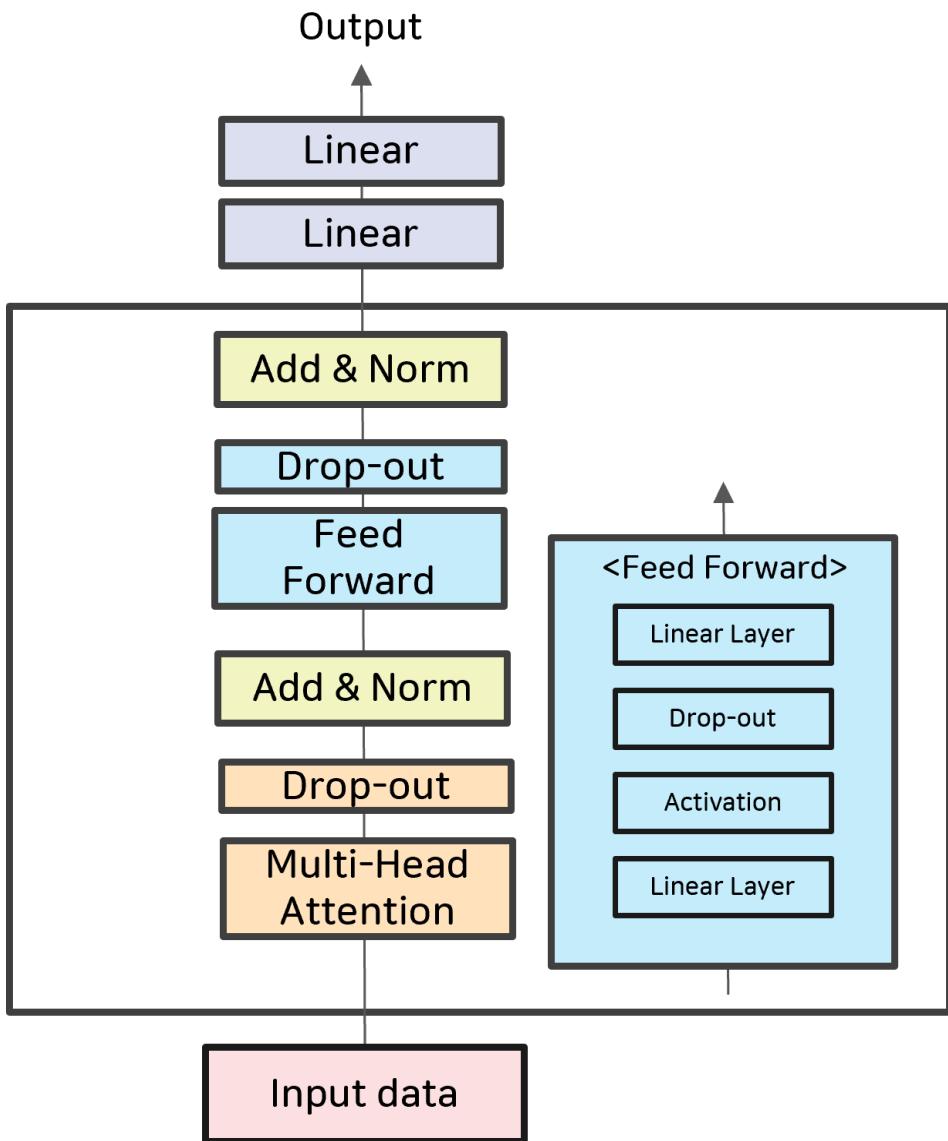


그림 4: [20]에서 제안된 트랜스포머 구조를 수정하여 인코더 부분만을 추출한 후 완전연결층을 연결하여 실험에 사용하였다.

## IV 제안모델

본 연구에서는 해수면의 고수온과 같은 극값을 예측하기 위해 고안된 수용장 블록 기반의 인코딩 모듈(LPRFBM)을 제안한다. 제안된 인코딩 모듈은 어느 시계열 예측 모델에도 삽입 될 수 있으므로 확장성이 뛰어나고, 시간 순서를 부여하는 작업을 따로 해주지 않아도 되기 때문에 편리하게 사용할 수 있다.

### 4.1 모델 구조

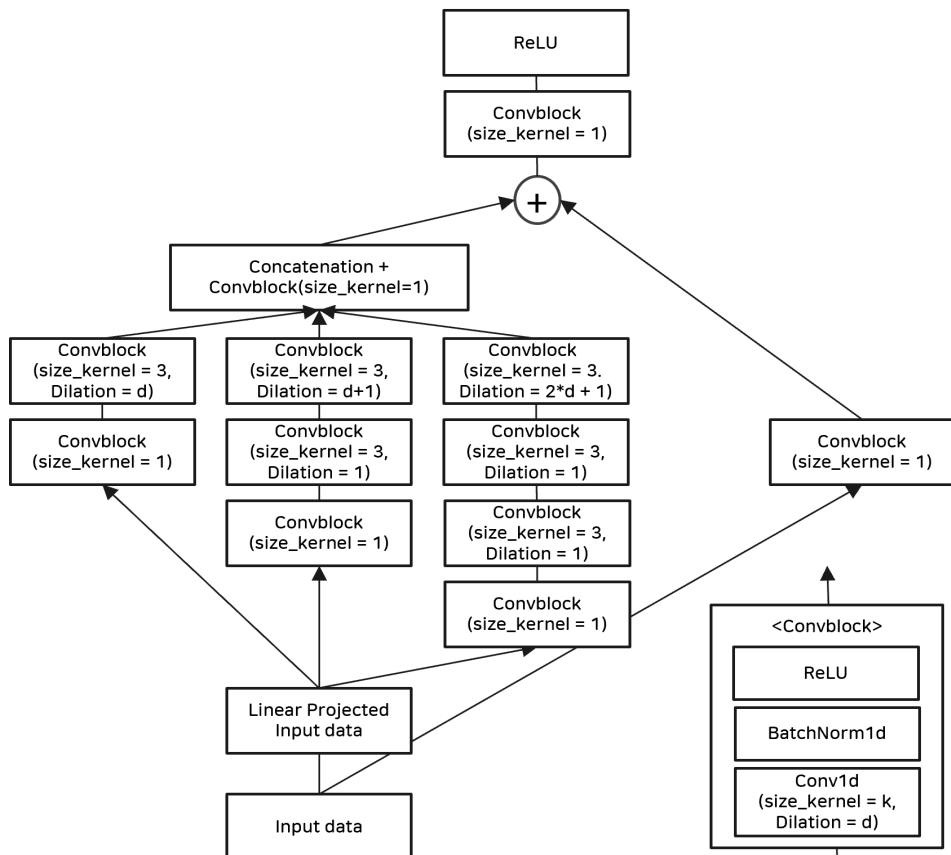


그림 5: 본 연구에서 제안하는 수용 장 블록 기반 인코딩 모듈

기존 신경망 기반의 시계열 예측모델의 경우 평균(level), 추세(trend), 계절성(seasonality)의 시간적 요소를 주로 고려하였다. 본 연구에서 제안 하는 인코딩 모듈은 입력 시계열 데이터를 선형사영시켜 시계열 데이터의 시간적 요소 뿐 아니라 시계열 데이터의 변수에 대한 특징도 고려할 수 있도록 하였다. 선형사영된 입력 시계열 데이터는 세 갈래로 나누어져 학습(그림 5 참고)된다. 커널사이즈(kernel size)가 1인 Convblock에서는 시간 채널에 대한 컨볼루션을 수행하여 중요하다고 여겨지는 시간에 더 큰 가중치를 주도록 학습된다. 커널 사이즈가 3인 Convblock에서는 시계열 데이터의 변수 채널에 대한 컨볼루션을 수행하여 중요하다고 여겨지는 변수에 더 큰 가중치를 주도록 학습된다. 또한 다양한 크기의 Dilated convolution을 수행함으로써 다양한 범위에서 특징을 학습할 수 있도록 하였다.

## V 실험

### 5.1 데이터셋

#### 5.1.1 거문도 지역 부표관측 데이터셋

한반도 남해 연안은 근래 고수온이 빈번하게 발생하여 많은 피해를 받고 있는 지역이다. 그 중 고수온 발생 빈도가 가장 높은 거문도 지역을 실험지역으로 선정하였다. 실험에 사용할 데이터는 1998년 1월 8일부터 2021년 6월 30일 동안 거문도 지역 부표에서 관측된 온도 데이터를 하루 단위로 평균한 일평균 데이터이다. 고수온 현상이 일어나는 횟수(훈련 기간 중 고수온 현상이 총 65회 발생하고 테스트 기간 중 고수온 현상이 총 36회 발생)를 고려하여 훈련에 사용할 데이터셋의 기간을 1998년 1월 8일부터 2015년 12월 31일 까지로 설정하였고, 테스트 데이터셋의 기간을 2016년 1월 1일부터 2021년 6월 30일로 설정하였다. 본 데이터셋은 한국과학기술원(KIEST)으로부터 제공받았다.

### 5.2 훈련 시퀀스와 정답 시퀀스 구성 방법

Training Sequence  $\mathbf{x} = (x_1, \dots, x_T)$ 를 모델의 입력으로 하여 Predicted Sequence  $\hat{\mathbf{y}} = (\hat{y}_1, \dots, \hat{y}_{\hat{T}})$ 를 출력한다. 그 이후 Predicted Sequence와 Answer Sequence  $\mathbf{y} = (y_1, \dots, y_{\hat{T}})$ 의 오차를 계산하여 모델을 학습하게 된다. 본 실험에서는 Training Sequence의 길이와 Answer Sequence의 길이를 각각 7일로 설정하였다.

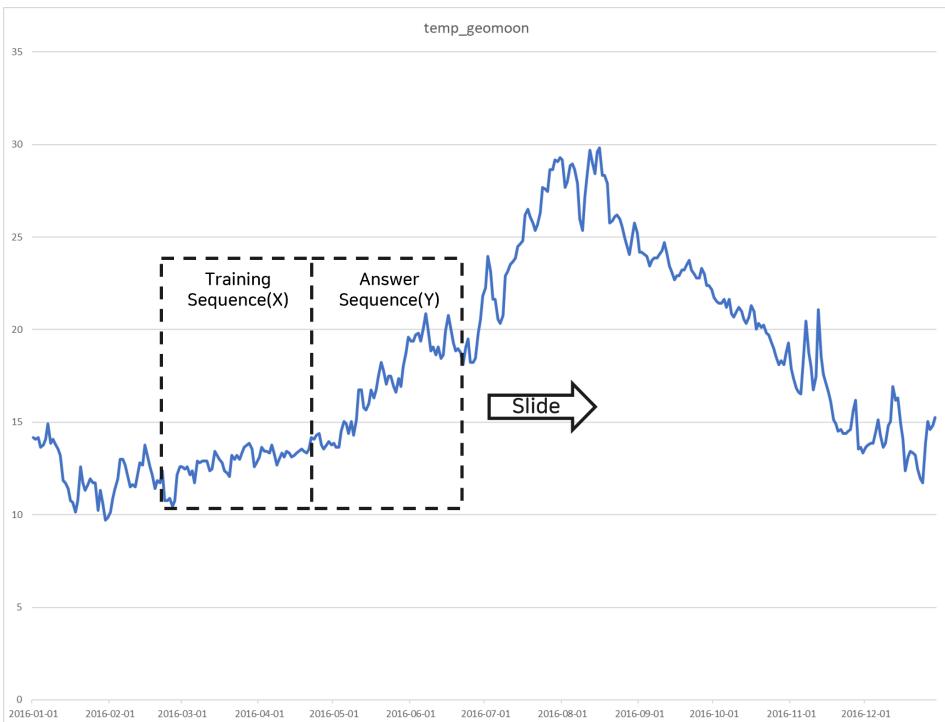


그림 6: 모델훈련을 위한 입력과 정답의 이동 창(sliding window)

### 5.3 평가 지표(Evaluation Metric)

예측모델의 성능을 양적(Quantitative)으로 평가하기 위한 지표로 평균 제곱 오차(Mean Squared Error), 평균 절대 오차(Mean Absolute Error),  $R^2$ 를 사용하였고, 질적(Qualitative)으로 평가하기 위한 지표로 lead time에 따른 예측 성능을 비교하였다.

#### 5.3.1 평균 제곱 오차(Mean Squared Error)

평균 제곱 오차(Mean Squared Error)는 예측값과 실제값의 차이를 제곱 계산한 후 이 값들을 평균하여 산출한다. 예측값이 실제값에서 많이 벗어날수록 평균 제곱 오차가 기하급수적으로 커지는 경향이 있어 실제값에서 크게 벗어

나는 예측값을 보정할 수 있다. 평균 제곱 오차의 값은 제곱을 해주기 때문에 예측값, 실제값과 스케일(scale)이 다를 수 있으며 이를 보완하기 위해 제곱근(root)을 취한 값(Root Mean Squared Error)을 쓰기도 한다. 평균 제곱 오차의 계산식은 다음과 같다.

$$MSE(\text{Mean Squared Error}) = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

### 5.3.2 평균 절대 오차(Mean Absolute Error)

평균 절대 오차(Mean Absolute Error)는 예측값과 실제값의 차이에 절대값을 취한 후 이 값들을 평균하여 산출한다. 예측값과 실제값의 차이에 절댓값을 취하기 때문에 예측값이 실제값에 대해 절대적으로 얼마나 떨어져 있는지를 비교할 수 있다. 또한 평균 절대 오차는 예측값과 실제값의 차이를 절대값을 취하기 때문에 스케일이 달라지지 않는 특징이 있다. 평균 절대 오차의 계산식은 다음과 같다.

$$MAE(\text{Mean Absolute Error}) = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

### 5.3.3 R<sup>2</sup>

R<sup>2</sup>는 실제값의 편차(실제값에서 실제값의 평균을 뺀 값의 제곱)들의 합과 잔차(실제값과 예측값의 차이의 제곱)들의 합의 비율을 1에서 뺀 값으로 정의 한다. 이는 예측모델이 출력한 예측값이 실제값에 얼마나 가깝게 분포하는지를 나타내는 척도로 사용되며 통계학에서 선형모델을 해석할 때 많이 사용된다. R<sup>2</sup>의 계산식은 다음과 같다.

$$R^2 = 1 - \frac{\sum_i^n (SST_i^{real} - SST_i^{predicted})^2}{\sum_i^n (SST_i^{real} - \bar{SST}^{real})^2}, \quad \bar{SST}^{real} = \frac{1}{n} \sum_i^n SST_i^{real}$$

### 5.3.4 lead time에 따른 예측 성능

lead time은 현재 시점으로부터 특정 미래 시점까지의 기간을 의미한다. lead time에 따른 예측 성능을 그래프로 그리게 되면 시계열 예측 모델이 미래를 얼마나 잘 예측하는지를 시각적으로 확인할 수 있다. 예를 들어 lead time 7의 예측 성능 그래프는 테스트 데이터셋(test data set)의 각 시점에서 7 단위시간 만큼 떨어진 기간의 예측을 나타내게 된다. lead time 1의 트랜스포머 모델 예측 성능 그래프와 lead time 7의 트랜스포머 모델 예측 성능 그래프는 다음과 같다.

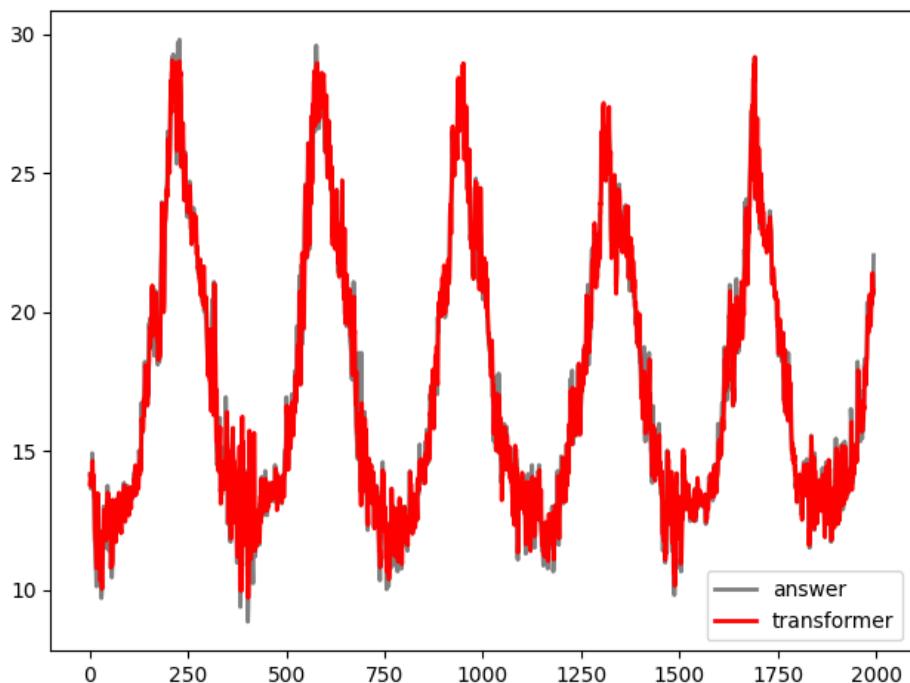


그림 7: 트랜스포머 모델의 lead time 1의 예측 성능 그래프

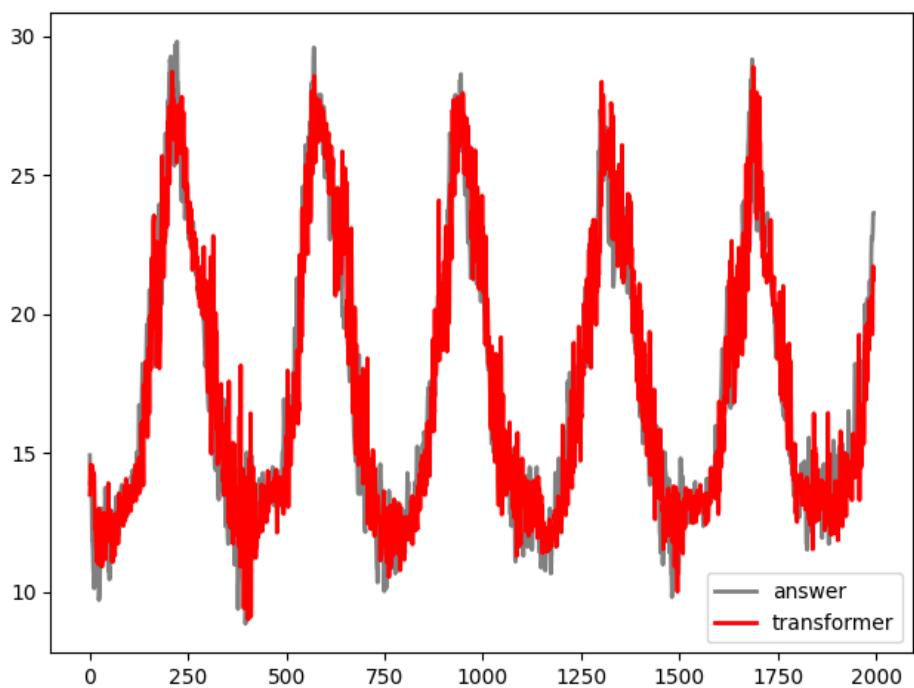


그림 8: 트랜스포머 모델의 lead time 7의 예측 성능 그래프

## VI 실험 결과

두가지 경우로 나누어서 모델 별로 실험을 진행하였다. 첫 번째 경우는 본 연구에서 제안하는 LPRFBM을 사용한 결과(Proposed)이고, 두 번째 경우는 월, 일에 대한 정보를 One-Hot encoding하여 모델의 출력값과 합친(Concatenation) 결과(Vanila)이다. 각 경우에 대한 흐름도는 다음과 같다.

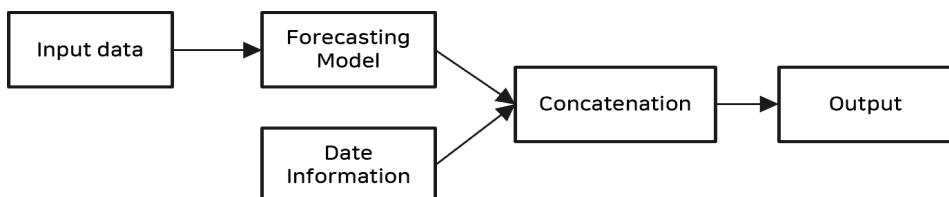


그림 9: 월, 일에 대한 정보를 One-Hot encoding하여 출력값과 합친 경우의 흐름도

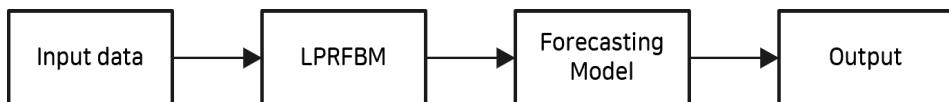


그림 10: 본 연구에서 제안하는 LPRFBM을 사용한 경우의 흐름도

### 6.1 양적 지표를 활용한 결과 분석

lead time<sup>o</sup>] 커짐에 따라 평균 제곱 오차, 평균 절대 오차의 값은 커지는 경향을 보였고,  $R^2$ 는 작아지는 경향을 보였다. lead time<sup>o</sup>] 커질수록 멀리 떨어진 기간을 예측하게 되어 성능이 하락한 것으로 생각된다. 그리고 단순히 날짜에 대한 정보를 제공하는 경우(Vanila)보다 본 연구에서 제안하는 LPRFBM을 사용 할 때 월등히 높은  $R^2$ 값과 월등히 낮은 평균 제곱 오차, 평균 절대 오차의 값을 기록하였다.

		Transformer(Vanila)	Transformer(Proposed)	GRU(Vanila)	GRU(Proposed)	LSTM(Vanila)	LSTM(Proposed)
lead time 1	MSE	2.371594	0.636887	1.586594	0.719411	1.563499	0.637823
	MAE	1.16662	0.589316	0.980138	0.625739	0.971322	0.594622
	R <sup>2</sup>	0.896186	0.972121	0.930549	0.968509	0.93156	0.97208
lead time 2	MSE	2.376074	1.46951	1.523813	1.376334	1.525432	1.275701
	MAE	1.167495	0.898459	0.960272	0.881297	0.959371	0.845523
	R <sup>2</sup>	0.89604	0.935705	0.933329	0.939781	0.933258	0.944184
lead time 3	MSE	2.380698	1.916413	1.477042	1.741156	1.468465	1.723169
	MAE	1.168418	1.04711	0.944598	1.001096	0.941275	1.003994
	R <sup>2</sup>	0.895886	0.91619	0.935405	0.923855	0.93578	0.924641
lead time 4	MSE	2.384757	2.24684	1.450441	2.060434	1.430619	2.048427
	MAE	1.169231	1.137648	0.936906	1.105206	0.929125	1.094544
	R <sup>2</sup>	0.895755	0.901784	0.936597	0.909932	0.937463	0.910457
lead time 5	MSE	2.390289	2.566113	1.436209	2.31539	1.408545	2.306201
	MAE	1.170427	1.212719	0.932872	1.171861	0.922418	1.172023
	R <sup>2</sup>	0.89556	0.887878	0.937247	0.898833	0.938456	0.899235
lead time 6	MSE	2.398074	2.745413	1.430188	2.522705	1.397451	2.643639
	MAE	1.17188	1.273445	0.929778	1.217293	0.917899	1.241984
	R <sup>2</sup>	0.895287	0.88012	0.93755	0.889845	0.938979	0.884564
lead time 7	MSE	2.40587	2.761429	1.428901	2.736272	1.392923	2.811209
	MAE	1.173236	1.277568	0.930275	1.269782	0.915555	1.287771
	R <sup>2</sup>	0.895018	0.879502	0.937648	0.8806	0.939218	0.87733

표 1: lead time에 따른 모델 별 평균 제곱 오차, 평균 절대 오차, R<sup>2</sup>의 결과

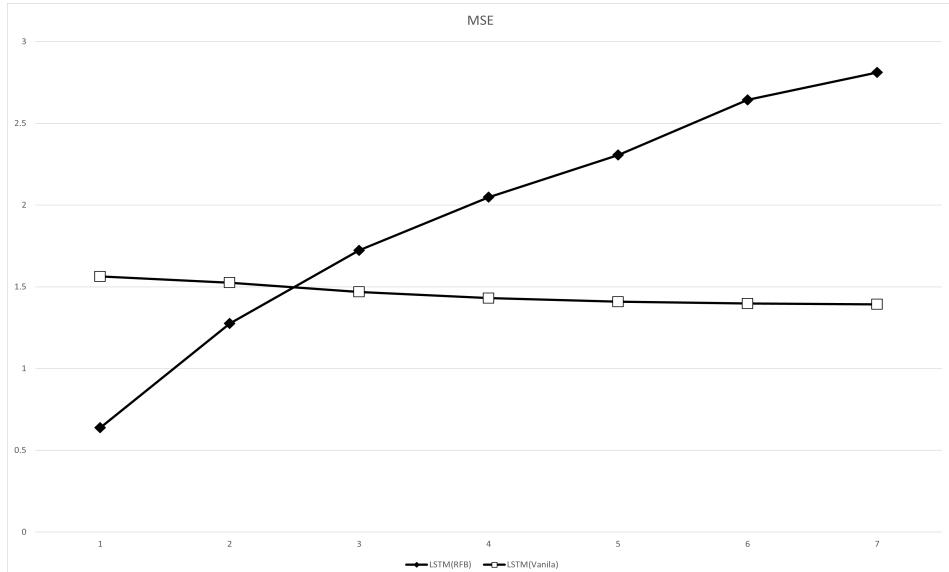


그림 11: lead time에 따른 LSTM모델의 평균 제곱 오차

## 6.2 질적 지표를 활용한 결과 분석

lead time 1일 때 각 모델의 예측 값을 나타내어보았을 때 단순히 날짜에 대한 정보를 제공하는 경우보다 본 연구에서 제안하는 LPRFBM을 사용할 때 극값을 더 잘 예측하는 것을 볼 수 있다.

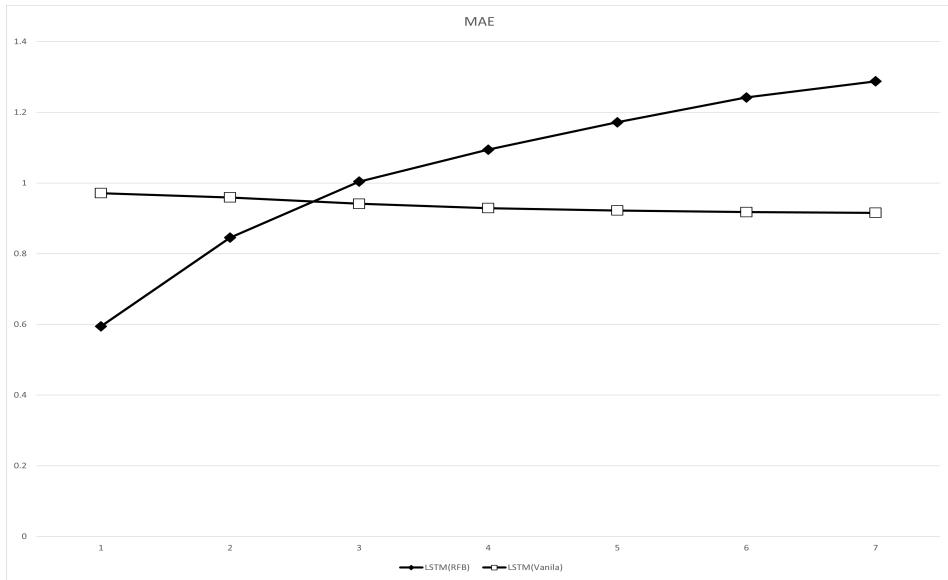


그림 12: lead time에 따른 LSTM모델의 평균 절대 오차

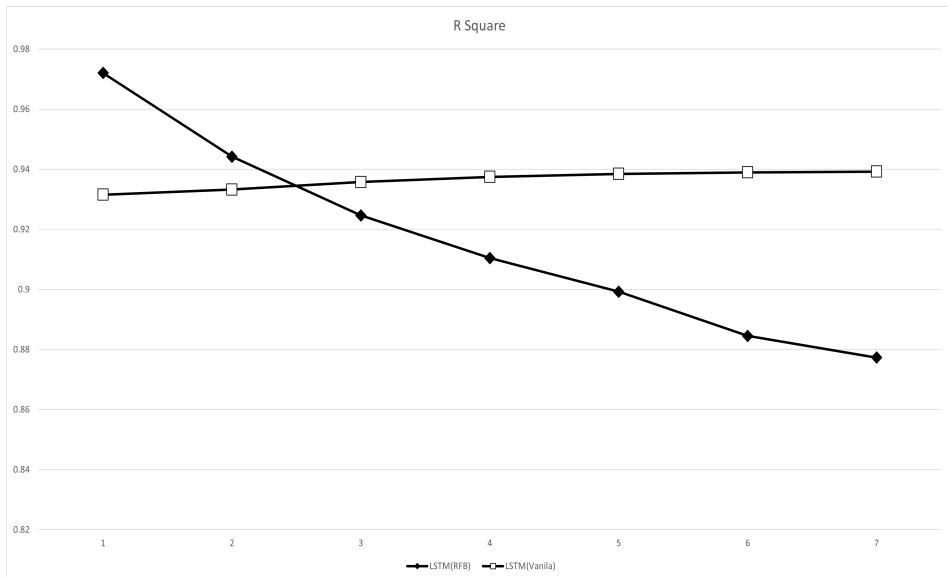


그림 13: lead time에 따른 LSTM모델의 R<sup>2</sup>

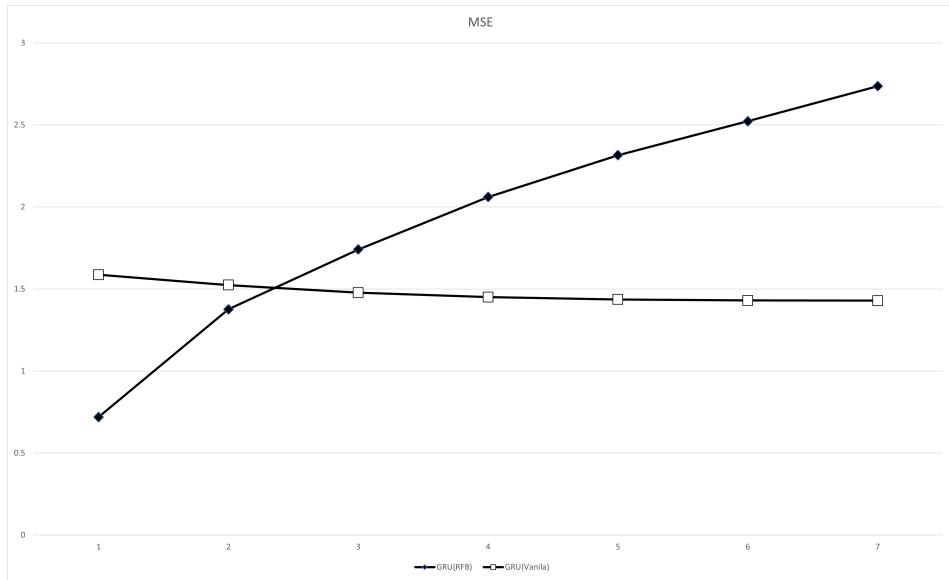


그림 14: lead time에 따른 LSTM모델의 평균 제곱 오차

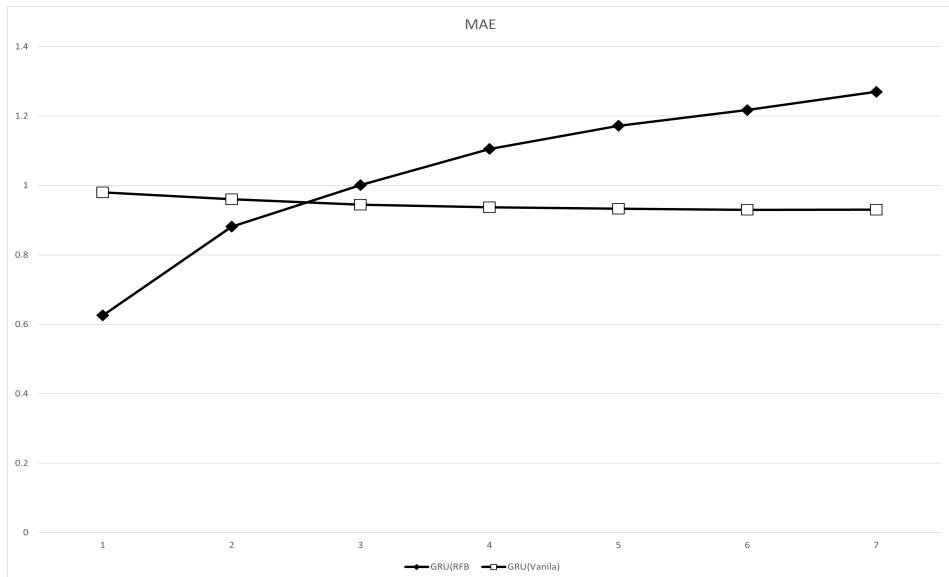


그림 15: lead time에 따른 LSTM모델의 평균 절대 오차

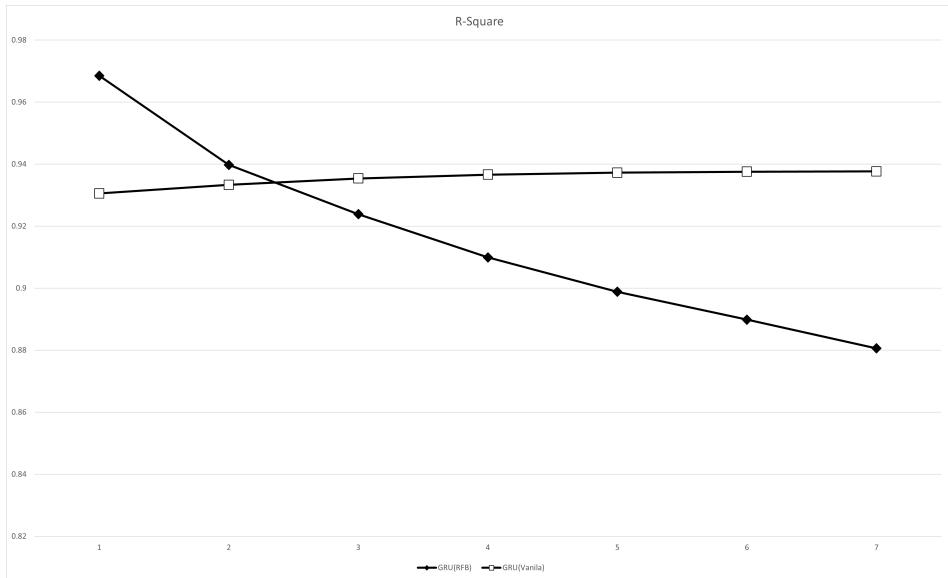


그림 16: lead time에 따른 LSTM모델의  $R^2$

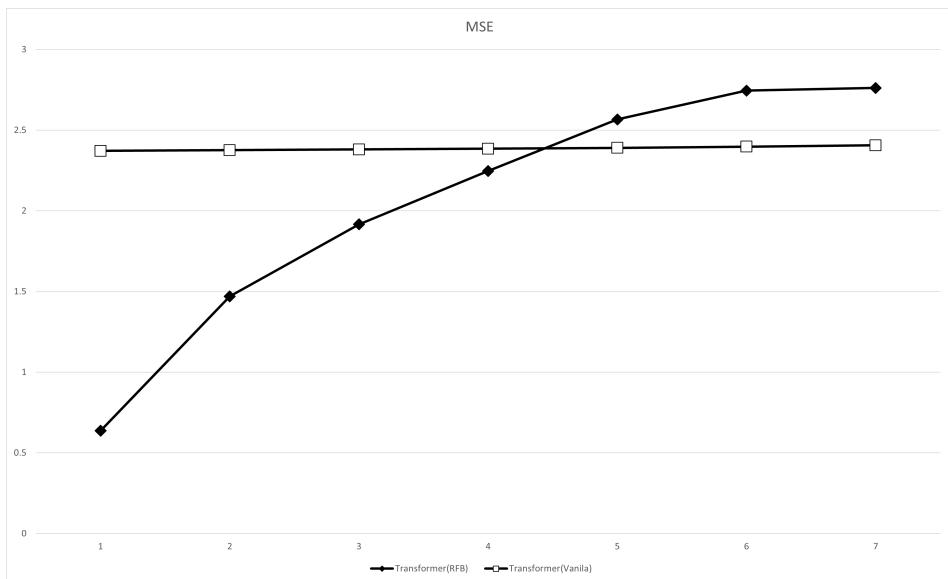


그림 17: lead time에 따른 LSTM모델의 평균 제곱 오차

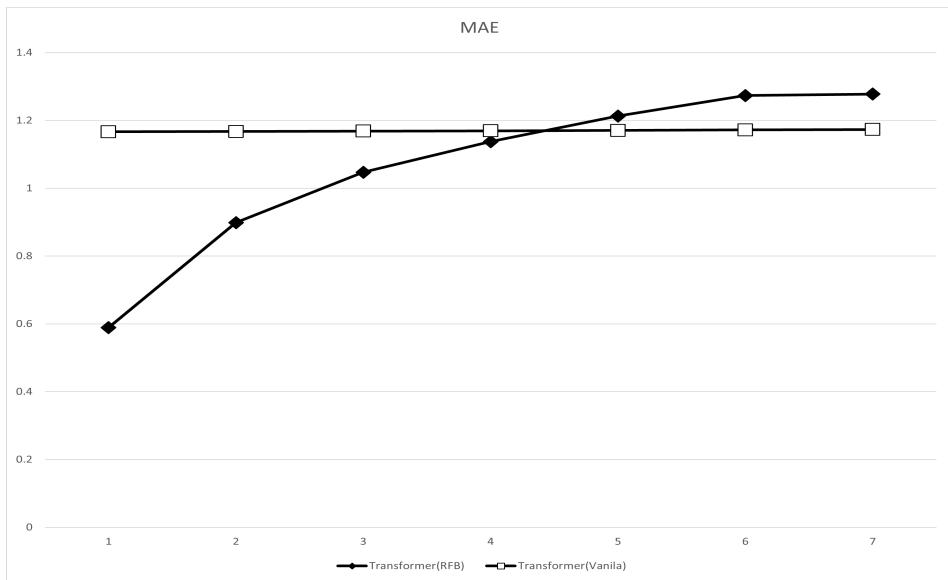


그림 18: lead time에 따른 LSTM모델의 평균 절대 오차

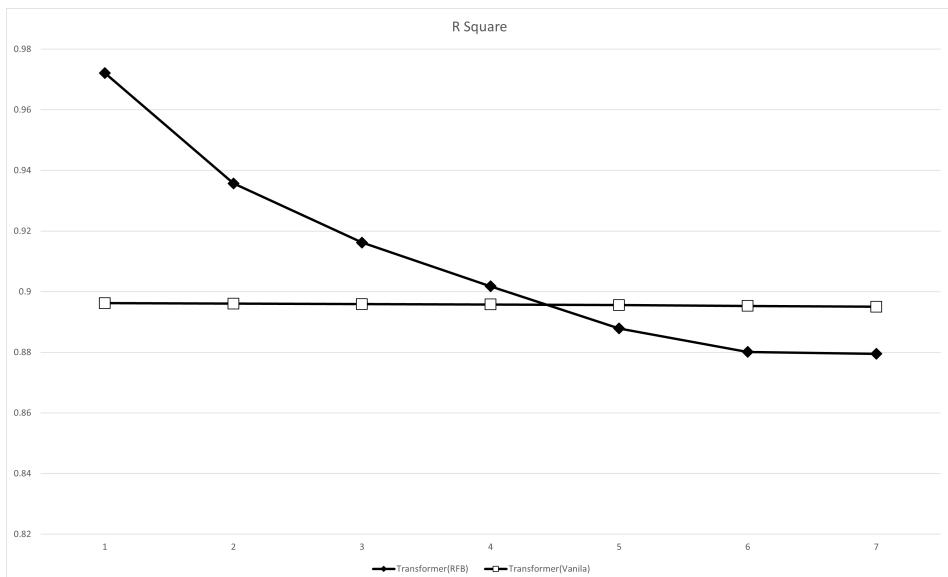


그림 19: lead time에 따른 LSTM모델의  $R^2$

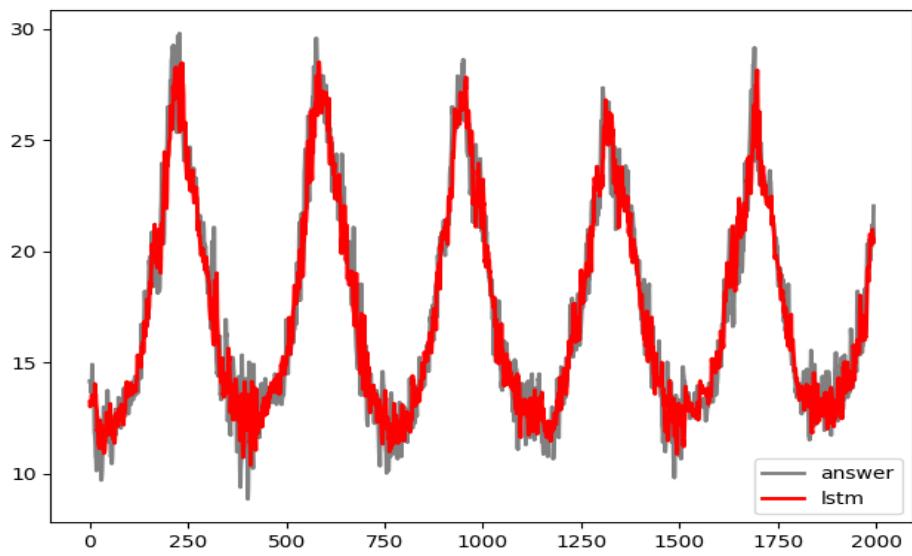


그림 20: LSTM(Vanila)모델의 lead time 1일 때의 실험결과

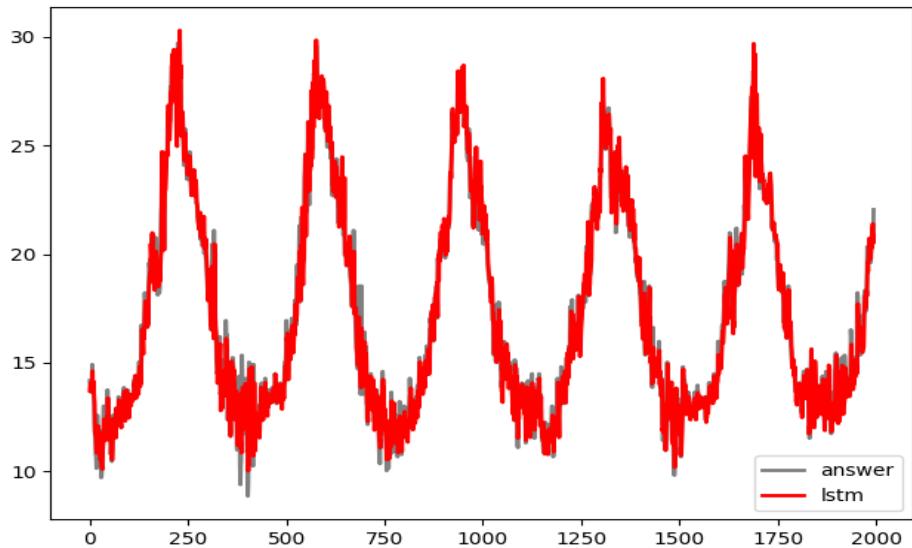


그림 21: LSTM(RFB)모델의 lead time 1일 때의 실험결과

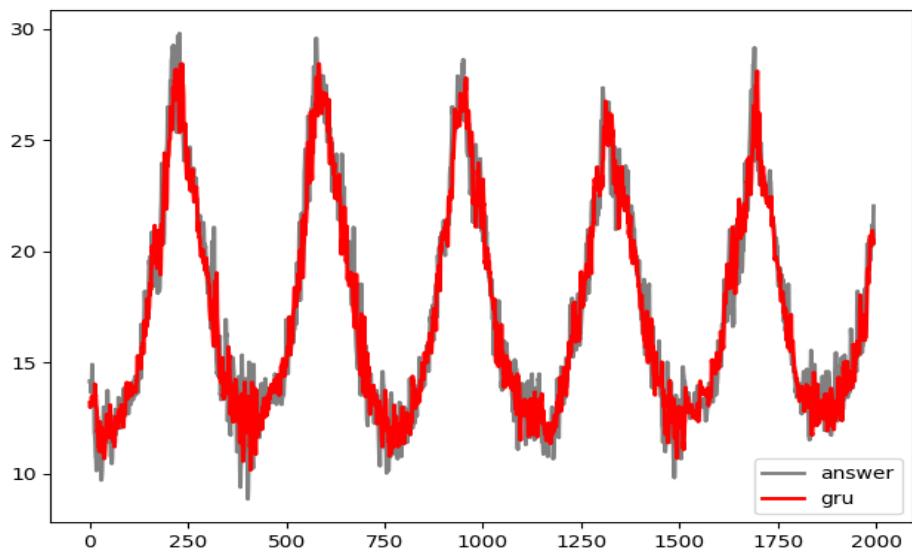


그림 22: GRU(Vanila)모델의 lead time 1일 때의 실험결과

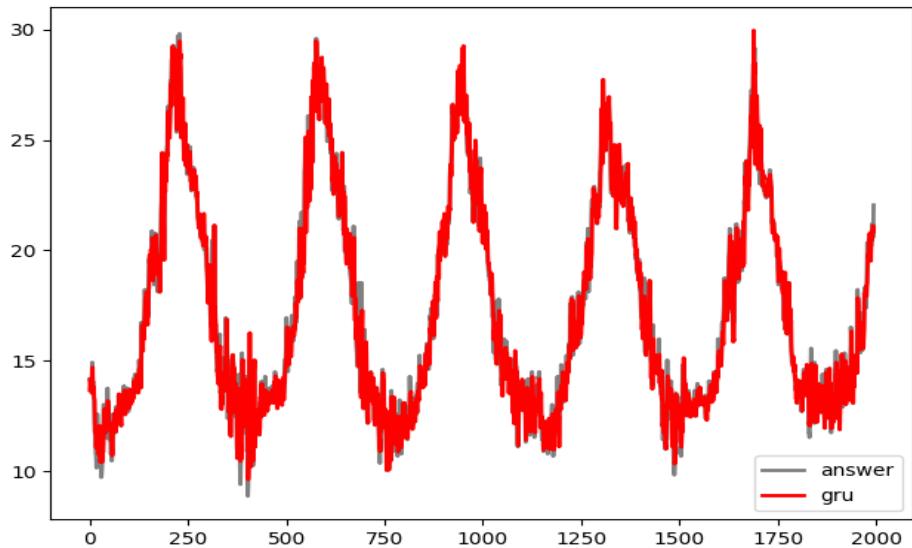


그림 23: GRU(RFB)모델의 lead time 1일 때의 실험결과

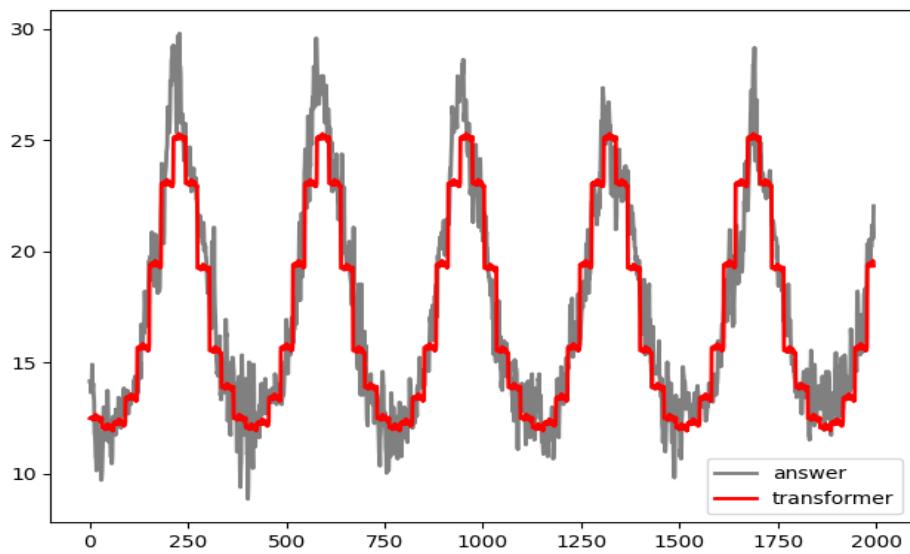


그림 24: Transformer(Vanila)모델의 lead time 1일 때의 실험결과

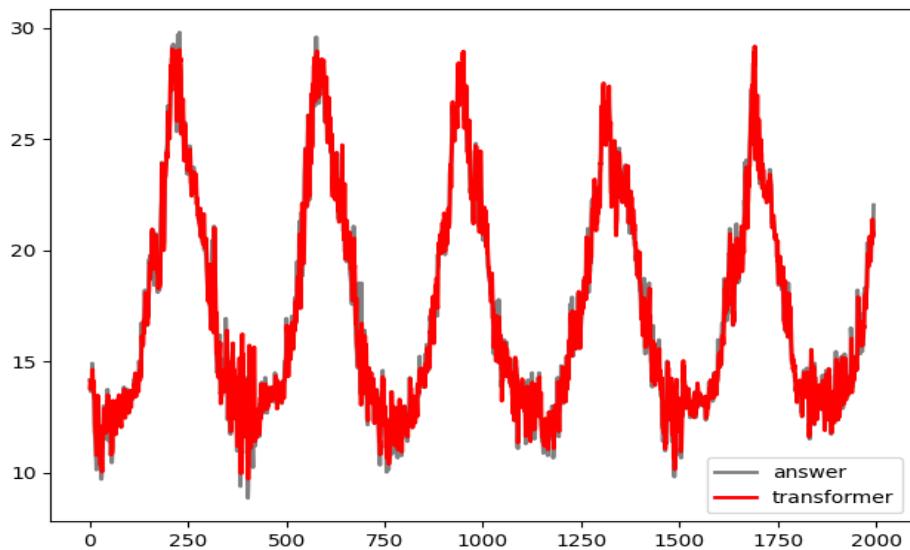


그림 25: Transformer(RFB)모델의 lead time 1일 때의 실험결과

## VII 결론

시계열 데이터 예측 모델을 만들 때 극값을 잘 예측하는 것은 극값으로 인해 발생하는 영향을 줄일 수 있기 때문에 중요한 문제이다. 따라서 본 연구에서는 시계열 데이터의 극값을 예측할 때 사용할 수 있는 수용 장 블록 (Receptive Field Block)기반의 인코딩 모듈 LPRFBM(Linear Projected Receptive Field Module)을 제안한다. LPRFBM은 시계열 데이터를 시계열 변수 축으로 선형사영시켜 시간적 요소 뿐 아니라 시계열 데이터의 변수까지 고려할 수 있도록 하였다. LPRFBM은 어느 시계열 예측모델의 인코딩에서도 적용할 수 있기 때문에 확장성이 높고 시간 정보를 부여하는 작업을 따로 해주지 않아도 되기 때문에 편리하게 사용할 수 있다. 여러 실험을 통해 LPRFBM은 시계열 예측 모델의 평균 제곱 오차, 평균 절대 오차,  $R^2$ 의 지표를 향상 시킬 수 있고, 시계열 예측모델이 극값을 더 잘 예측하도록 만들어 준다는 사실을 확인하였다. 미래 연구에서는 LPRFBM이 어떻게 극값 예측에 대한 성능을 향상시킬 수 있는지에 대한 이론적 측면에 대한 연구를 수행할 것이다.

## 참고 문헌

- [1] C. J. Donlon, M. Martin, J. Stark, J. Roberts-Jones, E. Fiedler, and W. Wimmer, “The operational sea surface temperature and sea ice analysis (ostia) system,” *Remote Sensing of Environment*, vol. 116, pp. 140–158, 2012.
- [2] S. Jung, Y. J. Kim, S. Park, and J. Im, “Prediction of sea surface temperature and detection of ocean heat wave in the south sea of korea using time-series deep-learning approaches,” *Korean Journal of Remote Sensing*, vol. 36, no. 5\_3, pp. 1077–1093, 2020.
- [3] J.-H. Böse, V. Flunkert, J. Gasthaus, T. Januschowski, D. Lange, D. Salinas, S. Schelter, M. Seeger, and Y. Wang, “Probabilistic demand forecasting at scale,” *Proceedings of the VLDB Endowment*, vol. 10, no. 12, pp. 1694–1705, 2017.
- [4] T. J. Brailsford and R. W. Faff, “An evaluation of volatility forecasting techniques,” *Journal of Banking & Finance*, vol. 20, no. 3, pp. 419–438, 1996.
- [5] N. Laptev, J. Yosinski, L. E. Li, and S. Smyl, “Time-series extreme event forecasting with neural networks at uber,” in *International conference on machine learning*, vol. 34, 2017, pp. 1–5.
- [6] R. Wen, K. Torkkola, B. Narayanaswamy, and D. Madeka, “A multi-horizon quantile recurrent forecaster,” *arXiv preprint arXiv:1711.11053*, 2017.
- [7] M. Mudelsee, “Trend analysis of climate time series: A review of methods,” *Earth-science reviews*, vol. 190, pp. 310–322, 2019.

- [8] D. S. Stoffer and H. Ombao, *Special issue on time series analysis in the biological sciences*, 2012.
- [9] E. J. Topol, “High-performance medicine: The convergence of human and artificial intelligence,” *Nature medicine*, vol. 25, no. 1, pp. 44–56, 2019.
- [10] G. E. Box and D. A. Pierce, “Distribution of residual autocorrelations in autoregressive-integrated moving average time series models,” *Journal of the American statistical Association*, vol. 65, no. 332, pp. 1509–1526, 1970.
- [11] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.
- [12] E. S. Gardner Jr, “Exponential smoothing: The state of the art,” *Journal of forecasting*, vol. 4, no. 1, pp. 1–28, 1985.
- [13] P. R. Winters, “Forecasting sales by exponentially weighted moving averages,” *Management science*, vol. 6, no. 3, pp. 324–342, 1960.
- [14] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and regression trees*. Routledge, 2017.
- [15] A. J. Smola and B. Schölkopf, “A tutorial on support vector regression,” *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.
- [16] J. Friedman, T. Hastie, R. Tibshirani, *et al.*, *The elements of statistical learning*, 10. Springer series in statistics New York, 2001, vol. 1.
- [17] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*, 3. MIT press Cambridge, MA, 2006, vol. 2.
- [18] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

- [19] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” *arXiv preprint arXiv:1406.1078*, 2014.
- [20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in neural information processing systems*, 2017, pp. 5998–6008.
- [21] Y. Wang, A. Smola, D. Maddix, J. Gasthaus, D. Foster, and T. Januschowski, “Deep factors for forecasting,” in *International conference on machine learning*, PMLR, 2019, pp. 6607–6617.
- [22] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, “Deepar: Probabilistic forecasting with autoregressive recurrent networks,” *International Journal of Forecasting*, vol. 36, no. 3, pp. 1181–1191, 2020.
- [23] S. S. Rangapuram, M. W. Seeger, J. Gasthaus, L. Stella, Y. Wang, and T. Januschowski, “Deep state space models for time series forecasting,” *Advances in neural information processing systems*, vol. 31, pp. 7785–7794, 2018.
- [24] K. Rasul, A.-S. Sheikh, I. Schuster, U. Bergmann, and R. Vollgraf, “Multivariate probabilistic time series forecasting via conditioned normalizing flows,” *arXiv preprint arXiv:2002.06103*, 2020.
- [25] Y. Gal, “Uncertainty in deep learning,” 2016.
- [26] Q. Zhang, H. Wang, J. Dong, G. Zhong, and X. Sun, “Prediction of sea surface temperature using long short-term memory,” *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 10, pp. 1745–1749, 2017.

- [27] Y. Yang, J. Dong, X. Sun, E. Lima, Q. Mu, and X. Wang, “A cfcc-lstm model for sea surface temperature prediction,” *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 2, pp. 207–211, 2017.
- [28] C. Xiao, N. Chen, C. Hu, K. Wang, J. Gong, and Z. Chen, “Short and mid-term sea surface temperature prediction using time-series satellite data and lstm-adaboost combination approach,” *Remote Sensing of Environment*, vol. 233, p. 111 358, 2019.
- [29] Z. Zhang, X. Pan, T. Jiang, B. Sui, C. Liu, and W. Sun, “Monthly and quarterly sea surface temperature prediction based on gated recurrent unit neural network,” *Journal of Marine Science and Engineering*, vol. 8, no. 4, p. 249, 2020.
- [30] L. Xu, Y. Li, J. Yu, Q. Li, and S. Shi, “Prediction of sea surface temperature using a multiscale deep combination neural network,” *Remote Sensing Letters*, vol. 11, no. 7, pp. 611–619, 2020.
- [31] S. Liu, D. Huang, *et al.*, “Receptive field block net for accurate and fast object detection,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 385–400.
- [32] B. Lim and S. Zohren, “Time-series forecasting with deep learning: A survey,” *Philosophical Transactions of the Royal Society A*, vol. 379, no. 2194, p. 20 200 209, 2021.
- [33] R. B. Cleveland, W. S. Cleveland, J. E. McRae, and I. Terpenning, “Stl: A seasonal-trend decomposition,” *J. Off. Stat.*, vol. 6, no. 1, pp. 3–73, 1990.
- [34] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.

- [35] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, “On the properties of neural machine translation: Encoder-decoder approaches,” *arXiv preprint arXiv:1409.1259*, 2014.
- [36] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” *arXiv preprint arXiv:1412.3555*, 2014.
- [37] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” in *Advances in neural information processing systems*, 2014, pp. 3104–3112.
- [38] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” *arXiv preprint arXiv:1409.0473*, 2014.

## A Study on Encoding Module for Predicting Extreme Value in Time Series Forecasting

Mingi Kim

Department of Artificial Intelligence  
Graduate School, Kyungpook National University  
Daegu, Korea  
(Supervised by Professor Jaeil Kim)

### Abstract

Sea surface temperature is the most important variable in numerical climate prediction models and ocean prediction models, and determines the global ocean heat distribution and ocean-atmosphere circulation. Sea surface temperature changes are caused by global climate change and interactions between ocean systems. Recently, various external factors are raising the sea surface temperature on the Korean Peninsula, which is causing enormous damage to fisheries such as coastal fishing and aquaculture. If we predict and prepare for such a high-temperature phenomenon in advance, it will be possible to reduce the damage caused by the high-temperature phenomenon. In this study, we propose a receptive field block-based linear projection encoding module LPRFBM (Linear Projected Receptive Field Block Module), which has excellent performance in predicting extreme values such as high sea surface temperature. LPRFBM linearly projects time series data onto the time series variable axis to consider not only time elements but also variables of time series data. Because LPRFBM can be applied to encoding of any time series forecasting model, it is highly expandable and because it does not require a separate task to assign a time sequence to time series data, it is very convenient. Based on the quantitative and qualitative metrics of several experiments, it can be said that LPRFBM can improve the performance of predicting extreme values in time series forecasting.