Automated Feature Engineering using Deep Reinforcement Learning

Ziyu Xiang		Supervisor: Prof. Kalathil, Prof. Qian

Feature Engineering : Feature engineering is the process of taking a dataset and constructing explanatory variables — features — that can be used to train a machine learning model for a prediction problem.

Manual Feature Engineering (MFE): The traditional approach to feature engineering is to build features one at a time using domain knowledge, a tedious, time-consuming, and error-prone process known as manual feature engineering.

Automated Feature Engineering (AFE): Automated feature engineering improves upon MFE standard workflow by automatically extracting useful and meaningful features from a set of related data tables with a framework that can be applied to any problem

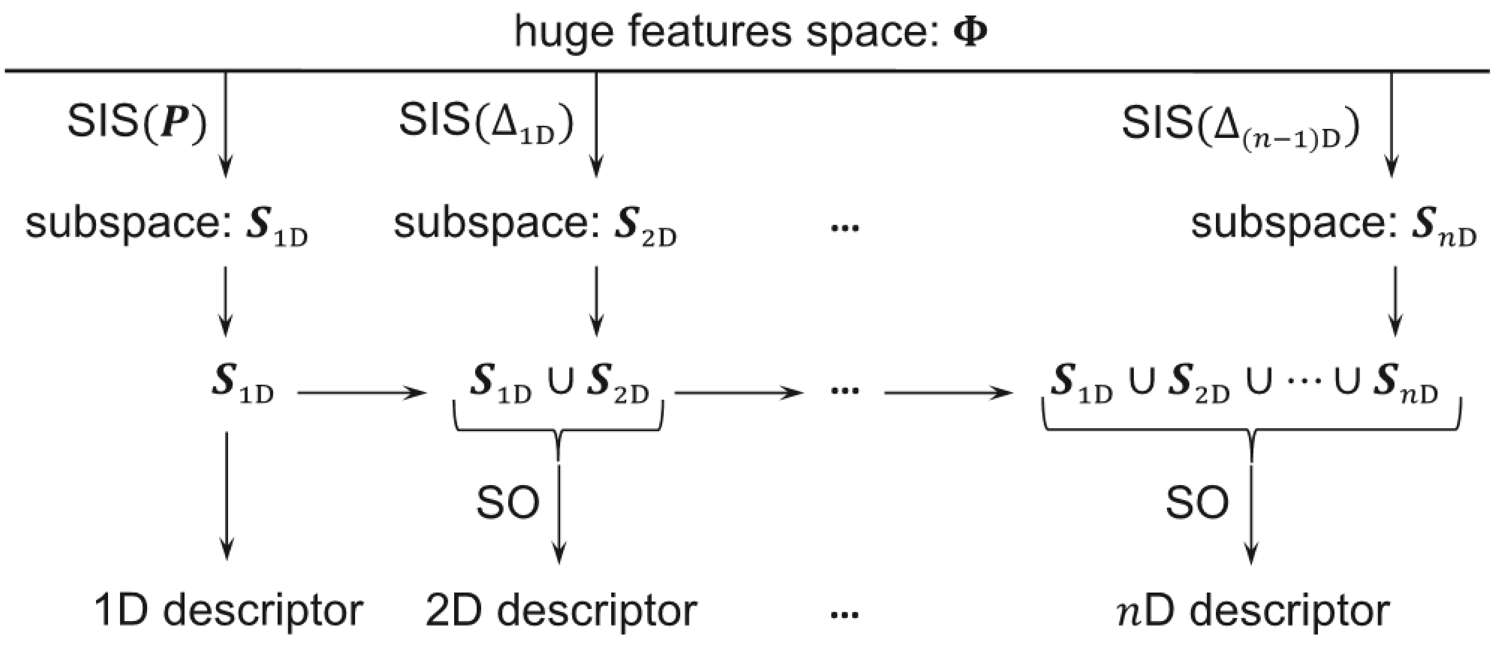
- creates interpretable features
- generally applicable

Ref: https://towardsdatascience.com/why-automated-feature-engineering-will-change-the-way-you-do-machine-learning-5c15bf188b96

The Sure Independent Screening and Sparse Operation(SISSO)

prototypes	#materials	primary features	descriptor	classification accuracy
NaCl	132	IE_A, IE_B, χ_A, χ_B, r_{covA}, r_{covB}, EA_A, EA_B, v_A, v_B, d_{AB}	$d_1 := \frac{IE_A IE_B (d_{AB} - r_{\mathrm{covA}})}{\exp(\chi_A)\sqrt{r_{\mathrm{covB}}}}$	100%

$$\hat{H}^{(\mathrm{m})} \equiv \{I, +, -, \times, /, \exp, \log, |-|, \sqrt{\ }, ^{-1}, ^{2}, ^{3}\}[\phi_1, \phi_2],$$



Ref: Ouyang, Runhai, et al. "SISSO: A compressed-sensing method for identifying the best low-dimensional descriptor in an immensity of offered candidates." *Physical Review Materials* 2.8 (2018): 083802.

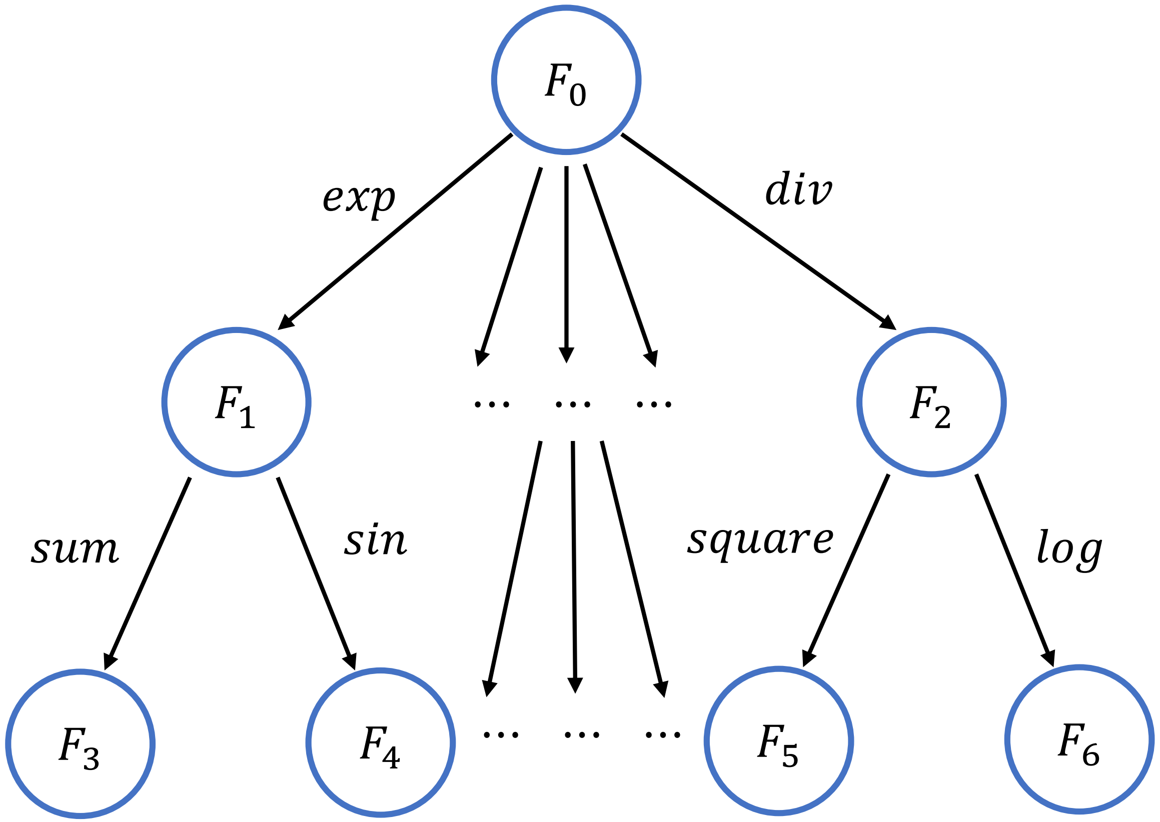
Automated Feature Engineering using Deep Reinforcement Learning

Tree search problem using Deep Q Learning

Pros: efficient learning

less computation resources requirement

Cons: No guarantee to contract to global optimal

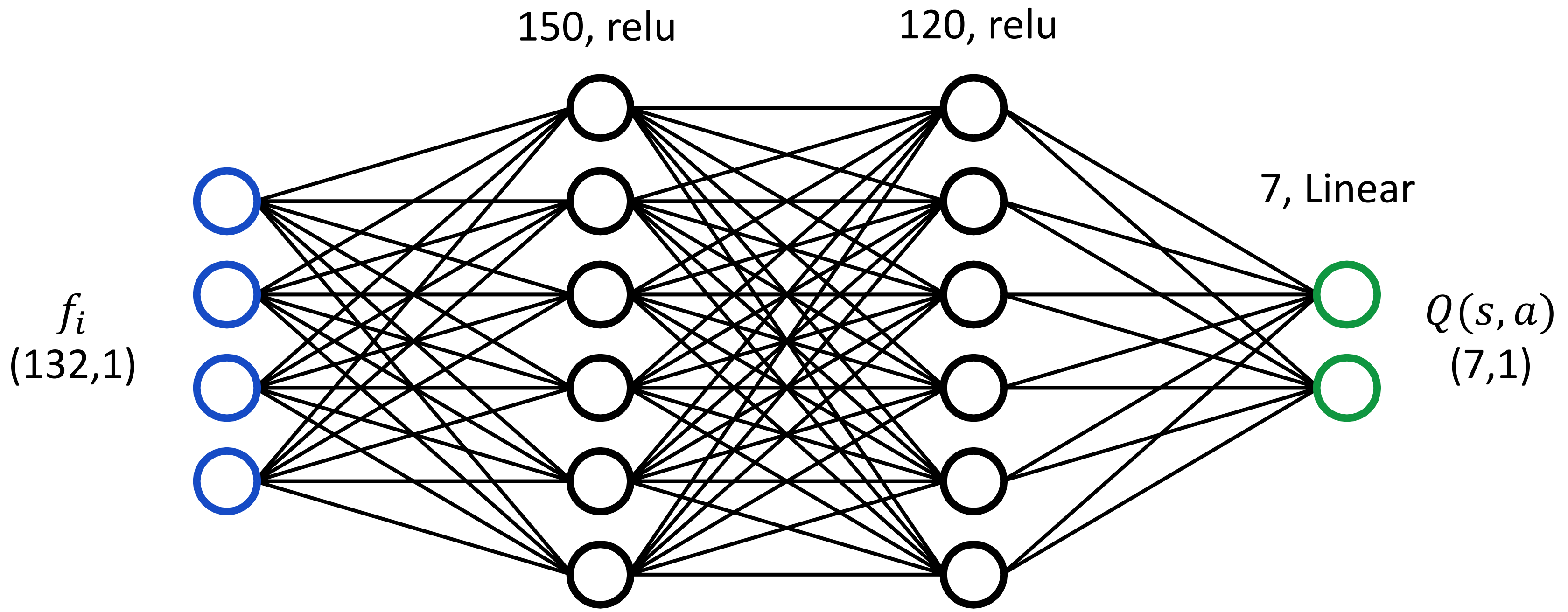


Dataset:

Primary features' dimension: (132,7)
Labels: 2

prototypes	#materials	primary features	descriptor	classification accuracy
NaCl	132	$IE_A, IE_B, \chi_A, \chi_B, r_{\mathrm{covA}}, r_{\mathrm{covB}}, EA_A, EA_B, v_A, v_B, d_{AB}$	$d_1 := \frac{IE_A IE_B (d_{AB} - r_{\mathrm{covA}})}{\exp(\chi_A)\sqrt{r_{\mathrm{covB}}}}$	100%

Algorithm: Deep Q-Learning



States: each feature f_i

Rewards: classification accuracy a, computed by Logistic Regression.

Actions: operation in the set

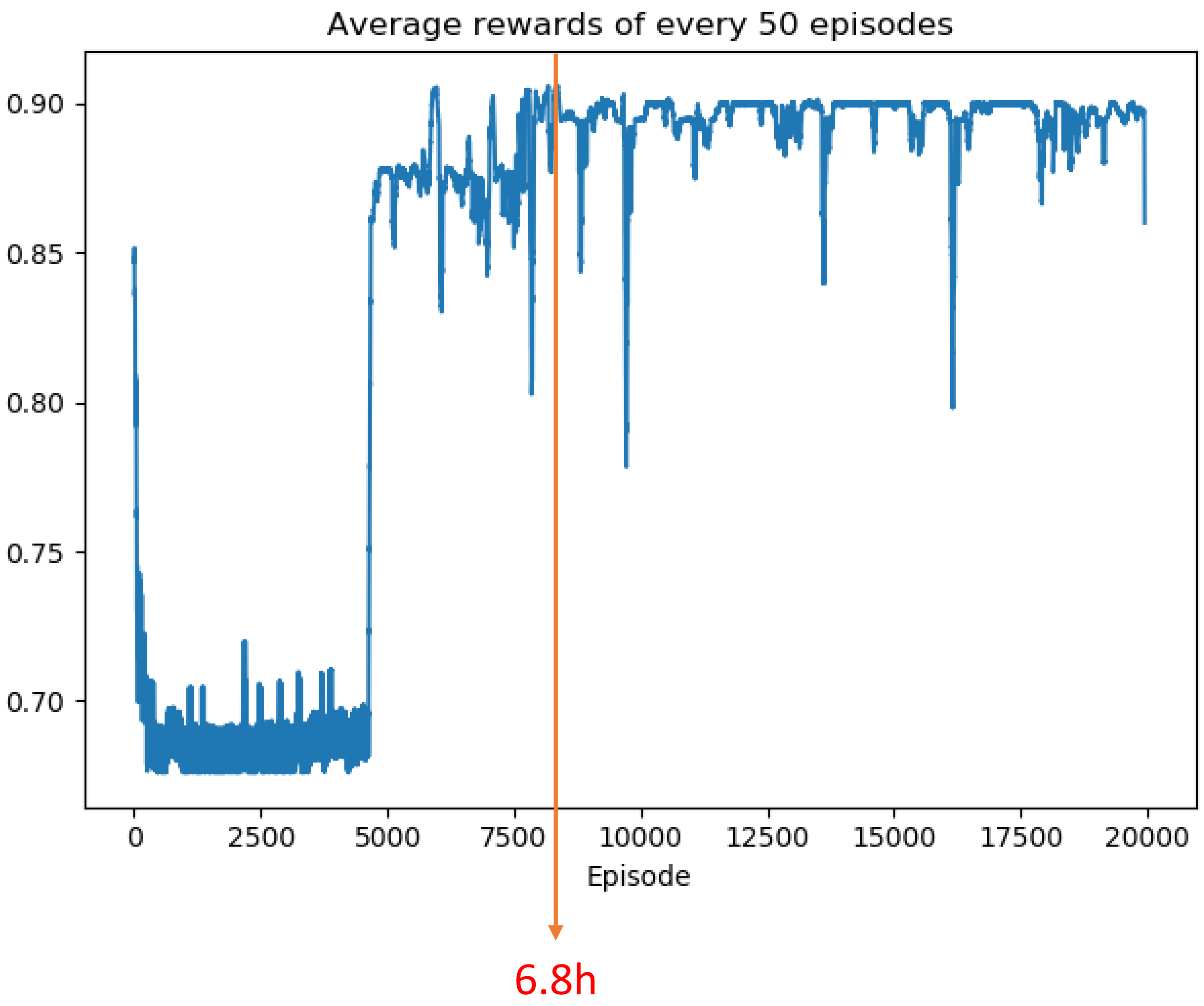
$$\hat{\boldsymbol{H}}^{(\mathrm{m})} \equiv \{I, +, -, \times, /, \exp, \log, |-|, \sqrt{}, ^{-1}, ^{2}, ^{3}\}[\phi_1, \phi_2],$$

Learning rate: 0.001 **Batch size**: 32 **Gamma** : 0.99 **Epsilon**: 1.0 (decay 0.99 min 0.1)

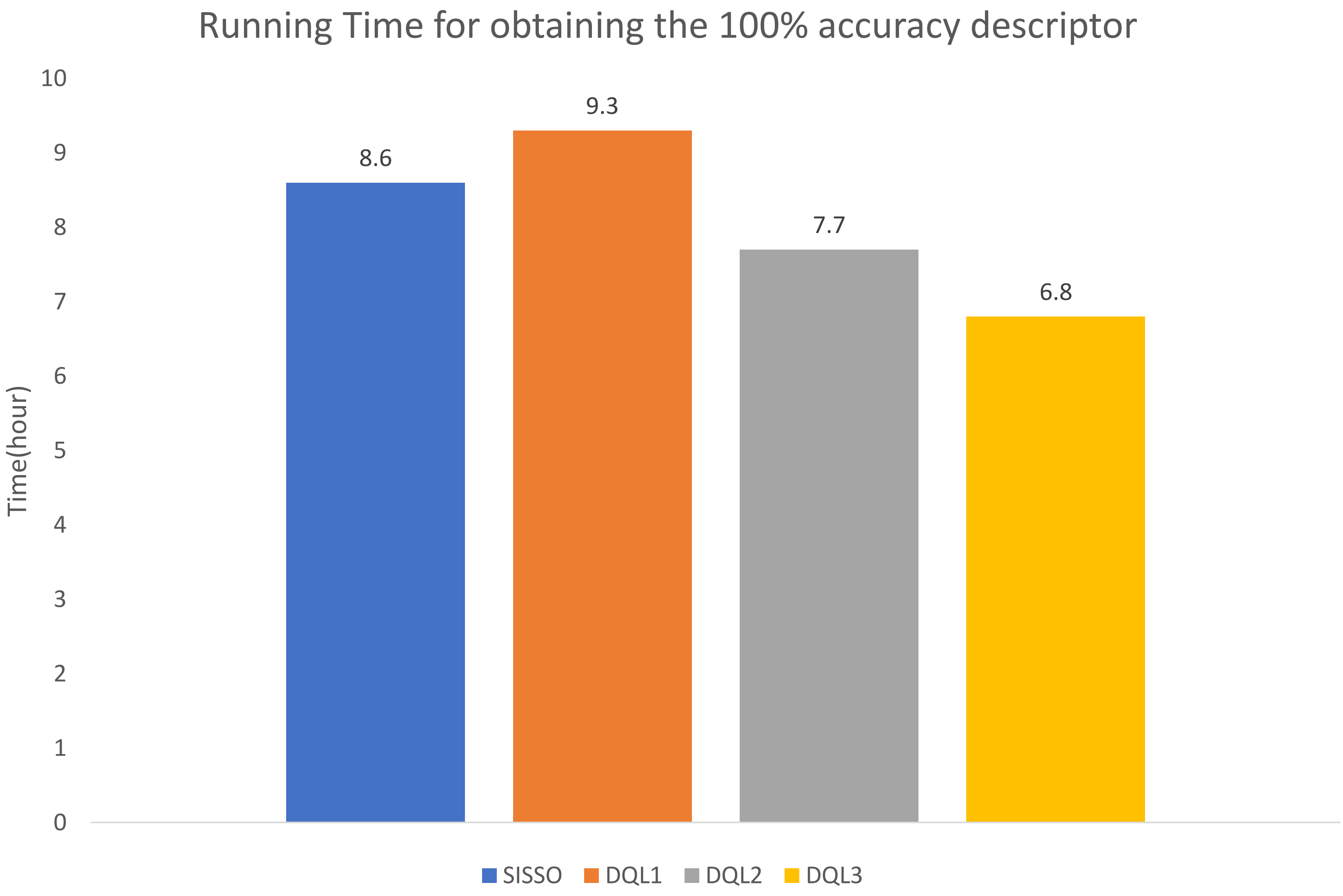
Results

Algorithm	Descriptors	Running Time	Accuracy	Features
Base	Primary features		82.5%	
SISSO	$\frac{IE_A IE_B(d_{AB} - r_{covA})}{\exp(\chi_A)\sqrt{r_{covB}}}$	8.6h	100%	830877
DQL	$\exp(d_{AB})(r_{covA} - \frac{d_{AB}}{r_{covA}})$	0.5h	97.5%	68923
DQL	$\exp(r_{covA})(d_{AB} - \frac{d_{AB}}{\chi_A})$	3h	99.2%	295512
DQL	$\frac{IE_A IE_B(d_{AB} - r_{covA})}{\exp(\chi_A)\sqrt{r_{covB}}}$	6.8h	100%	514196

Results


Average rewards of every 50 episodes
0.90
0.85
0.80
0.75
0.70
0
2500
5000
7500
10000
12500
15000
17500
20000
Episode
6.8h

Results



Running Time for obtaining the 100% accuracy descriptor
Time(hour)
10
9
8
7
6
5
4
3
2
1
0
8.6
9.3
7.7
6.8
SISSO
DQL1
DQL2
DQL3

Challenges and Future work

Actions space: Setting multidimensional actions is more accurate, but the action space will become inconsistent.

Monotonicity of Rewards: With more features combined together, classification accuracies may not be monotonic.

The many-armed issue: A general limitation of Multi-Arm Bandit algorithms (e.g. UCT), when dealing with a large number of actions compared to the number of allowed iterations, is to be biased toward exploration.

Ref: Gaudel, Romaric, and Michele Sebag. "Feature selection as a one-player game." 2010.

Dulac-Arnold, Gabriel, et al. "Deep reinforcement learning in large discrete action spaces." *arXiv preprint arXiv:1512.07679* (2015).

Khurana, Udayan, Horst Samulowitz, and Deepak Turaga. "Feature engineering for predictive modeling using reinforcement learning." *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.