KIPS Trans. Softw. and Data Eng.
Vol.12, No.6 pp.251~258
ISSN: 2287-5905 (Print), ISSN: 2734-0503 (Online)

https://doi.org/10.3745/KTSDE.2023.12.6.251

A Study on Peak Load Prediction Using TCN Deep Learning Model

Lee Jung Il†

ABSTRACT

It is necessary to predict peak load accurately in order to supply electric power and operate the power system stably. Especially, it is more important to predict peak load accurately in winter and summer because peak load is higher than other seasons. If peak load is predicted to be higher than actual peak load, the start-up costs of power plants would increase. It causes economic loss to the company. On the other hand, if the peak load is predicted to be lower than the actual peak load, blackout may occur due to a lack of power plants capable of generating electricity. Economic losses and blackouts can be prevented by minimizing the prediction error of the peak load. In this paper, the latest deep learning model such as TCN is used to minimize the prediction error of peak load. Even if the same deep learning model is used, there is a difference in performance depending on the hyper-parameters. So, I propose methods for optimizing hyper-parameters of TCN for predicting the peak load. Data from 2006 to 2021 were input into the model and trained, and prediction error was tested using data in 2022. It was confirmed that the performance of the deep learning model optimized by the methods proposed in this study is superior to other deep learning models.

Keywords : Peak Load Prediction, Artificial Neural Network, Deep Learning, Hyper Parameter Optimization

TCN 딥러닝 모델을 이용한 최대전력 예측에 관한 연구

이 정 일†

요 약

안정적으로 전력을 공급하고 전력계통을 운영하기 위해서는 최대전력을 정확히 예측해야 한다. 특히, 최대전력이 높게 발생하는 겨울과 여름에는 그 중요성이 매우 커진다. 최대전력을 실제 수요보다 높게 예측하면 발전소 기동 비용이 증가하여 경제적 손실이 발생하고, 최대전력을 실제 수요보다 낮게 예측하면 기동이 가능한 발전소가 부족하여 정전이 발생할 수 있다. 최대전력의 예측 오차를 최소화함으로써 경제적 손실과 정전을 예방할 수 있다. 본 논문에서는 최대전력 예측의 오차를 최소화하기 위하여 최신 딥러닝 모델인 TCN을 이용한다. 딥러닝 모델은 하이퍼 파라미터를 어떻게 설정하느냐에 따라 성능 차이가 발생하므로, TCN의 하이퍼 파라미터를 최적화하는 방법을 제안한다. 2006년부터 2021년까지의 데이터를 입력하여 모델을 훈련하고, 2022년의 데이터를 이용하여 예측 오차를 실험하였다. 실험을 수행한 결과 본 논문에서 제안한 최적화 방법을 이용한 TCN 모델의 성능이 다른 딥러닝 모델보다 성능이 우수한 것을 확인하였다.

키워드 : 최대전력 예측, 인공 신경망, 딥러닝, 하이퍼 파라미터 최적화

1. 서 론

전력을 안정적으로 공급하고 정전을 예방하기 위해서는 다음 날의 최대전력을 정확히 예측하여야 한다. 최대전력을 실제 수요보다 높게 예측하면 발전소 기동 비용이 증가하여 경제적 손실이 발생한다. 반면에, 최대전력을 실제 수요보다 낮게 예측하면 기동이 가능한 발전소가 부족하여 정전이 발생할 수 있다. 최대전력의 예측 오차를 최소화함으로써 경제적 손실과 정전을 예방할 수 있다.

※ 이 논문은 한국전력공사 전력연구원의 디지털 플랫폼 정합화 및 전력 AI엔진 기술 개발 연구과제에 의하여 연구되었음.
† 정 회 원 : 한국전력공사 전력연구원 선임연구원
Manuscript Received : March 6, 2023
First Revision : April 17, 2023
Accepted : April 26, 2023
* Corresponding Author : Lee Jung Il(jlee@kepco.co.kr)

최대전력이 공급능력을 초과하면 전력 계통의 보호를 위하여 지역단위의 정전이 불가피하다. 예비력이 부족한 전력 수급 위기 상황에서 전력 계통을 효율적으로 운영하고 정전을 예방하기 위해서는 최대전력을 정확히 예측하고, 예측 결과에 따라 적절한 수요관리를 시행하여 수요를 감축하여야 한다. 수요관리란 전력 수급 불균형 상황에 대비하여 한전과 고객이 약정을 체결하고 전력 수급 악화 시 한전의 요청에 따라 고객이 일정 수준 이상의 전력 사용을 줄이는 경우 지원금을 지급하는 프로그램이다.

전력수요 예측 방법은 통계적인 기법과 기계학습 기법으로 나누어 볼 수 있다. 통계적인 기법으로는 선형회귀분석과 자기상관회귀모형(ARIMA)가 대표적이다. 기계학습 기법으로는 과거에는 결정트리 모델이 주로 이용되었으나, 최근에는 LSTM이나 TCN과 같은 딥러닝 모델의 이용이 늘어나고 있다. 딥러닝 모델은 아키텍처를 구성하는 방법과 하이퍼 파라

※ This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/ licenses/by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

미터의 최적화 여부에 따라 성능에 큰 차이를 보이므로 최적화가 필수적이다. 본 논문에서는 합성곱을 기반으로 하는 시계열 딥러닝 모델인 TCN을 이용하여 최대전력의 예측 정확도를 개선하는 방법에 대하여 제안하고, 실험 결과를 기술한다.

2. 관련 연구 및 데이터의 탐색적 분석

2.1 최대전력 예측 모델

최대전력 예측의 연구사례를 살펴보면 통계분석과 기계학습 방법이 주로 이용되었다. 예측변수는 최대전력이고, 모델에 입력되는 특성으로는 계절적인 요인을 나타내는 범주형 특성과 예측변수에 영향을 미치는 외부요인을 나타내는 수치형 특성으로 나누어진다. 범주형 특성으로는 월, 요일, 공휴일 등이 있으며, 수치형 특성으로는 기온, 풍속, 습도, 과거 최대전력 등이 있다. 예측 모델의 성능은 Equation (1)과 같이 MAPE(Mean Average Percentage Error)를 통하여 검증한다. MAPE는 모델의 성능을 퍼센트로 확인할 수 있는 장점이 있어서 최대전력 예측 모델의 비교 평가에 주로 이용된다.

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{y_t - \hat{y_t}}{y_t}\right| \times 100\% \quad (1)$$

최근 연구를 중심으로 통계분석, 기계학습, 앙상블 방법을 이용한 최대전력 예측 연구를 소개하면 다음과 같다. 이근철 등(2017)[1]은 요일, 계절, 공휴일과 같은 범주형 특성과 평균 기온, 최저기온, 최고기온, 과거 기간의 최대전력 특성을 입력하는 회귀모형을 제안하였고, 2009년부터 2018년까지의 최대전력을 일별로 예측하였으며, 평균적으로 1.51%의 MAPE를 달성하였다. 안준영 등(2017)[2]은 일 지수, 평균기온, 최저기온, 최고기온, 일 강수량, 평균풍속, 평균 이슬점온도, 평균습도, 난방도일, 체감온도를 입력하는 딥러닝 모델을 제안하였다. 딥러닝 모델은 3개의 은닉층으로 구성되며, 각 층마다 128개의 노드를 구성하였다. 2009년도부터 2016년까지의 데이터를 이용하여 모델을 학습하고, 2017년도 데이터를 이용하여 테스트를 수행하였다. 테스트 결과 MAPE는 평균적으로 1.672%로 나타났다. 권보성 등(2020)[3]은 평균 기온, 최고기온, 최저기온, 월별 GDP, 주별 GDP, 주 특성, 특수한 주 여부를 입력하는 LSTM 앙상블 모델을 제안하였다. 앙상블 모델은 3개의 LSTM과 1개의 완전연결층이 병렬로 연결되는 방식이다. 2017년부터 2020년까지의 동계 및 하계 최대전력을 예측한 결과, MAPE는 1.2%로 다중회귀분석모형과 일반적인 심층 신경망 모델에 비하여 높은 성능을 보였다.

윤협상 등(2022)[4]은 전국 전력망에서 분리된 도서 지역의 24시간 기온과 전력수요 데이터를 입력하는 WaveNet 변형 모델을 제안하였다. 입력 데이터는 총 8층으로 구성되는 5개의 Conv1D 블록을 순차적으로 지나고, 각 블록의 출력은 ADD 블록에서 모두 합산되어 Conv1D 층에서 출력된다. 실험 결과 MAPE는 1.33%로 다른 연구에 비하여 낮은 오차를 보인다. 다만, 다른 연구와 달리 전국 전력망의 최대전력이 아닌 분리된 도서 지역의 전력수요 데이터이기 때문에 단순한 성능 비교는 어렵다.

2.2 TCN(Temporal Convolutional Network)

기존에는 시계열 데이터의 예측에 순환신경망(RNN)인 LSTM[5]을 주로 이용하였다. 순환신경망은 입력 데이터를 순차적으로 처리하여 속도가 느리고, 데이터의 시퀀스가 길어질수록 과거 데이터에 대한 기억이 감소하는 문제가 있다. 이러한 문제점을 보완하기 위하여 WaveNet[6]에서는 합성곱(Convolution)과 팽창(Dilations)을 이용하여 수용 필드(Receptive Field)를 넓히는 방법으로 TCN을 제안하였다. TCN은 1차원 CNN을 이용하여 순차적인 데이터를 처리할 수 있도록 제안된 모델이다. TCN은 1차원 합성곱 필터와 팽창을 통하여 은닉층을 구성하며, 안정적인 학습을 위하여 매 층마다 잔차 연결(Residual Connection)을 적용한다. Fig. 1은 커널 크기가 2이고, 팽창이 4인 TCN 모델의 예시이다. TCN은 Fig. 2와 같이 잔차 블록(Residual Block)을 통하여 연산을 수행한다. 잔차 블록은 Dilated Casual Convolution, Weight Normalization, ReLU, Dropout 연산을 순차적으로 2회 수행하고, 그 결과를 1x1 Convolution 연산 결과에 더한다. Fig. 2는 스택 수가 1인 TCN의 잔차 블록을 나타낸다.

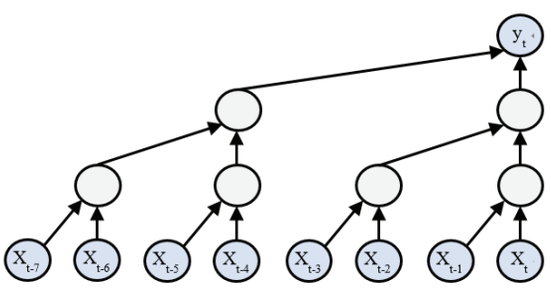

Fig. 1. TCN Architecture[6]

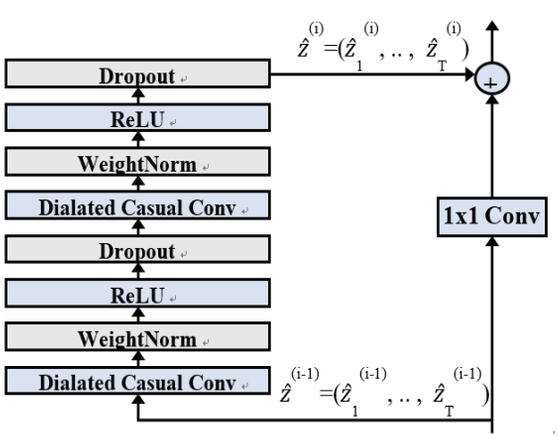

Fig. 2. TCN Residual Block[6]

TCN은 시계열 데이터의 처리에 있어서 다음과 같은 장점을 가지고 있다. 첫째, 순차적인 시퀀스에서 선행 작업이 완료되어야 다음 작업이 실행되는 RNN과 달리 각 층에서 합성곱 연산을 병렬로 처리할 수 있어서 빠른 연산을 지원한다. 둘째, 커널 크기와 팽창을 조절하여 더 넓은 수용 필드를 구성할 수 있어서 모델의 성능을 최대화할 수 있다. 마지막으로, 역전파의 방향이 시퀀스의 방향과 달라서 RNN 계열의 주요 문제인 그래디언트 폭발과 소실 문제를 방지할 수 있다. 다만, TCN은 RNN보다 더 많은 메모리가 필요한 단점이 있다. 왜냐하면, RNN은 시퀀스에 대한 상태정보를 유지하므로 과거 데이터가 메모리에서 제거되어도 상관없지만, TCN은 전체 시퀀스에 대해서 합성곱 연산을 수행하기 때문이다.

2.3 데이터의 탐색적 분석

본 연구에서 예측하고자 하는 변수는 전력거래소에서 공개하는 최대전력[7] 데이터로, 단위는 MW이다. 데이터의 수록기간은 2006년부터 2022년까지이며, 데이터의 주기는 일 단위이다. 선행 연구에서 살펴보면 전력수요는 기온과 일사량에 영향을 받는 것을 알 수 있다. 기온은 냉난방 부하에 영향을 미치고, 일사량은 태양광 발전량에 영향을 미친다. 이러한 외부요인 데이터는 미래의 예측값이 존재하여야 모델에 입력할 수 있다. 기온과 일사량은 기상 예보기관에서 다음 날의 예보를 제공하므로 모델에서 이용할 수 있다. 기상 데이터는 시간 단위로 제공되므로 일별로 최고, 평균, 최솟값으로 계산하여 이용한다. 최소 일사량은 항상 0이므로 제외한다. Fig. 3은 평균 기온, 평균 일사량과 최대전력의 시계열 추이를 나타낸다. 모든 데이터가 주기성을 보이며, 최대전력은 시간에 따라 값이 증가하는 추세를 보인다. Table 1은 입력변수 및 출력 데이터의 분산 통계정보를 나타낸다. Table 1의 Mean은 평균값, Std는 표준편차, Min은 최솟값, Max는 최댓값을 나타내며, 25%, 50%, 75%는 각각 첫 번째, 두 번째, 세 번째 4분위수를 나타낸다.

Table 1. Statistical Information Showing the Distribution of Temperature, Insolation and Peak Load Data

	Input variables					Output
	Max Temp-erature	Mean Temp-erature	Min Temp-erature	Max Insolat-ion	Mean Insolat-ion	Peak load
Mean	17.8	13.7	10.1	2.0	0.6	63282
Std	9.6	9.6	9.9	0.7	0.3	10322
Min	-8.9	12.2	-15.6	0.1	0.0	31589
25%	9.7	5.6	1.7	1.5	0.4	55549
50%	19.4	14.8	10.7	2	0.6	63140
75%	26.2	22.3	19	2.6	0.8	70120
Max	37.3	32.1	28.1	3.7	1.3	94509

최대전력은 계절에 따라 특성이 달라서 계절별로 나누어 상관관계를 분석하였으며, 결과는 Table 2와 같다. 상관분석 결과를 살펴보면, 모든 입력변수가 여름에 최대전력과 가장 높은 상관관계를 보이고, 변수 중에서는 평균 기온의 상관계수가 가장 크고, 최대 일사량의 상관계수가 가장 작은 것으로 나타났다.

Table 2. Correlation Coefficient Between Input Variables and Peak Load by Season

Input Variables	Correlation coefficient			
	Spring	Summer	Fall	Winter
Max Temperature	-0.34	0.55	-0.43	-0.36
Mean Temperature	-0.34	0.58	-0.47	-0.38
Min Temperature	-0.3	0.49	-0.47	-0.36
Max Insolation	-0.09	0.31	-0.12	0.11
Mean Insolation	-0.1	0.36	-0.14	0.09

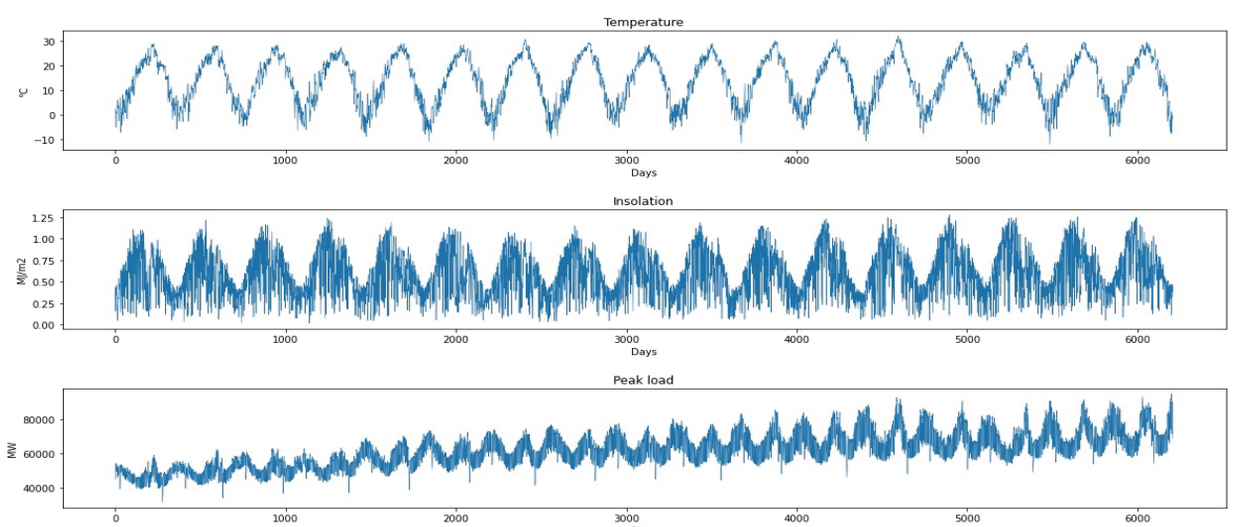

Fig. 3. Time Series Trend of Temperature, Insolation and Peak Load Data

3. 하이퍼 파라미터 최적화

인공 신경망 모델을 다양한 분야의 데이터에 적용할 수 있는 이유는 신경망 모델의 구성과 학습을 위한 하이퍼 파라미터를 조정할 수 있기 때문이다. 하이퍼 파라미터를 어떻게 조정하느냐에 따라 같은 데이터를 입력하더라도 출력이 다를 수 있다. 즉, 입력하는 데이터에 최적화된 하이퍼 파라미터를 탐색하여 신경망 모델을 설계하고 훈련함으로써 성능을 높일 수 있다. TCN 모델의 하이퍼 파라미터는 아키텍처와 관련된 하이퍼 파라미터와 훈련과 관련된 하이퍼 파라미터로 구분할 수 있다. 아키텍처와 관련된 하이퍼 파라미터로는 커널 크기, 팽창, 잔차 블록 스택 수, 필터 개수가 있다. 훈련과 관련된 하이퍼 파라미터는 학습률, 배치 크기가 있다. 본 연구에서는 TCN 모델의 최적 설계 및 학습을 위하여 그리드 탐색방식[8]으로 하이퍼 파라미터를 탐색한다.

3.1 커널 크기 및 팽창 탐색

TCN 모델은 합성곱 신경망을 시계열 예측에 이용할 수 있도록 개발된 모델이다. 합성곱 연산을 위하여 커널 크기(Kernel Size)와 팽창(Dilations)을 결정하여야 한다. 커널 크기는 1차원 공간에서의 합성곱 연산을 위한 필터의 범위를 지정하고, 팽창은 층을 얼마나 깊게 구성할지를 결정한다. 커널 크기와 팽창이 너무 크면 과거의 더 많은 시계열을 연산에 포함하게 된다. 과거의 불필요한 시계열을 모델에 입력하면 모델의 오차가 증가할 수 있다. 그러므로 적정한 커널 크기와 팽창을 탐색해야 한다. Table 3은 커널 크기와 팽창에 따른 모델의 오차를 나타낸다. 커널 크기가 2이고, 팽창이 4일 때 오차가 가장 작고, 커널 크기가 4이고, 팽창이 8일 때 오차가 가장 높게 나타난다.

3.2 잔차 블록의 스택 개수 탐색

잔차 블록의 스택 개수가 증가하면 모델의 복잡도가 증가하여 모델의 성능을 높일 수 있지만, 과적합이 발생할 수 있다. 그러므로 탐색을 통하여 적정한 스택 개수를 결정할 필요가 있다. Fig. 4는 스택 개수에 따른 모델의 오차를 나타낸다. 스택 개수가 1일 때 오차가 가장 작고, 스택 수가 3일 때 오차가 가장 크게 나타났다.

Table 3. MAPE Comparison by Kernel Size and Dilations

Kernel Size	Dilations	Receptive Field	MAPE(%)
2	2	4	1.832
2	4	8	1.721
2	8	16	1.873
3	2	6	1.812
3	4	12	1.963
3	8	24	2.301
4	2	8	1.782
4	4	16	2.346
4	8	32	4.455

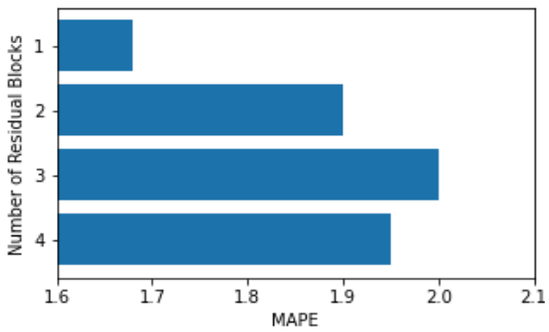

Fig. 4. MAPE Comparison by Number of Residual Blocks

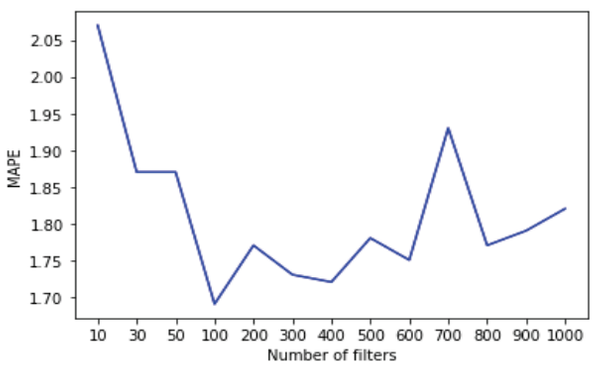

Fig. 5. MAPE Comparison by Number of Filters

3.3 필터 개수 탐색

TCN 모델의 하이퍼 파라미터인 필터 개수는 모델의 복잡도와 관련이 있다. 필터의 개수가 많을수록 모델의 복잡도가 증가하며, 복잡도가 낮은 모델보다 성능을 높일 수 있다. 다만, 모델의 복잡도가 너무 커지면 훈련 데이터에 과적합 되어 예측 정확도가 낮아지는 문제가 있으며, 훈련해야 하는 파라미터의 개수가 증가하므로 더 많은 메모리 자원을 소비한다. 따라서, 적절한 필터 개수 탐색이 필요하다. Fig. 5는 필터 개수에 따른 모델의 오차를 나타내며, 필터 개수가 100일 때 오차가 가장 낮게 나타났다.

3.4 학습률 탐색

딥러닝 모델은 수많은 파라미터로 구성된다. 최초 훈련이 시작되면 파라미터는 무작위로 초기화된다. 초기화된 파라미터를 시작으로 비용함수가 최소화되는 방향으로 반복 학습하고, 파라미터를 조정한다. 이러한 반복적인 학습 과정에서 가장 중요한 요소는 스텝의 크기이며, 학습률에 따라 스텝의 크기가 결정된다. 학습률이 너무 낮으면 모델의 훈련 반복과정을 더 많이 진행해야 하므로 모델의 훈련 시간이 오래 걸리는 단점이 있다. 그러므로 적정한 학습률을 탐색할 필요가 있다. Fig. 6은 ADAM[9] 옵티마이저의 학습률에 따른 모델의 오차를 나타낸다. 학습률이 0.0005일 때 오차가 가장 작고, 학습률이 0.00001일 때 오차가 가장 높게 나타났다.

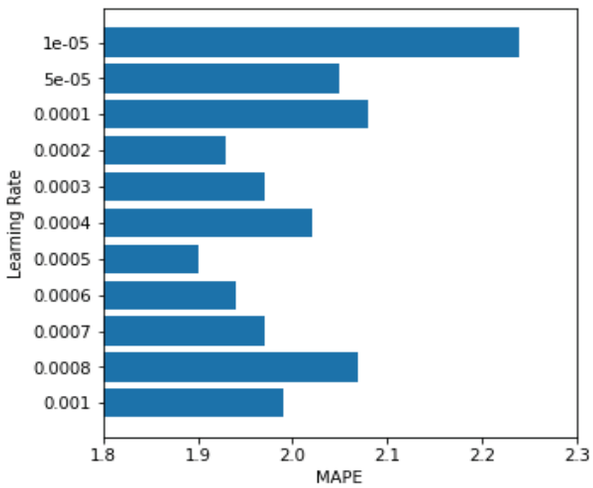

Fig. 6. MAPE Comparison by Learning Rate

3.5 배치 크기 탐색

딥러닝 모델은 경사 하강법(Gradient Descent)을 통하여 비용이 최소화되는 파라미터를 계산한다. 경사 하강법에는 배치 경사 하강법, 확률적 경사 하강법, 미니배치 경사 하강법이 있다. 배치 경사 하강법은 전체 훈련 데이터 세트를 훈련과정에서 매회 반복하여 입력하는 방법으로 속도가 느린 단점이 있다. 확률적 경사 하강법은 하나의 샘플을 기반으로 그래디언트를 계산하기 때문에 속도는 개선되었으나, 불안정한 단점이 있다. 미니배치 경사 하강법은 임의의 작은 샘플 데이터 세트에 대하여 그래디언트를 계산한다. 행렬 연산에 최적화된 GPU와 같은 하드웨어를 이용하면 미니배치 경사 하강법이 성능 측면에서 가장 효율적이다. 또한, 미니배치를 어느 정도 크게 하면 파라미터 계산 과정에서 덜 불규칙하게 움직인다. 모델 훈련에서 가장 효과적인 미니배치 경사 하강법을 이용하기 위해서는 배치 크기를 결정하여야 한다. Fig. 7은 배치 크기에 따른 모델의 오차를 나타낸다. 배치 크기가 2부터 16까지는 오차가 감소하고, 배치 크기가 16일 때 가장 작은 오차를 나타낸다. 반면에, 배치 크기가 16부터 256까지는 오차가 증가한다.

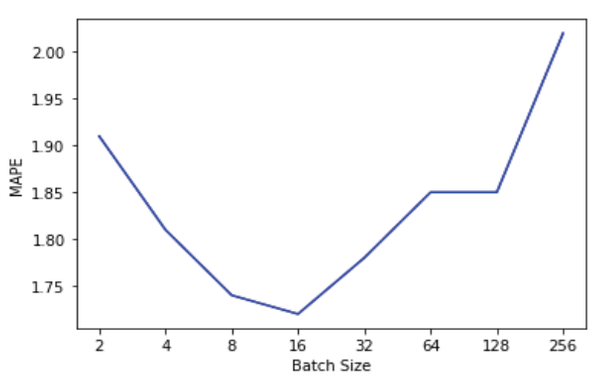

Fig. 7. MAPE Comparison by Batch Size

4. 모델 훈련 및 예측결과

4.1 데이터 입력 기간

복잡도가 높은 딥러닝 모델의 수많은 파라미터를 최적화하기 위해서는 많은 샘플 데이터가 필요하다. 하지만, 현실 세계에서는 오랜 시간이 흐른 데이터는 대부분 보존되지 않는다. 또한, 시계열 데이터는 시간이 지날수록 데이터의 분포가 달라질 수 있다. 그러므로 데이터를 과거 어느 시점부터 입력할 것인지 결정할 필요가 있다. Fig. 8은 훈련에 입력되는 샘플 데이터의 기간을 년 단위로 나누어 오차를 나타낸다. 모델에 입력되는 샘플 데이터의 기간이 최근 1년부터 최근 5년까지의 구간에서 오차가 크게 감소하였다. 입력 데이터의 개수가 늘어남에 따라 오차가 감소하는 것을 알 수 있다. 최근 5년 이후에도 오차가 소폭 감소하며, 최근 16년 데이터를 입력했을 때 가장 낮은 오차를 보였다.

4.2 특성 선택

최대전력 예측을 위하여 범주형 특성으로는 월, 요일, 공휴일을 입력하고, 수치형 특성으로는 일 최고 일사량, 일 평균 일사량, 일 최고기온, 일 평균 기온, 일 최저기온을 입력하였다. 예측하고자 하는 최대전력과 상관관계가 없는 불필요한 특성을 제거하면 오차를 개선할 수 있다. Table 4는 전체 특성을 포함한 모델과 특성을 1개씩 제거한 모델의 오차를 나타낸다. 전체 특성을 입력한 모델의 오차가 가장 낮고, 공휴일 범주형 특성을 제거한 모델의 오차가 가장 크게 나타났다.

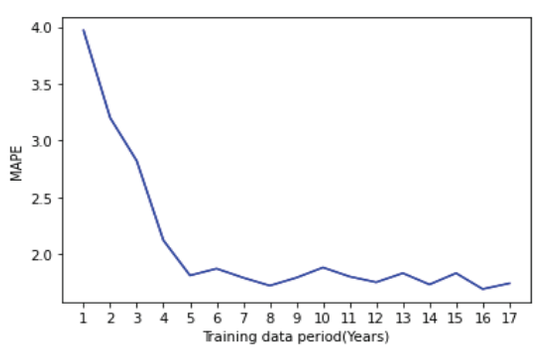

Fig. 8. MAPE Comparison by Training Data Period(Years)

Table 4. MAPE Comparison by Feature Selection

Feature Selection		MAPE(%)
All features		1.697
W/O	Max Insolation	1.822
	Mean Insolation	1.705
	Max Temperature	1.863
	Mean Temperature	1.872
	Min Temperature	1.914
	Month	1.83
	The day of week	1.912
	Holiday	2.131

4.3 모델 개수

미니배치 경사 하강법에서는 전체 샘플 데이터를 임의로 작은 샘플 데이터로 나누어 훈련한다. 그래서 같은 데이터를 입력하여도 배치 순서가 달라서 모델의 파라미터가 다르게 훈련될 수 있다. 본 연구에서는 같은 데이터를 입력하여 총 30개의 모델을 훈련한 후, 훈련된 전체 모델 중에서 손실이 가장 작은 모델을 이용한다. 손실의 계산은 Equation 2와 같이 MSE(Mean Squared Error)를 이용한다.

$$MSE = \frac{1}{n}\sum_{t=1}^{n}\left(\hat{y}_t - y_t\right)^2 \quad (2)$$

검증 단계의 손실이 예측 단계에서 그대로 재현되지 않을 수 있다. 검증 단계에서 손실이 작더라도 예측 단계에서는 손실이 커질 수 있다. 훈련된 복수의 모델을 이용하여 예측하고 그 결과를 평균함으로써 예측 단계에서의 모델의 일반화를 높일 수 있다. Fig. 9는 TCN 모델을 30회 반복 학습한 후에 모델 중에서 검증 손실이 가장 작은 복수의 모델을 이용하여 예측을 수행하고, 그 결과를 평균하여 최종 예측 값을 계산한 결과이다. 손실이 가장 작은 1개의 모델을 이용하여 예측을 수행하는 것보다, 손실이 작은 순으로 복수의 모델을 이용하여 예측을 수행하고, 그 결과를 평균하여 예측 값을 계산하는 것이 모델을 좀 더 일반화하여 성능을 높일 수 있는 것을 보여준다. 모델 8개를 이용하여 예측하고 그 결과를 평균한 경우의 오차가 가장 낮게 나타났다.

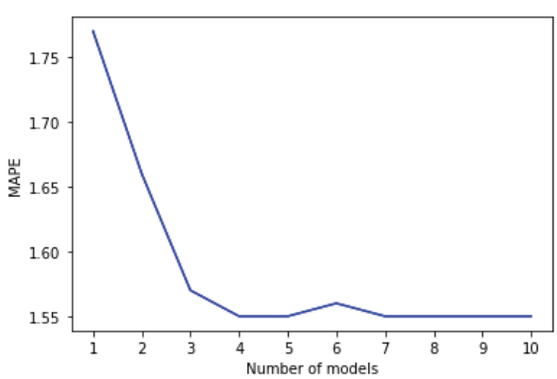

Fig. 9. MAPE Comparison by Number of Models

4.4 예측 결과

앞선 실험을 통하여 하이퍼 파라미터인 커널 크기는 2, 팽창은 4, 잔차 블록의 스택 개수는 1, 필터 개수는 100, 학습률은 0.0005, 배치 크기는 16으로 선정하였고, 데이터의 입력 기간은 최근 16년, 학습모델 개수는 8개로 선정하였다. 실험에 입력한 데이터에는 평일인 공휴일은 제외하였다. 본 연구에서 제안한 모델의 성능 비교를 위하여 RNN, LSTM, GRU[10]를 이용하여 Table 5와 같이 테스트하였다. 비교 모델인 RNN, LSTM, GRU에 모두 동일한 특성을 입력하였고, 하이퍼 파라미터인 셀 개수는 7, 은닉층의 개수는 3, 유닛의 개수는 100, 학습률은 0.0005, 배치 크기는 16으로 선정하였다. 실험 결과, RNN의 오차는 3.756%, LSTM은 2.777%, GRU는 2.863%로 나타났다. 순환신경망 중에서 RNN의 오차가 가장 크고, LSTM의 오차가 가장 작으며, LSTM과 GRU의 오차에는 큰 차이가 없다.

RNN은 순환신경망 중에서 가장 먼저 개발되었으며, 구조가 단순하여 긴 시퀀스에 대하여 그래디언트가 소실되는 문제가 있다. 이러한 이유로 LSTM과 GRU에 비하여 오차가 더 크게 나타난 것으로 판단된다. LSTM은 RNN의 단점을 해결하기 위하여 제안된 모델이다. RNN은 단순한 1개 층으로 구성되지만, LSTM은 3개의 게이트로 구성된다. LSTM의 각 게이트는 서로의 정보를 주고받으며 어떤 정보를 유지하고 어떤 정보를 버릴지를 학습함으로써 긴 시퀀스를 효율적으로 처리할 수 있다. GRU는 LSTM의 게이트를 간소화한 모델이다. LSTM은 3개의 게이트로 구성되지만, GRU는 2개의 게이트로 구성됨으로써 연산량을 줄일 수 있는 장점이 있다. 데이터의 양이 적을 때는, GRU의 성능이 조금 더 좋을 수 있지만, 데이터의 양이 많을 때는 LSTM의 성능이 조금 더 좋을 수 있다.

본 연구에서 제안하는 TCN 최적화 방법의 오차는 1.551%로 가장 작게 나타났다. 본 연구에서 제안하는 TCN 모델의 최적화 방법이 비교 대상 모델보다 성능이 좋은 것을 알 수 있다. TCN이 순환신경망 모델인 RNN, LSTM, GRU보다 오차가 작은 이유는 다음과 같다. 첫째, 순차적인 연산으로 처리되는 순환신경망과는 달리 각 층에서 합성곱 연산을 병렬로 처리하기 때문에 더 많은 파라미터를 포함하는 신경망을 구성할 수 있다. 파라미터의 개수와 모델의 성능은 비례한다. 둘째, TCN은 잔차 블록을 이용하여 깊은 층에서 데이터를 효과적으로 학습한다. 딥러닝 모델은 깊은 층에서 그래디언트 소실 문제가 발생하기 때문에 성능을 높이기 위하여 무한정 층을 깊게 구성할 수 없다. 하지만, 잔차 연산을 이용하면 그래디언트 소실 문제를 해결할 수 있고, 깊은 층을 구성함으로써 성능을 최대화할 수 있다[11]. Fig. 10은 계절별로 모델의 오차를 나타낸다. 모든 계절에서 TCN의 오차가 가장 작게 나타난다.

Table 5. Predict Result

Period	Baseline models			TCN
	RNN	LSTM	GRU	
January, 2022	3.915	2.020	3.011	1.640
February, 2022	3.819	2.713	2.913	1.605
March, 2022	2.768	2.547	3.024	1.593
April, 2022	1.646	1.798	1.733	1.186
May, 2022	2.136	2.238	2.094	1.395
June, 2022	4.665	3.970	3.619	1.994
July, 2022	5.355	3.653	3.414	1.507
August, 2022	5.185	4.021	3.990	2.022
September, 2022	6.506	5.021	4.902	2.563
Obtober, 2022	2.027	1.540	1.479	0.857
November, 2022	3.153	1.524	1.974	0.956
December, 2022	4.189	2.253	2.119	1.312
All	3.756	2.777	2.863	1.551

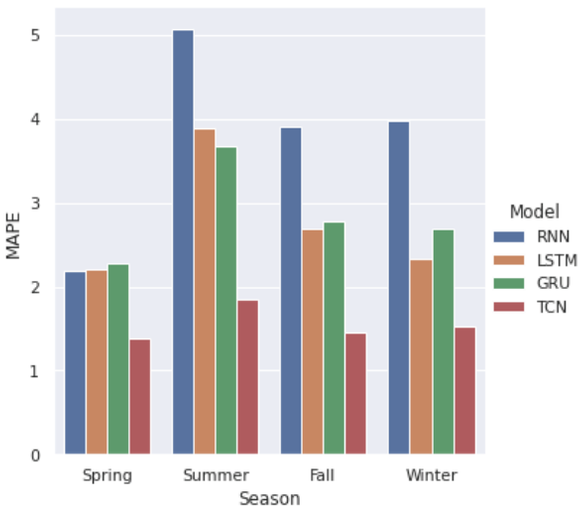

Fig. 10. MAPE Comparison by Season

5. 결 론

안정적인 전력공급과 전력 계통의 운영을 위해서는 정확한 최대전력 예측이 필수적이다. 최대전력 예측 오차에 따라 경제적·사회적 손실이 발생하거나, 정전이 발생할 수 있다. 특히, 동계와 하계에는 예비력이 부족하여 전력 수급 위기 상황이 빈번히 발생하므로, 최대전력을 정확히 예측하고, 적절한 수요관리 프로그램으로 대응하여야 한다. 최대전력의 예측 오차를 최소화함으로써 전력회사의 경제적 손실과 정전을 예방할 수 있다.

본 논문에서는 최대전력 예측의 정확도를 개선하기 위하여 최신 딥러닝 모델인 TCN을 이용하였으며, 그리드 탐색 기법을 이용하여 하이퍼파라미터를 최적화하는 방법을 제안한다. 최적화 대상은 커널 크기, 팽창, 잔차 블록의 스택 개수, 학습률, 배치 크기, 훈련 데이터 입력 기간, 입력 특성, 예측 모델 개수이다. 실험을 통하여 TCN 모델의 아키텍처를 결정하는 하이퍼 파라미터인 커널 크기는 2, 팽창은 4, 잔차 블록의 스택 개수는 1, 필터 개수는 100으로 최적 도출되었다. 커널 크기가 2이고, 팽창이 4인 경우 Receptive Field는 8로 최근 8일의 데이터가 미래 예측값에 영향을 미친다. 다음으로, 모델의 학습과 관련된 하이퍼 파라미터인 학습률은 0.0005, 배치 크기는 16이 최적해로 나타났다. 최근 16년 동안의 데이터를 입력했을 때 오차가 가장 낮았으며, 전체 특성을 입력하는 것이 성능에 도움이 되는 것을 확인하였다. 30회 반복 학습 후에 모델 중에서 검증 손실이 가장 작은 8개의 모델을 이용하여 예측을 수행하고, 그 결과를 평균하여 최종 예측값을 계산한 결과 가장 높은 성능을 보였다. 비교 대상 모델인 RNN의 MAPE는 3.756%, LSTM은 2.777%, GRU는 2.863%로 나타났고, 본 논문에서 제안하는 방법으로 최적화된 모델인 TCN의 MAPE는 1.551%로 가장 낮게 나타났다. 본 연구에서 제안하는 방법으로 최적화된 TCN 모델이 비교 대상 모델보다 성능이 우수함을 확인하였다. 향후에는 Transformer 기반의 Informer[12] 모델을 이용하여 최대전력 예측 연구를 수행할 예정이다. Informer 모델은 서로 다른 시점의 시퀀스들 간의 관계를 시계열 학습에 반영할 수 있는 장점을 가지고 있어서 LSTM이나 TCN 모델과 결합함으로써 최대전력 예측의 정확도를 향상시킬 수 있을 것으로 기대된다.

References

[1] G.-C. Lee and J. Han, "Forecasting daily peak load of domestic electricity demand," *Journal of Industrial Economics and Business*, Vol.30, No.4, pp.1205-1218, 2017.

[2] J.-Y. Ahn, S.-M. Park, and C.-B. Kim, "A study on neural network model for winter electric power demand prediction," *The Journal of Korean Institute of Information Technology*, Vol.15, No.9, pp.1-9, 2017.

[3] B.-S. Kwon and K.-B. Song, "An ensemble forecasting method of winter and summer peak load using deep neural networks," *The transactions of The Korean Institute of Electrical Engineers*, Vol.69, No.6, pp.765-771, 2020.

[4] H.-S. Yoo and S.-B. Jeong, "Electric power demand prediction using deep learning model with temperature data," *KIPS Transactions on Software and Data Engineering*, Vol.11, No.7, pp.307-314, 2022.

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, Vol.9, No.8, pp.1735-1780, 1997.

[6] A. V. D. Oord et al., "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[7] Electric Power Statistics Information System [Internet], https://epsis.kpx.or.kr/epsisnew/

[8] P., Liashchynskyi and P. Liashchynskyi, "Grid search, random search, genetic algorithm: A big comparison for NAS," *arXiv preprint arXiv:1912.06059*, 2019.

[9] D. P. Kingma and J. ba, "Adam: A Method for Stochastic Optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[10] K. Cho et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation", *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp.1724-1734, 2014.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.770-778, 2016.

[12] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol.35, No.12, pp.11106-11115, 2021.

이 정 일

https://orcid.org/0000-0002-0719-312X
e-mail : jlee@kepco.co.kr
2004년 전북대학교 컴퓨터공학과(학사)
2022년 충남대학교 컴퓨터공학과(석사)
2004년 ~ 2010년 한국전력공사
전력연구원 일반연구원
2011년 ~ 현 재 한국전력공사 전력연구원 선임연구원
관심분야 : 클라우드 컴퓨팅, 전력수요예측, 데이터 사이언스,
MLOps, AutoML