머신러닝을 통한 웹 기반 시뮬레이션 결과 예측

김지수*, 강민규**, 권훈**, 이정철***
*제주대학교 전산통계학과
**제주대학교 컴퓨터공학과
***한국과학기술정보연구원
e-mail:jclee@kisti.re.kr

Prediction on Web-based simulation result through Machine learning*

JiSu Kim*, MinKyu Kang**, Hoon Kwon**, JeongCheol Lee***
*Dept of Computer Science & Statistics, Jeju University
**Dept of Computer Engineering, Jeju University
***Korea Institute of Science and Technology Information

요 약

최근 IT 기술의 발전으로 웹 기반 시뮬레이션이 많은 연구에 도움을 주고 있다. EDISON은 그러한 시뮬레이션 환경을 제공해주는 플랫폼으로 전산열유체, 나노물리, 계산화학, 등 다양한 전문분야의 앱(이하 솔버)을 제공해준다. 이러한 솔버는 사용자가 그 솔버에 맞는 간단한 파라미터들만 입력하면 다양한 결과를 알아서 계산해 주는 편의를 제공해 주지만 입력 데이터에 따라 작업 시간이 상당히 혹은 무한히 걸릴 수 있기 때문에 언제 끝날지 모르는 작업의 완료 여부를 수시로 확인해야만 하는 불편함이 있다. 때문에 그 시간을 예측할 수 있다면 수시로 확인하는 불편함을 줄일 수 있다. 또한 오랜 시간이 걸리는 작업의 결과를 미리 알 수 있으면 사용자들에게 큰 도움이 될 것이다. 이런 점에서 본 논문에서는 시뮬레이션의 작업 결과와 수행 시간의 예측 모델을 적용해 보았다. 본 논문에서는 계산화학 분야의 uChem 솔버의 결과 예측을 진행하였는데 uChem 솔버는 1주기 및 2주기 원자들로 이루어진 화합물의 최적화된 상태의 에너지 값과 구조를 보여주는 프로그램이다. 예측을 진행한 결과 에너지는 99%이상의 상당히 높은 정확도를 얻을 수 있었고 수행 시간의 경우 약 90%의 정확도를 얻었다. 이를 통해서 사용자로 하여금 더욱 편리한 서비스를 제공할 수 있을 것이다.

1. 서론

IT 기술이 발달하면서 최근 많은 복잡한 문제들을 해결하기 위해 웹 기반 시뮬레이션이 유용하게 쓰이고 있다. EDISON은 온라인 플랫폼으로 다양한 분야의 솔버를 제공해준다. uChem은 계산화학 분야의 시뮬레이션으로 1, 2주기의 원자들로 이루어진 화합물의 분자 에너지 계산과 최적화 구조를 보여주는 프로그램이다.

작업을 수행하기 위해서는 솔버에 맞는 파라미터들을 입력해야 하는데 uChem 솔버에서는 화합물의 총 원자 개수, 각 원자의 원자번호와 3차원 공간상의 좌표 값을 입력받는다. 그러면 각 원자들은 화합물의 최적화된 구조에 맞게 이동하면서 그 과정에서의 좌표 값과 에너지 값이 로그에 기록된다. 간단한 파라미터들만 입력하면 알아서 최적화를 진행하며 에너지 값을 계산해주는 편리함을 제공하나 입력 데이터에 따라서 수행 시간의 편차가 크기 때문에 사용자들은 수시로 작업의 진행 상태를 확인하여야 한다. 또한 잘못된 입력 데이터로 인해서 작업이 끝나지 않고 무한정으로 돌아가는 상황이 발생할 수 있다. 만약 이러한 시뮬레이션의 수행 시간을 미리 알 수 있다면 수시로 확인하는 불편함을 줄일 수 있을 것이다. 또한 오랜 시간이 걸리는 작업의 결과값을 미리 알 수 있다면 많은 시간을 절약할 수 있을 것이다.

본 논문에서는 지금까지 uChem 솔버 사용자들의 시뮬레이션 결과 데이터를 추출, 분석하여 첫 번째로는 최종 분자 에너지 예측, 두 번째로는 작업 수행 시간 예측 모델을 적용해 보았다.

2. 분자 에너지 예측

먼저 예측을 진행하기 위한 첫 단계로 분석을 진행할 데이터를 추출하는 작업이 필요했다. uChem 솔버는 시뮬레이션 작업의 결과가 로그파일에 기록되는데 이 로그파일의 경로를 csv파일로 저장했다. 하지만 모든 작업이 성공적으로 끝난 것은 아니다. 잘못된 파라미터 값으로 인해서 작업이 실패한 것과 현재 진행중인 작업들도 있기 때문에 그 중 작업이 성공적으로 끝난 파일만 추출했다. 그리고 로그파일을 읽어와서 정규식을 이용해 목표변수인 분자 에너지 값을 추출했다.

좋은 예측 모델을 만들기 위해서는 예측을 위해서 어떤

* “본 연구는 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학지원사업의 연구결과로 수행되었음”(2018-0-01863)

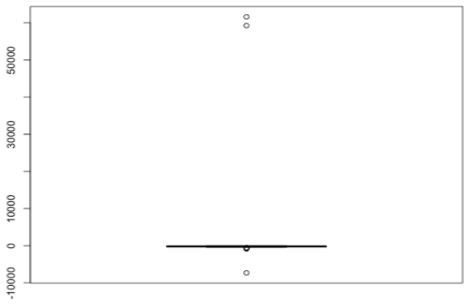
변수를 설명변수로 가져올지 정하는 것이 상당히 중요하다. 분자에너지는 그 분자가 어떠한 원소들로 구성되어있는지가 큰 영향을 미친다. 그래서 화합물을 구성하는 각 원소들의 개수(number of each atom)와 총 원자 개수(number of total atom)를 설명변수로 가져왔다.

그렇게 Y를 분자 에너지 값, X를 각 원소들의 개수와 총 원자 개수로 두어서 R 프로그램을 이용하여 간단한 회귀분석을 진행하였고 결과는 아래의 (그림 1)과 같이 어떠한 변수도 유의하지 않게 나왔다. 문제를 찾기 위해서 에너지 값의 분포를 Box Plot을 통해 살펴보았는데, 몇 개의 이상점이 존재한다는 것을 알 수 있었다. 아래의 (그림 2)를 보면 box의 형태가 하나의 직선으로 보일 정도로 범위를 크게 벗어난 이상점들이 존재한다는 것을 알 수 있다. 추정치는 이상점에 상당히 민감해서 이상점이 포함된 자료의 분석결과는 큰 영향을 받을 수 있기 때문에 반드시 전처리가 필요하다. (그림 3)은 1,3 사분위수에서 IQR(Inter Quartile Range)의 1.5배 만큼보다 더 떨어진 값들을 이상점으로 판단하고 제거한 후의 Box Plot이다. 제거를 하고 나니 정상적인 box의 형태를 갖추었고 이를 바탕으로 다시 회귀분석을 진행 하였다.

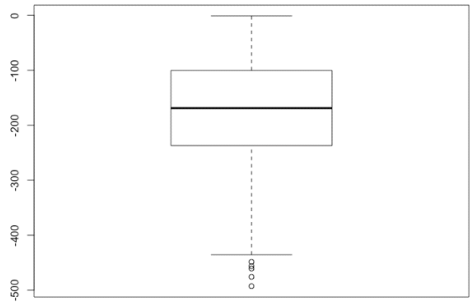
설명변수는 위와 동일하게 하여 전체 데이터를 8:2 비율로 train data와 test data로 나누었고, train data만으로 회귀분석을 진행하여 나온 선형 모델에 test data를 적용해 보았다. (그림 4)는 진행 결과 예측한 값이 실제 값과 얼마나 유사한지 알아보기 위해 plot을 찍어 가시화한 그림이다.

```
Coefficients: (1 not defined because of singularities)
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -642.48     974.31  -0.659    0.511
natom         257.01    2931.22   0.088    0.930
H            -422.17    2926.07  -0.144    0.885
He             62.80    4309.61   0.015    0.988
Li             88.94    3273.25   0.027    0.978
Be           -439.74    3361.11  -0.131    0.896
B             265.58    3064.72   0.087    0.931
C              26.70    2922.54   0.009    0.993
N             555.35    2934.32   0.189    0.850
O             763.17    2919.62   0.261    0.794
F            -335.80    2916.75  -0.115    0.908
Ne                NA         NA      NA       NA
```

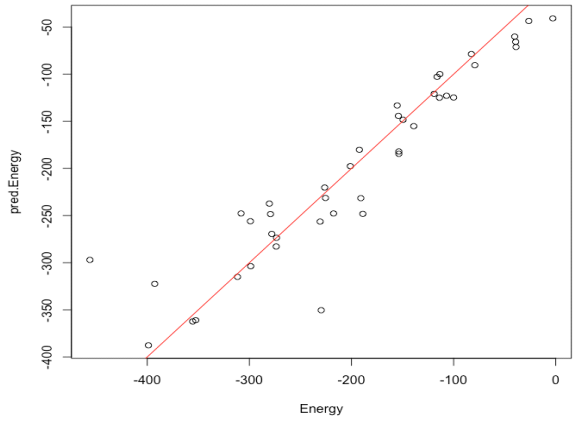
(그림 1) 이상점 제거 전 회귀분석 결과



(그림 2) 이상점 제거 전 Box Plot



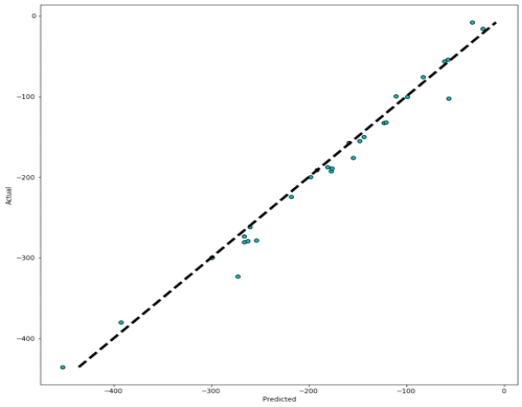
(그림 3) 이상점 제거 후 Box Plot



(그림 4) Actual / Predict Plot

결과를 보면 전체적으로 꽤나 잘 예측하지만 몇 개의 튀는 점들은 꽤나 큰 오차를 보여준다. -300, -400에 가까워질수록 실제 에너지 값과 예측한 에너지 값의 차이가 점점 커지는 것을 볼 수 있다. 이는 -적으로 클수록 데이터가 충분하지 않아서 제대로 된 학습이 이루어지지 않았기 때문이다. 때문에 더욱 높은 정확도를 얻기 위해서 데이터를 추가로 생성하는 작업이 필요하다. 그러기 위해선 uChem 솔버에 맞는 대량의 입력 데이터셋이 필요한데 이를 POSCAR 파일을 변환하여 해결하였다. POSCAR 파일이란 격자 정보와 이온들의 위치를 포함한 여러 화학정보를 담고잇는 파일로 많은 연구에 쓰이고 있다. 이 파일을 파이썬 프로그램에서 ase 라이브러리를 사용해 xyz 파일로 변환하였고, 간단한 코드를 작성해서 xyz 파일을 uChem 솔버에 맞는 입력 데이터 형태로 변환시켰다. 이 데이터를 시뮬레이션 돌리는데 앞서 말했듯이 하나하나 전부 작업을 수행하려면 상당히 오랜 시간이 걸리기 때문에 클러스터 컴퓨팅 자원을 이용했다. 이는 한 번에 수십 개의 시뮬레이션을 돌릴 수 있게 해주는 자원으로 비교적 짧은 시간에 작업을 수행할 수 있었다. 그리고 전과 동일한 방법으로 작업 수행 결과를 받아와서 기존 데이터와 병합하였다. 그리고 다시 회귀분석을 진행하기에 앞서 설명변수 몇 개를 추가했다. 기존에는 각 원소들의 개수와 총 원자 개수를 설명변수로 예측을 진행하였는데, 이에 추

가적으로 분자량(molecular weight)과 총 양성자 수(number of total proton)를 설명변수로 가져왔다. 분자량은 질량 이라고도 하는데 탄소의 원자량 12를 기준으로 각 원소마다 고유한 값이 있다. 여기서는 화합물을 이루는 원자들의 질량을 모두 더해서 변수로 가져왔고 원자번호와 양성자 수는 같기 때문에 원자번호를 모두 더해서 양성자 수라는 변수로 가져왔다. 그리고 랜덤 포레스트와 딥 러닝 알고리즘을 사용해서 예측 모델을 적용해 보았는데 랜덤 포레스트는 약 90%, 딥 러닝의 경우는 약 97% 정도의 높은 정확도를 보여주었다. 아래의 (그림 5)은 그 중 딥 러닝을 진행한 결과로 정확도와 오차값을 담고 있다. 이는 실제 에너지 값과 예측한 에너지 값이 상당히 비슷하다는 것을 알 수 있다.

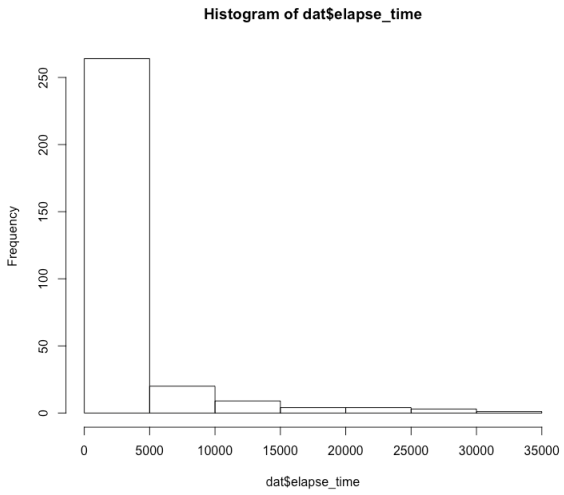


*** Mean Absolute Error = 11.70049967869303
*** Mean Absolute Percentage Error = 17.653103844802484
*** Root Mean Squared Error = 16.654557216713776
*** Root Mean Squared Log Error = nan
*** Coefficient of Determination (R Squared) = 0.9742546615220475

(그림 5) 에너지 예측 딥 러닝 결과

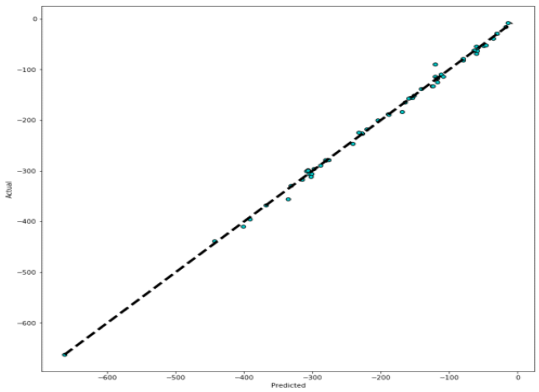
3. 작업 수행 시간 예측

다음으로는 시뮬레이션 작업 수행 시간을 예측해 보았다. 먼저 목표 변수인 수행 시간을 받아오는 작업이 필요했다. 로그파일을 보면 작업 시작 시간과 종료 시간이 기록되어 있는데 이를 정규식으로 추출해서 둘의 차이를 초 단위로 바꾸어 수행 시간이라는 목표 변수로 가져왔다. 그리고 잘못된 입력 데이터로 인해 비정상적으로 오래 걸리는 작업과 무한정으로 돌아가는 작업은 이상점으로 제외했다. 먼저 수행 시간의 분포가 알기 위해서 히스토그램을 그려보았는데 대부분의 작업은 5000초 이하로 끝나지만 오래 걸리는 경우에는 30000초 이상 걸리기도 한다는 것을 알 수 있다. 이렇게 편차가 큰 수행시간이기 때문에 언제 끝날지도 모르고 수시로 작업의 결과를 확인한다는 것은 여간 불편한 일이 아니다. 이런 점에서 수행 시간을 예측한다면 시뮬레이션 이용자들은 더욱 편리하게 솔버를 사용할 수 있을 것이다.



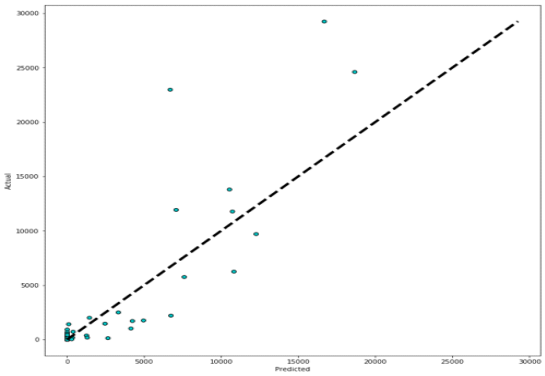
(그림 6) 작업 수행 시간 히스토그램

작업 수행 시간을 예측하기 위한 변수로 처음에는 초기 에너지 값과 최적화가 끝난 후 최종 에너지 값(앞서 예측한 값)의 차이가 수행 시간에 영향을 줄 것이라 생각해서 gap_energy 라는 변수로 그 차이를 받아왔다. 하지만 예측의 결과가 그리 좋지 못하였고 이유를 찾아보니 energy가 일정한 방향으로 증가하거나 감소하지 않는 다는 사실을 알게 되었다. 초기에너지와 최종에너지가 비슷하더라도 그 중간 과정에서 수십 번 증가, 감소가 일어나서 작업의 시간이 오래 걸릴 수 있기 때문에 에너지의 차이는 좋은 변수가 되지 못했다. 다음으로 uChem 솔버는 임의로 입력한 원소들의 위치를 안정화 된 상태로 만들어주는 프로그램이기 때문에 초기 원소들 사이의 거리가 안정화 되었을 때의 거리와 얼마나 차이가 있는지가 작업 수행 시간에 큰 영향을 미칠 것이라 생각했다. 초기 원자간 거리는 쉽게 구할 수 있으나 안정화 되었을 때 거리를 아는 것이 중요했다. 화합물을 구성하는 동종 원자의 경우 공유 결합 반지름이 존재하는데, 이를 두배 한 값을 공유 결합 거리라 한다. 이 거리를 안정하 된 거리라 두고 초기 원자간 거리와 얼마나 차이 나는지 그 값을 변수로 가져왔다. 그리고 에너지 예측할 때 사용했던 변수들과 추가로 부피와 질량, 원자들의 중심점과의 거리 합을 설명변수로 두고 예측 모델을 적용해 보았는데 랜덤 포레스트와 딥 러닝 알고리즘 모두 약 50%정도 밖에 안되는 정확도를 보여주었다. 예측의 정확도를 높이기 위해 로그 파일을 살펴보던 중 uChem 솔버는 원자들의 구조를 최적화된 구조로 바꾸기 위해서 수차례의 작업이 진행 되는데 최대 999번까지 작업을 진행하고 999번으로도 구조를 찾지 못하는 경우 그 상태로 프로그램을 종료한다는 사실을 알게 되었다. 그래서 이 데이터들은 정상적으로 마쳐진 데이터가 아니기 때문에 데이터셋에서 제외시켰다. 그러고 난 후 다시 예측 모델을 적용해 본 결과 에너지 예측과 작업 수행 시간 예측 모두 보다 훨씬 좋은 결과를 얻을 수 있었다.



*** Mean Absolute Error = 4.329199566517915
*** Mean Absolute Percentage Error = 4.711327008098339
*** Root Mean Squared Error = 6.6227216812192635
*** Root Mean Squared Log Error = nan
*** Coefficient of Determination (R Squared) = 0.9973271424454961

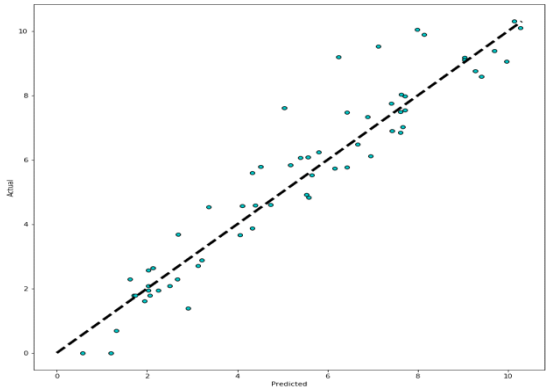
(그림 7) 에너지 예측 딥 러닝 결과-2



*** Mean Absolute Error = 1312.6379467463885
*** Mean Absolute Percentage Error = 122.56678844941531
*** Root Mean Squared Error = 3104.0129693988206
*** Root Mean Squared Log Error = 1.8624232349825236
*** Coefficient of Determination (R Squared) = 0.7391961756876624

(그림 8) 수행 시간 예측 딥 러닝 결과

위의 [그림 7]과 [그림 8]을 보면 데이터를 제거 후, 에너지 예측과 작업 수행 시간 예측 모두 정확도가 커진 것을 알 수 있다. 특히 에너지 예측에 경우 정확도가 99%가 넘고 거의 100%에 가까운 높은 수치를 보여주고 낮은 오차로 좋은 결과를 얻었다. 이로 인해서 사용자들이 오랜 시간을 기다리지 않고 충분히 정확한 에너지 값을 가져올 수 있을 것이다. 그리고 작업 수행 시간 예측의 경우에도 약 74%로 아직 그리 높은 정확도를 보여주지는 못하지만 이 역시 전에 비해서 많이 높아진 것을 알 수 있다. 하지만 수행 시간의 편차가 너무 커서 대부분의 데이터가 한 점에 모여 있고 소수의 데이터는 그 값이 너무 커서 많이 분산 되어있다. 그러한 값들은 예측을 하는 데 큰 영향을 미치기 때문에 데이터 값의 폭을 줄이는 작업이 필요했다. 이를 위해서 목표 데이터인 작업 수행 시간에 log를 취하고 딥 러닝 알고리즘을 다시 적용해 본 결과 값들이 한 곳에 몰려있지 않고 고루 퍼져있으며 약 90%에 달하는 높은 정확도를 보여주었다.



*** Mean Absolute Error = 0.65736949232923
*** Mean Absolute Percentage Error = inf
*** Root Mean Squared Error = 0.9011474517897661
*** Root Mean Squared Log Error = 0.18547341490779384
*** Coefficient of Determination (R Squared) = 0.9021960782582953

(그림 9) 수행 시간 예측 딥 러닝 결과-2

4. 결론 및 향후연구

먼저 분자 에너지 값을 예측한 결과 99%가 넘는 정확도로 상당히 잘 예측한다. 이로 인해서 솔버 사용자에게 상당히 오랜 시간이 걸리는 작업의 결과를 기다리지 않아도 거의 정확한 값을 예측할 수 있는 편의를 제공할 수 있다. 그리고 작업 수행 시간 예측의 정확도는 약 74%정도이고 로그변환을 통해서 약 90%의 정확도를 얻을 수 있었다. (그림 8)에서 볼 수 있듯이 대부분의 작업들이 금방 끝나 0에 가까울수록 데이터가 많지만 상당히 오랜 시간이 걸리는 작업의 경우 데이터양이 매우 적어서 충분한 학습을 시키기 어렵다. 따라서 학습이 잘 이루어질 수 있도록 충분한 데이터가 있다면 더욱 정확한 예측을 할 수 있을 것이다. 또한 수행 시간에 영향을 미치는 다른 설명변수를 찾는다면 현재 변수만으로 충분한 설명을 하지 못하는 부분까지도 다룰 수 있을 것이다.

참고문헌

[1] UChem4.0.2 Manual 26 Sept. 2018 Cheol Ho Choi KyungpookNational University
[2] Jinseung Yu;Hangi Kim;Soonwook Hwang;Jae-Cheol Ryou “Simulation Data Learning Model for Prediction of Computing Resources in Web-based Simulation Environment”
[3] https://wiki.fysik.dtu.dk/ase/ase/io/io.html
[4] http://openbabel.org/wiki/XYZ_(format)
[5] http://www.xcrysden.org/doc/XSF.html