최소한의 함수 산출로 학습한 신경망에 의한 최적화

한철호[O1], 곽하녹[1], 윤상웅[2], 심문보[3], 장병탁[1,2]
[1]서울대학교 컴퓨터공학부
[2]서울대학교 협동과정 뇌과학전공
[3]삼성전자 종합기술원
{chhan, hnkwak, swyoon}@bi.snu.ac.kr, munbo.shim@samsung.com, btzhang@bi.snu.ac.kr

Optimization by Neural Networks Learned from Minimum Function Evaluations

Cheolho Han[O1], Hanock Kwak[1], Sangwoong Yoon[2], Munbo Shim[3], Byoung-Tak Zhang[1,2]
[1]School of Computer Science & Engineering, Seoul National University
[2]Interdisciplinary Program in Neuroscience, Seoul National University
[3]Samsung Advanced Institute of Technology, Samsung Electronics Co., Ltd.

요 약

이 논문은 주어진 화학 물질 사이에서 어떤 특성이 최댓값을 갖는 물질을 탐색하기 위해 신경망(neural network)과 베이지안 최적화(Bayesian optimization)를 결합하여 화학물질 데이터셋인 QM7b 내의 물질 중에서 최대 밴드 갭(HOMO-LUMO gap)을 갖는 물질을 탐색하였다. 신경망은 함수를 추정할 수 있는 모델이며, 베이지안 최적화는 최소한의 함수 산출로 최적화를 하는 알고리즘이다. 신경망에 rectified linear unit (ReLU) 및 드롭아웃(dropout)을 적용하였고, 신경망에 ReLU를 사용하는 것이 알아야하는 밴드 갭의 수를 크게 낮추는 것을 확인하였다.

1. 서 론

주어진 화학 물질 사이에서 어떤 특성이 최댓값을 갖는 혹은 특정 값에 가까운 물질을 탐색하기 위해 직접 실험하고 측정하는 대신 양자 계산을 이용할 수 있다. 그러나 이러한 양자 계산 또한 시간 및 비용이 크게 들어, 어떤 물질의 특성을 아는데 비용이 많이 든다. 따라서 특성을 예측하기 위해 신경망을 이용하는 시도가 있었다[1-2]. 그러나 이 또한 상당량의 물질들의 특성을 양자 계산 등을 통해 알고 있어야만, 다른 물질들의 특성을 예측할 수 있다. 따라서 최소한의 특성만으로 원하는 물질을 탐색할 필요가 있다. 이것을 가능하게 하는 알고리즘으로 베이지안 최적화(Bayesian optimization)가 있다. 베이지안 최적화는 최소한의 함숫값만으로 최적화 문제를 풀도록 새로운 데이터를 선택하는 알고리즘이다[3]. 베이지안 최적화는 주로 가우스 과정(Gaussian process)와 결합하여 사용하는데, 함수를 추정할 수 있는 모델인 신경망과도 결합할 수 있다[4].

이 논문은 먼저 화학 물질의 구조와 특성을 포함하는 데이터셋과 그 표현 방법에 대해 논의한다. 그리고 신경망과 베이지안 최적화, 이 둘의 결합 방법을 소개한다. 제시된 알고리즘을 적용한 실험의 내용 및 결과를 제시한 후, 결론으로 끝맺는다.

2. 화학 물질 특성 데이터셋

주어진 화학 물질 사이에서 어떤 특성이 최댓값을 갖는 물질을 탐색하기 위해 유기 분자 데이터셋인 QM7b[1-2]를 사용하였다. QM7b는 7211개의 분자 구조와 물리적 특성을 갖고 있다. 분자 구조는 쿨롱(Coulomb) 행렬의 형태로 주어지며, 물리적 특성은 최고준위 점유 분자궤도(highest occupied molecular orbital, HOMO), 최저준위 비점유 분자궤도(lowest unoccupied molecular orbital, LUMO), 분극률(polarizability), 원자화(atomization) 에너지 등이 포함되어 있다.

쿨롱 행렬은 하나의 분자를 하나의 행렬로 표현하며, 행렬의 (i, j) 성분은 다음과 같이 주어진다:
여기에서 Z는 원자의 핵전하량, R은 원자의 3차원 위치 벡터이다. 이와 같이 쿨롱 행렬은 분자의 구조를 간략히 나타낼 수 있다. 그리고 [1-2]에서 제시한 정렬 및 이진화를 적용하여, 이 행렬을 각 성분이 −1에서 1 사이인 고정된 길이의 벡터로 표현하였다.

$$c_{ij} = \begin{cases} 0.5Z_i^{2.4}, & \text{if } i = j \\ \dfrac{Z_i Z_j}{|R_i - R_j|}, & \text{if } i \neq j. \end{cases}$$

또한 실험에서는 물리적 특성으로서 GW 근사(GW approximation)를 통해 예측된 최고준위 점유 분자궤도와 최저준위 비점유 분자궤도의 차이인 밴드 갭(HOMO-LUMO gap)을 선택하여, 밴드 갭이 최대가 되는 물질을 탐색해 보았다.

3. 알고리즘

3.1 신경망(Neural Network)

신경망(neural network)은 오랜 역사를 지니고 있으며 최근에도 널리 사용되는 기계학습 모델이다. 신경망은 입력층, 은닉층, 출력층의 세 종류의 층을 가지며, 각 층은 여러 뉴런들로 이루어져 있다. 은닉층이 2-3개로 적을 경우 다층 퍼셉트론(multilayer perceptron, MLP), 은닉층이 4-30개 정도로 많을 경우 심층 신경망(deep neural network, DNN)이라고 부른다. 은닉층의 뉴런의 개수가 충분히 많으면, 신경망을 통해 임의의 함수를 임의의 정

확도로 근사할 수 있다는 것이 증명되어 있기 때문에[5], 유기 분자 데이터의 구조와 특성의 학습 외에도 여러 문제에 널리 적용할 수 있다.

은닉층이 여러 개인 심층 신경망은 모델의 복잡도 때문에 컴퓨터가 발달한 현대에 들어서야 활발히 쓰이고 있다. 여기에 rectified linear unit (ReLU)와 드롭아웃(dropout)을 적용하여 신경망의 성능을 더욱 높일 수 있다[6-11].

ReLU는 활성화 함수를 hyperbolic tangent, logistic function 대신 rectifier를 적용한 뉴런을 의미한다. Rectifier $f(x)$ 는

$$f(x) = \max(0, x)$$

과 같이 정의된다. ReLU를 적용함으로써 학습 과정에서 문제가 되던 vanishing gradient problem이 완화되었고[10], 감독 학습의 경우 초기 딥 러닝에서 쓰이던 사전 학습(pretraining) 과정도 필요가 없게 되었다[11].

드롭아웃은 학습 시 임의의 확률(0.5가 가장 널리 사용됨)로 각 뉴런의 출력을 강제로 0으로 설정하는 방법이다. 추론 시에는 다시 모든 뉴런의 출력을 사용하게 된다. 이때 학습과 추론 과정에서 활성화된 뉴런의 수가 다르므로, 출력의 크기를 상수를 곱하여 조절하게 된다. 드롭아웃은 뉴런들의 co-adaptation을 방지하여, 모델이 이미 주어진 데이터만을 잘 설명하고 새로운 데이터를 전혀 설명하지 못하는 현상인 overfitting을 피하는 데 도움을 준다.

3.2 베이지안 최적화(Bayesian Optimization)

베이지안 최적화는 출력 값과 출력 분산을 이용하여 출력이 클 가능성이 높은 데이터를 새로운 데이터로 쓴다[3-4]. 다시 말해, 획득(acquisition) 함수 중 하나인 개선 기댓값(expected improvement)이 큰 데이터를 선택하게 되는데, 이는 다음과 같이 계산된다.

먼저 현재 최대 출력 y_{CM}을 어떤 점 x_r에서의 평균 $\hat{y}$과 표준편차 $\sigma_{\hat{y}|x_r}$으로 정규화한다:

$$-z = \frac{y_{\mathrm{CM}} - \hat{y}}{\sigma_{\hat{y}|x_r}}.$$

표준정규분포의 누적분포함수와 확률 밀도 함수를 각각 $G(\cdot)$, $g(\cdot)$이라 하면, 개선 기댓값 a_{EI}는

$$a_{\mathrm{EI}} = \sigma_{\hat{y}|x_r}[zG(z) + g(z)]$$

가 된다.

베이지안 최적화는 a_{EI}가 큰 데이터를 순차적으로 선택함으로써 진행된다. 베이지안 최적화는 주로 가우스 과정(Gaussian process)과 결합하여 사용하지만, 신경망과도 결합하여 사용할 수 있다.

3.3 신경망에 의한 베이지안 최적화

위에서 보듯이 베이지안 최적화를 위해서는 어떤 점에서의 출력의 평균과 분산을 구해야 한다. 평균은 신경망의 출력 값을 이용하면 되지만, 출력 분산의 경우에는 단순하지 않다. 출력 분산은 다음과 같은 과정을 통해 계산된다.

먼저, 출력 민감도 $g(x)$는 데이터 x가 주어졌을 때, 출력 $\hat{y}$을 무게 w로 미분한 값

$$g(x) = \frac{\partial \hat{y}|x}{\partial w}$$

가 되고, 피셔(Fisher) 정보 행렬 A는

$$A = \frac{1}{E^2}\frac{\partial^2 E^2}{\partial w^2} \approx \frac{1}{E^2}\sum_{i=1}^{m} g(x_i)g(x_i)^{\mathrm{T}}$$

가 된다[12]. 여기에서 E는 비용 함수로서 평균 제곱 오차이다. 그리고 참조하고자 하는 어떤 입력 x_r에서의 출력 분산 $\sigma^2_{\hat{y}|x_r}$은

$$\sigma^2_{\hat{y}|x_r} \approx g(x_r)^{\mathrm{T}} A^{-1} g(x_r)$$

로 근사된다[13].

4. 실험 및 결과

앞서 설명한 신경망과 베이지안 최적화를 결합하여 QM7b 내의 물질 중에서 최대 밴드 갭을 갖는 물질을 탐색해 보았다. 이를 위해 먼저 QM7b 내의 전체 7211 개의 물질들을 정렬 및 이진화를 거쳐 2495 길이의 벡터로 변환하였다.

이후 개선 기댓값이 큰 데이터를 하나씩 선택하며, 지금까지 선택된 데이터를 배치(batch) 방식으로 신경망 모델을 학습하였다.

개선 기댓값을 계산하기 위해서는 피셔 정보 행렬의 역행렬을 계산해야 한다. 출력 민감도 벡터의 길이는 각 뉴런을 연결하는 간선(edge)의 수 N_{E}와 같다. 피셔 정보 행렬은 $N_{\mathrm{E}} \times N_{\mathrm{E}}$의 크기를 가지므로, 상당히 큰 행렬이 된다. 따라서 메모리가 부족할 경우, 모든 간선으로부터 출력 민감도를 계산하는 대신, 출력층과 연결된 간선 만으로부터 출력 민감도의 일부를 계산하여, 작은 크기의 피셔 정보 행렬을 얻을 수 있다. 이번 실험에서는 이렇게 작은 크기의 피셔 정보 행렬을 이용하였다. 또한 피셔 정보 행렬의 역행렬을 계산할 경우, 수치적인 불안정성으로 계산이 잘 되지 않을 경우가 발생하는데, 이때 행렬에 단위행렬의 (0.001과 같이 작은 크기의) 상수 배를 더해줌으로써 문제를 해결할 수 있다.

신경망 모델은 은닉층이 하나인 다층 퍼셉트론과 은닉층이 두 개인 다층 퍼셉트론을 사용하였다. 은닉층이 하나인 경우에는 은닉층의 뉴런의 수를 20개로 하였고, 은닉층이 두 개인 경우에는 은닉층의 뉴런의 수를 입력층에서 가까운 층에서부터 100개, 50개로 하였다. 또한 추가적으로 은닉층이 두 개인 경우에 ReLU를 적용하고, 여기에 더해 드롭아웃까지 적용해보았다. 드롭아웃을 할 때, 출력을 0으로 설정할 확률은 0.5로 하였다.

QM7b 내의 물질들이 갖는 밴드 갭 분포는 그림 1과 같다. 하나의 물질 만이 15 eV가 넘는 값을 가지며, 나머지는 14 eV 미만이고, 평균 10 eV 정도의 값을 가진다.

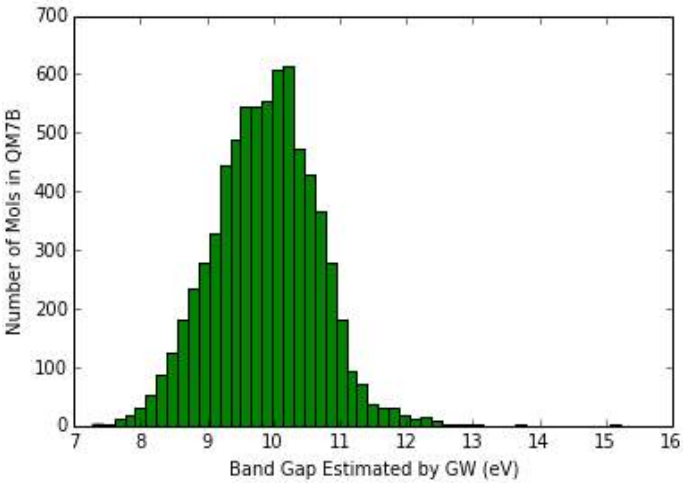

그림 1. QM7b 내의 물질들의 밴드 갭 분포

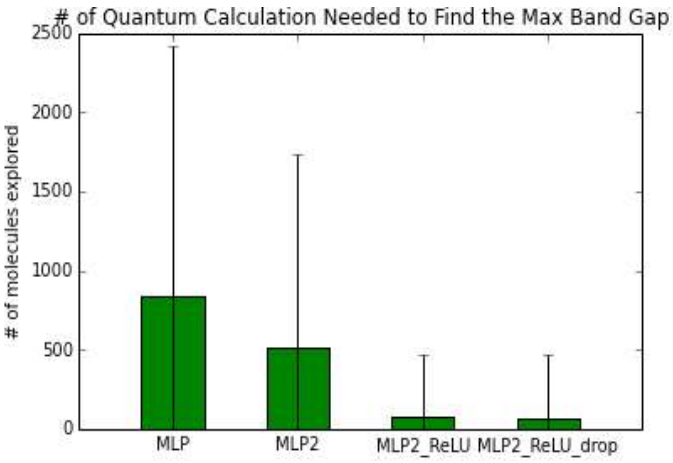

그림 2. 최대 밴드 갭을 찾기 위해 알아야하는 밴드 갭의 수.

앞서 설명한 과정을 거쳐, 각 모델에 대하여 QM7b 내에서 몇 개의 물질의 밴드 갭을 이용하여 최대 밴드 갭을 찾는지를 확인해 보았다(그림 2). 이때 각 모델 별로 100번 반복 수행하였으며, 편의상 1000개 이내에서 최대 밴드 갭을 찾지 못할 경우 4000으로 설정하였다. 총 100번 수행 중에 이런 경우는 은닉층이 하나인 신경망(MLP), 은닉층이 두 개인 신경망(MLP2), ReLU를 적용한 신경망(MLP2_ReLU), 여기에 드롭아웃을 더한 신경망(MLP2_ReLU_drop) 별로 각각 20, 11, 1, 1번 발생하였다. 평균적으로 MLP2_ReLU와 MLP2_ReLU_drop이 가장 적은 수의 밴드 갭을 알고 최대 밴드 갭을 찾아냈다. 그러나 평균값이 70-900 사이에 분포하는 반면, 표준편차가 390-1600 사이에 분포하여 평균에 비해 표준편차가 매우 크게 나왔다. 이는 알고리즘이 알아야 하는 밴드 갭 수를 (10개 미만으로도 상당 수 나타남) 큰 폭으로 줄일 수 있는 경우가 많지만, 안정적으로 줄이지는 못함을 보여준다.

5. 결론 및 향후 연구

이 논문은 주어진 화학 물질 사이에서 어떤 특성이 최댓값을 갖는 물질을 탐색하기 위해 최소한의 특성값 만을 이용하는 베이지안 최적화를 신경망에 적용하였다. 그 결과 상당수가 매우 적은 수의 특성값만으로도 최댓값을 찾아내었으나, 쉽사리 찾지 못하는 경우도 발생하였다. 향후 이러한 알고리즘의 불안정성을 개선할 수 있는 연구가 필요할 것으로 보인다.

감사의 말

이 논문은 삼성전자 종합기술원의 지원을 받아 수행된 연구이며, 정부(미래창조과학부 및 정보통신기술진흥센터)의 정보통신·방송 연구개발사업 지원(R0126-15-1072-SW 스타랩, 10035348-mLife, 10044009-HRI.MESSI)과 정부(미래창조과학부 및 한국연구재단)의 지원(NRF-2010-0017734-Videome)을 일부 받았음.

참고문헌

[1] G. Montavon, K. Hansen, S. Fazli, M. Rupp, F. Biegler, A. Ziehe, A. Tkatchenko, O. A. von Lilienfeld, K.-R. Müller, "Learning Invariant Representations of Molecules for Atomization Energy Prediction," Advances in Neural Information Processing Systems (NIPS), 2012

[2] G. Montavon, M. Rupp, V. Gobre, A. Vazquez-Mayagoitia, K. Hansen, A. Tkatchenko, K.-R. Müller, O.A. von Lilienfeld, "Machine Learning of Molecular Electronic Properties in Chemical Compound Space," New Journal of Physics, 2013.

[3] E. Brochu, V. M. Cora, and N. de Freitas, "A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning," http://arxiv.org/abs/1012.2599

[4] J. Snoek, O. Rippel, K. Swersky, R. Kiros, N. Satish, N. Sundaram, M. M. A. Patwary, Prabhat, R. P. Adams, "Scalable Bayesian Optimization Using Deep Neural Networks," http://arxiv.org/abs/1502.05700

[5] K. Hornik, "Approximation Capabilities of Multilayer Feedforward Networks," Neural Networks, vol. 4, no. 2, pp. 251–257, 1991.

[6] Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. R., "Improving neural networks by preventing co-adaptation of feature detectors," http://arxiv.org/abs/1207.0580

[7] N. Srivastava, "Improving Neural Networks with Dropout," Master's thesis, University of Toronto, January 2013.

[8] George E. Dahl, Tara N. Sainath, and Geoffrey E. Hinton, "Improving Deep Neural Networks for LVCSR Using Rectified Linear Units and Dropout," In ICASSP 2013.

[9] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," The Journal of Machine Learning Research, v.15 n.1, p.1929-1958, January 2014.

[10] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng, "Rectifier Nonlinearities Improve Neural Network Acoustic Models," in Proc. 30th International Conference on Machine Learning, Atlanta, Georgia, USA, 2013.

[11] Xavier Glorot, Antoine Bordes, and Yoshua Bengio, "Deep Sparse Rectifier Neural Networks," in Proc. 14th International Conference on Artifical Intelligence and Statistics, Fort Lauderdale, FL, USA, 2011.

[12] D. A. Cohn, "Neural Network Exploration Using Optimal Experiment Design," Neural Information Processing Systems 1994, Jun. 16, 1994.

[13] R. Thisted, "Elements of Statistical Computing," Chapman and Hall, NY, 1988.