**딥러닝을 활용한 여름철 일 최대 전력 수요 예측 모델 설계**

**요약**

전력 수요량이 꾸준히 증가하면서 여름 철에는 공급 부족 현상이 일어나는 등 전력 수요 예측을 정확하게 하는 것이 최근 매우 중요한 문제로 대두되고 있다. 본 연구에서는 텐서플로우의 딥러닝 패키지를 이용하여 여름철 일 최대 전력 수요 예측을 위한 신경망(Neural Network)을 모델링하였다. 신경망의 학습을 위해서는 입력요소로 들어갈 전력 수요에 영향을 미치는 요소를 결정하는 것이 필요하기 때문에 상관분석을 통해 요소로써 기온, 체감온도, 불쾌지수, 휴일을 결정하였다. 이후 RNN/LSTM/GRU의 구조와 활성함수를 tanh와 ReLU로 바꿔가며 가장 정확한 예측 모델이 어떤 것인지 정하였다. 학습결과 ReLU함수를 활성함수로 한 GRU구조의 신경망 모델이 최대 오차율 4%정도로 가장 정확하였고, 여름 철 평균 공급예비율인 20%에 비해 만족할 만한 결과를 얻었다.

**Ⅰ. 서론**

전력 수요량은 국내 산업의 발전 및 국가 경제의 고도성장으로 인해 그 수요량이 꾸준히 증가하고 있고 지구온난화에 따른 급격한 기후변화 또한 수요량을 증가시키게 하는 원인이 되고 있다. 이에 매년 여름이면 전력 공급 부족으로 인한 대규모 정전사태인 ‘블랙아웃’에 대한 위험성이 강조된다.

전력은 경제성장 및 산업발전에 있어 매우 중요한 요소이기 때문에 수요가 많다고 하여 단순히 공급량을 늘리는 것은 큰 문제점을 일으킬 수 있다. 따라서 전력 수요량을 정확히 예측한 후, 공급량을 그에 맞게 적절히 조절하여 불필요한 전력 생산을 줄이는 것이 매우 중요하다.

과거 전력 수요 예측 방법에는 ARMA나 SARIMA 등의 통계적 방법과 인공신경망(Artificial Neural Network)을 이용한 인공지능 방법이 있다. 본 연구에서는 인공지능 방법인 딥러닝을 활용하여 전력부족 관련 이슈가 많이 발생하는 하계 기간(7 ~ 9월)의 일 최대 전력 수요 예측 모델을 모델링하려고 한다. 그 과정으로 먼저 전력 수요에 영향을 미치는 요인들을 분석하고 영향력 있는 요인들과 전력의 과거 데이터들을 수집할 것이다. 이후 연속된 데이터 학습에 적합한 구조 (RNN/LSTM/GRU)와 활성함수의 종류를 바꿔가며 어떠한 모델이 앞서 얻은 데이터들을 이용하여 신경망 모델을 학습시켰을 때 가장 정확한지 비교함으로써 일 최대 전력 수요를 예측하는 모델을 설계하고자 한다.

**Ⅱ. 분석**

**2.1** 전력 수요에 영향을 미치는 요인 설정

표 1의 2008년 ~ 2017년 최대 전력 대비 냉방부하를 보면 최대 전력에서 냉방부하가 차지하는 비율은 20 ~ 30%로 많은 비율을 차지하고 있고 그 비율은 증가하는 추세임을 알 수 있다. 냉방부하의 주원인인 에어컨 이나 선풍기와 같은 냉방장치는 단순히 더울수록 찾는 것이 아니라 더워지거나 습해질 때 또는 바람이 잘 불지 않을 경우 찾게 된다. 이로부터 냉난방 부하에 기온, 습도, 풍속이 많은 영향을 미칠 것이라 예상할 수 있다.

전력 거래소에서는 일 최대 전력 수요량 데이터를 육지와 해상으로만 구분하여 제공하고 있다. 본 논문에서는 전력 자료로 육지의 일 최대 전력 수요를 사용하고 있고 기후 자료의 경우 한 지점에서의 자료만을 사용하기에는 부정확할 수 있으므로 6개 지역의 자료를 이용하였다. 이 때 6개 지역은 전력 수요가 인구에 비례한다고 가정하고 인구수 순으로 상위 6개 지역(서울, 부산, 인천, 대구, 대전, 광주)으로 정했다. 6개 지역의 자료를 모두 사용하기에는 입력자료의 차원이 높아지기 때문에 하나로 줄일 필요성을 느껴 표 2의 인구수에 따라 가중치를 정하여 식 (1)을 통해 하나의 대표 값을 만들어 활용하였다.

|  |  |
| --- | --- |
|  | (1) |

표 1. 최대전력 대비 냉방부하 비율(단위: 만kW)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 2008 | 2009 | 2010 | 2011 | 2012 | 2013 | 2014 | 2015 | 2016 | 2017 |
| 최대전력 | 6279 | 6321 | 6989 | 7219 | 7429 | 7402 | 7605 | 7692 | 8518 | 8459 |
| 냉방부하 | 1314 | 1278 | 1539 | 1532 | 1766 | 1763 | 1528 | 1892 | 2407 | 2393 |
| 비율(%) | 20.9 | 20.2 | 22.0 | 21.2 | 23.8 | 23.8 | 20.1 | 24.5 | 28.3 | 28.3 |

표 2. 지역 별 인구수 및 수요 가중치

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 서울 | 인천 | 대전 | 광주 | 부산 | 대구 | 계 |
| 인구수 | 9806538 | 2953841 | 1495029 | 1461203 | 3453198 | 2468824 | 21638633 |
| 가중치 | 0.453196 | 0.136508 | 0.069091 | 0.067528 | 0.159585 | 0.114093 | 1 |

기온, 습도, 풍속이 독립적으로 냉방의 원인이 되기보다는 복합적으로 영향을 미치기 때문에 3가지의 기후요소 외에도 체감온도와 불쾌지수를 요소에 추가하기로 하였다.

체감온도는 일반적으로 풍속과 온도를 이용하여 외부에 있는 사람이 한기에 노출된 피부로부터 열을 빼앗길 때 느끼는 추운 정도를 나타내는 지수로 겨울에 사용하고 실제로 기상청에서도 11월~3월 동안에만 산출한다. 여름의 경우 풍속이 아닌 습도를 이용하는데 습도가 높을수록 땀의 증발을 막아 체감온도를 올리기 때문이다. 여름철 체감온도는 T를 온도(℃), H를 습도(%)라 할 때 식(2)를 이용한다.

|  |  |
| --- | --- |
|  | (2) |

불쾌지수 역시 기온과 습도를 이용하여 사람이 느끼는 온도를 표현한 것으로 온습도지수(THI)라고도 한다. 68미만인 경우 대부분의 사람들이 쾌적함을, 68이상 75미만인 경우 일부 사람들만 불쾌감을, 75이상 80미만인 경우 50%의 사람들이 불쾌감을, 80이상인 경우 대부분의 사람들이 불쾌감을 느낀다고 정의한다. 불쾌지수는 T가 온도(℃), RH를 상대습도(%)라 아래의 식 (3)을 이용한다.

|  |  |
| --- | --- |
|  | (3) |

추가로 전체 전력 사용량의 약 60%를 차지하는 산업부문은 주말 또는 공휴일에 평일보다 적게 가동되기 때문에 그 영향이 전력 사용량 변화에 뚜렷하게 드러난다. 따라서 마지막 요소로써 휴일을 추가하였다. 휴일의 경우 휴일이면 1 아니면 0인 방식으로 값을 정하였다.

**2.2** RNN/LSTM/GRU

기존의 인공신경망은 각 층의 node가 단방향으로 연결되어 있는 구조이기 때문에 이전의 결과가 현재에 계산 결과에 영향을 미치는 반복적이거나 순차적인 데이터를 학습하는 것에는 부적합하였다. 이에 과거의 결과를 기억할 수 있도록 메모리 기능을 넣은 것이 RNN(Recurrent Neural Network)이다. RNN에서 메모리 기능은 은닉층의 node가 루프를 가지도록 함으로 하여 구현된다. t시점의 입력을 , 출력을 , hidden state를 그리고 activation function을 라 하면 식 (4)와 같이 표현된다.

|  |  |
| --- | --- |
|  | (4) |

RNN에는 층의 크기와 입력, 출력의 크기에 따라 기본구조인 Vanilla RNN부터 시작하여 one to many, many to one 그리고 many to many의 구조를 가지게 된다. 일반적으로 one to many는 사진에서 단어들을 얻어내는 경우, many to many는 단어들을 가지고 감성분석을 할 경우 그리고 many to many는 번역할 경우 사용된다.

RNN 네트워크가 학습하는 Gradient Descent 알고리즘과 Backpropagation 알고리즘을 이용하는 일반적인 인공신경망과 비슷하다. 하지만 현재 시간 단계에만 의존하지 않고 여러 시간 단계에 의존하기 때문에 Backpropagation 알고리즘 대신 이를 확장시킨 BPTT(Backpropagation Through Time) 알고리즘을 사용하여 학습한다. 이로 인해 학습 시 현재 단계로부터 멀어질수록 gradient값이 잘 계산되지 않고 0으로 점점 수렴하는 Vanishing/Exploding gradient 문제가 발생하게 된다.

이 문제를 해결하기 위해 사용되는 방법으로는 activation function으로 tanh나 sigmoid함수 대신 ReLU(Rectified Linear Unit)함수를 사용하거나 RNN대신 변형구조인 LSTM(Long Short-Term Memory)이나 GRU(Gated Recurrent Unit)를 사용하는 것이다.

본 논문에서는 ReLU함수를 이용한 RNN구조, ReLU, tanh함수를 이용한 LSTM과 GRU구조를 이용하여 모델링 할 것이다.

**Ⅲ. 수행**

**3.1** 전력 수요에 영향을 미치는 요소 분석

앞서 설명한대로 전력 수요에 영향을 미치는 요소로는 온도, 풍속, 상대습도, 체감온도, 불쾌지수, 휴일을 고려하였다. 상관분석은 2016년 하계(7 ~ 9월) 자료로 Excel의 상관분석 도구를 사용하여 수행하였다. Excel의 상관분석에 사용되는 Correlation 함수는 식 (5)와 같이 상관계수를 계산한다.

|  |  |
| --- | --- |
|  | (5) |

도출된 상관계수는 -1에서 1사이의 값을 갖는데 -1에 가까울수록 ‘음의 상관관계’가 높은 것이고, 1에 가까울수록 ‘양의 상관관계’가 높은 것이다. 상관계수가 0에 가까우면 상관관계가 없다고 볼 수 있다.

상관분석의 결과, 표 3을 보면 온도와 체감온도 그리고 불쾌지수는 상관계수가 약 0.61로 뚜렷한 양적 선형관계를 가지고 휴일은 약 -0.54로 뚜렷한 음적 선형관계를 그리고 풍속과 상대습도는 거의 무시될 수 있는 선형관계를 가짐을 알 수 있다.

표 3. 상관계수

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 온도 | 풍속 | 상대습도 | 체감온도 | 불쾌지수 | 휴일 | 전력 |
| 온도 | 1 |  |  |  |  |  |  |
| 풍속 | -0.26504 | 1 |  |  |  |  |  |
| 상대습도 | -0.10413 | -0.06818 | 1 |  |  |  |  |
| 체감온도 | 0.989069 | -0.27511 | 0.040933 | 1 |  |  |  |
| 불쾌지수 | 0.989643 | -0.27415 | 0.033814 | 0.999896 | 1 |  |  |
| 휴일 | 0.0231 | -0.00644 | 0.032014 | 0.024396 | 0.02263 | 1 |  |
| 전력 | 0.614709 | -0.01582 | -0.07951 | 0.605844 | 0.606356 | -0.54464 | 1 |

상관분석 결과 관계가 없는 풍속과 상대습도를 제외한 온도, 체감온도, 불쾌지수 그리고 휴일을 신경망 모델의 입력 요소로 사용하기로 결정하였다.

**3.2** 최대 전력 수요 예측 모델링

최대 전력 수요 예측 모델로는 그림 1과 같이 모델링하였다. 입력층은 차원이 5로 [온도, 체감온도, 불쾌지수, 휴일, 전력]의 일주일 치 시간 순으로 자료를 입력으로 하고 출력층 역시 차원이 5가 되도록 구성하였다. 입력층과 출력층 사이의 은닉층은 연산 량이 너무 많게 하지 않도록 3개의 층으로 구성하였다. 마지막으로 다음 날 최대 전력 수요 예측 값을 얻기 위하여 마지막 데이터를 Fully Connected Layer와 연결하여 1차원 값을 결과 데이터로 출력하도록 만들었다.

은닉층은 RNN, LSTM, GRU구조 중 하나를 이용하여 구성할 수 있도록 하였고 RNN은 ReLU함수를, LSTM과 GRU는 ReLU와 tanh함수를 활성함수로 사용할 수 있도록 구성하였다.

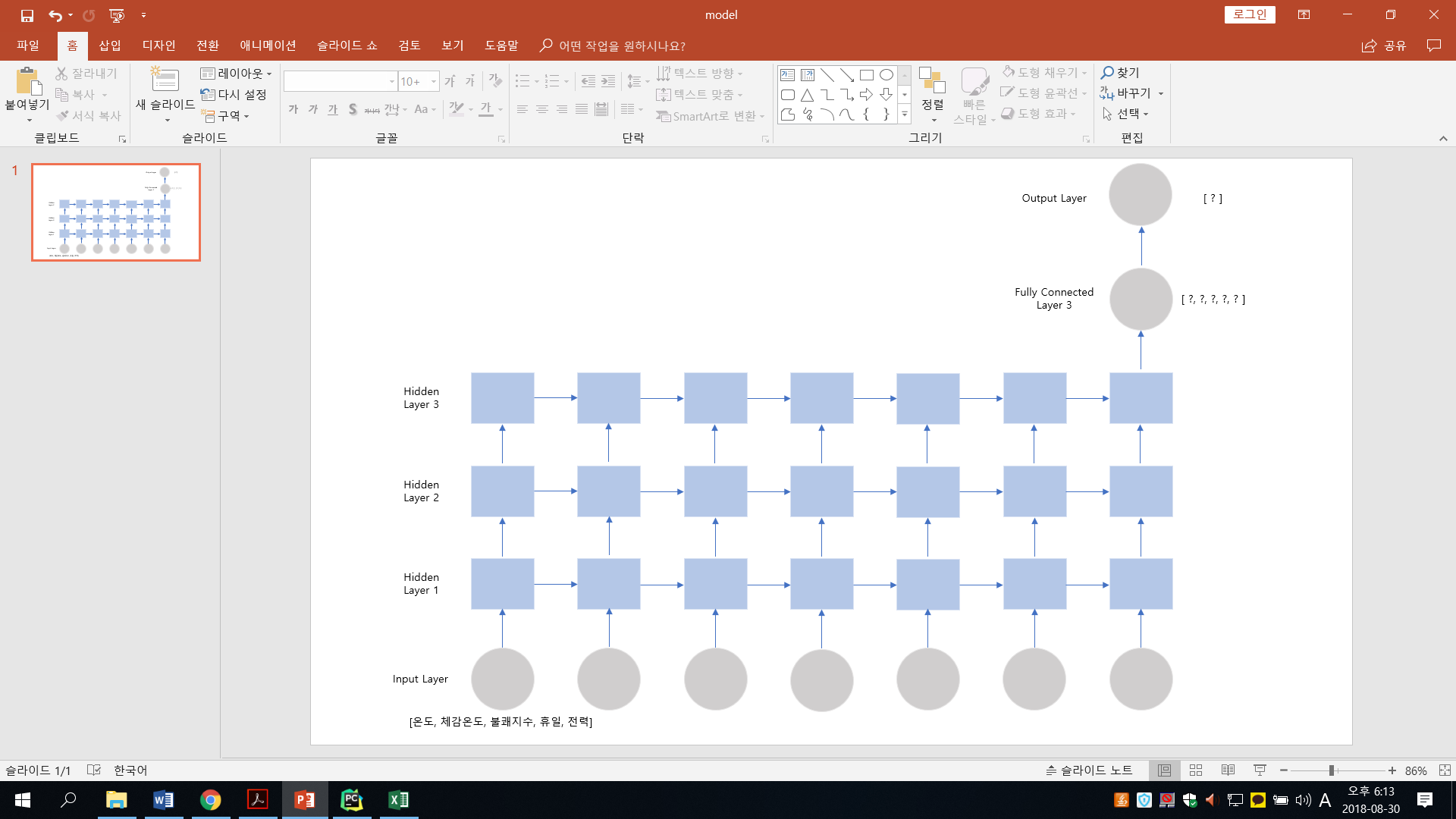
Loss(Cost)의 경우 RMSE(Root Mean Square Error)을 사용하여 예측 값과 실제 값 사이의 오차를 구하였다. 최적화 알고리즘은 오차 감소속도가 빠른 Adam Optimizer 알고리즘을 사용하여 learning rate는 0.001, 반복횟수인 epoch는 5000번으로 하였다.

2014년 ~ 2017년 하계 데이터를 Train set으로 2018년 하계(7월 ~ 8월 23일) 데이터를 Test set으로 사용하였다.

최종 Test set의 전력 수요 예측의 정확성은 아래의 식에 해당하는 백분율 오차를 사용하여 평균 오차율을 구하였고 최대 오차 값이 최소가 되도록 해야 잉여 생산 전력도 적으므로 최대 오차율도 결과에 포함하였다.

|  |  |
| --- | --- |
|  | (6) |

그림 1. 전력 수요 예측 모델



**3.3** 결과  
 위의 모델에 Test set인 2018년 하계 데이터를 넣어 수행시켜본 결과 표 4와 같았다. 평균 오차는 3.0886%로 ReLU함수를 이용한 RNN 구조가 가장 낮았고 최대 오차는 ReLU 함수를 이용한 GRU 구조가 가장 낮았다. 전력 수요 예측 시 평균 오차가 비슷할 때 최대 오차가 더 적은 것이 더 정확한 모델이므로 ReLU함수를 이용한 GRU 구조가 전력 수요 예측 모델로 가장 적합하다고 볼 수 있다. 그림 2는 GRU(ReLU)의 수행결과로 예측 값(Predict)과 실제 값(Truth)을 도시하였다.

표 4. 수행 결과

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | RNN(ReLU) | LSTM(tanh) | LSTM(ReLU) | GRU(tanh) | GRU(ReLU) |
| Avg RMSE | 0.11126 | 0.11819 | 0.16136 | 0.12331 | 0.11292 |
| Avg. Error (%) | 3.0886 | 3.9170 | 5.0148 | 3.7560 | 3.3445 |
| Max Error (%) | 13.9309 | 10.7043 | 13.7022 | 12.7022 | 4.8823 |

그림 2. GRU(ReLU) 수행 결과

**Ⅳ. 결론**

본 연구는 먼저 전력 소비에 영향을 미치는 요소를 추측하고 이후 예측 값의 오차율을 감소시키기 위해 Excel의 상관분석 도구를 이용하여 상관관계 있는 요소만을 추려냈다. 추려진 요소로는 온도, 체감온도, 불쾌지수, 휴일이 있다. 이 때 기후데이터는 한 지점에서의 값을 대표 값으로 사용하기에 힘들기 때문에 인구수 순으로 상위 6개 지점(서울, 부산, 인천, 대구, 대전, 광주)의 값을 인구수 비율로 가중치를 곱하여 그것들의 합을 대표 값으로 사용하였다. 이 예측요소들은 RNN, LSTM, GRU 구조를 사용한 신경망 모델의 입력 값으로 사용되었다. 모델의 학습결과 활성화함수를 ReLU함수로 사용한 GRU 구조가 가장 좋은 결과를 도출하였고 이 때 평균 오차는 3.34%, 최대 오차는 4.88%정도였다. 일반적으로 여름 철(7 ~ 9월) 공급 예비율이 평균 20%인 점에서 이 모델을 통해 예측한다면 잉여 전력 생산량을 줄일 수 있어 경제적으로 많은 이익이 있을 것이라 생각한다.

**Ⅴ. 참고문헌**

[1] Dong-Ha Shin, Kwang-Ho Choi, and Chang-Bok Kim, Deep Learning Model for Prediction Rate Improvement of Stock Price Using RNN and LSTM, The Journal of Korean Institute of Information Technology, Vol 15, No. 10, pp. 9-16, Oct. 2017.

[2] Dong-Ha Shin, Chang-Bok Kim, A Study on Deep Learning Input Pattern for Summer Power Demand Prediction, The Journal of Korean Institute of Information Technology, Vol 14, No. 11, pp. 127-134, Nov. 2016.

[3] Daehee Kim, Bangbyung Gwan, Hyungkeun Ahn, Neungsoo Park, Designing Solar Power Generation Forecasting Algorithm using RNN, The Information and Control Symposium, pp. 156-157, Oct. 2017.