新KPI快速部署异常检测算法调研（算法自动选型及参数自适配）

霍茵桐

2021/03/26 Workshop

HUAWEI

ARISE
Automated Reliable Intelligent Software Engineering

香港中文大學
The Chinese University of Hong Kong

Outline

Section I

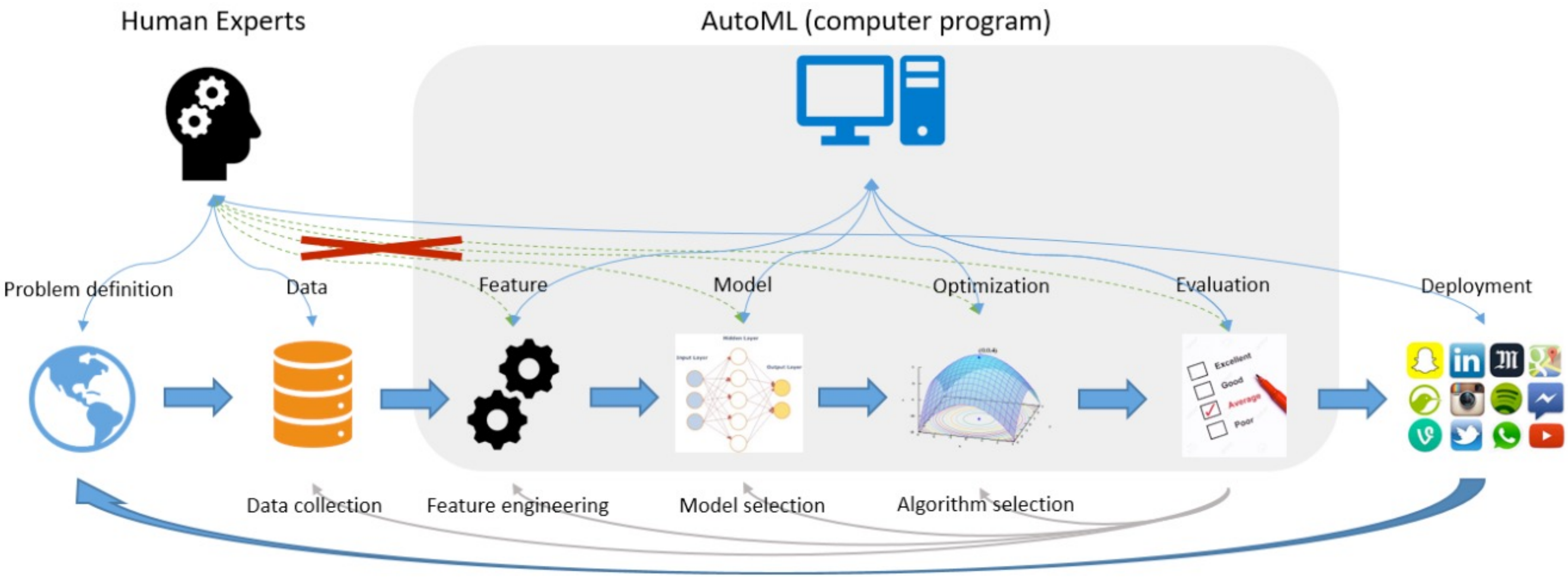
Automated Machine Learning (AutoML)

- Feature Engineering
- Optimization method
- Model selection
- Evaluation method

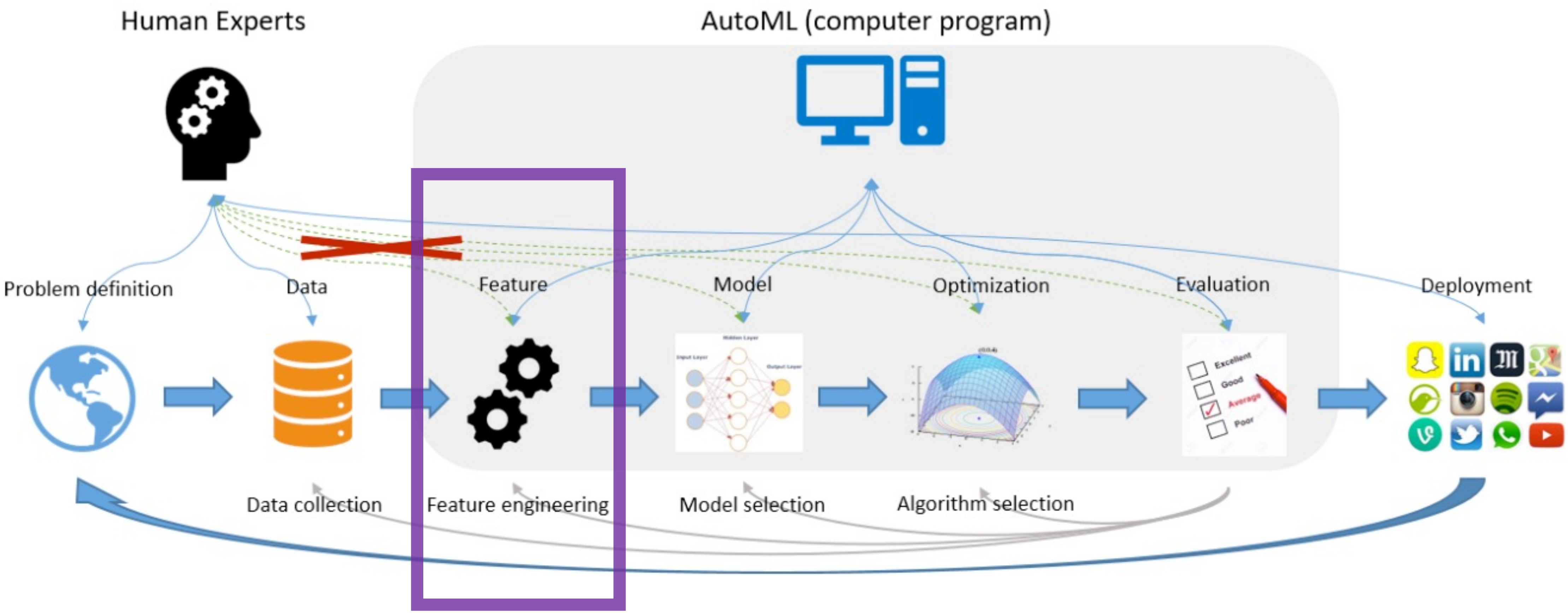
Section II

- Representative examples

Pipeline of AutoML



Pipeline of AutoML

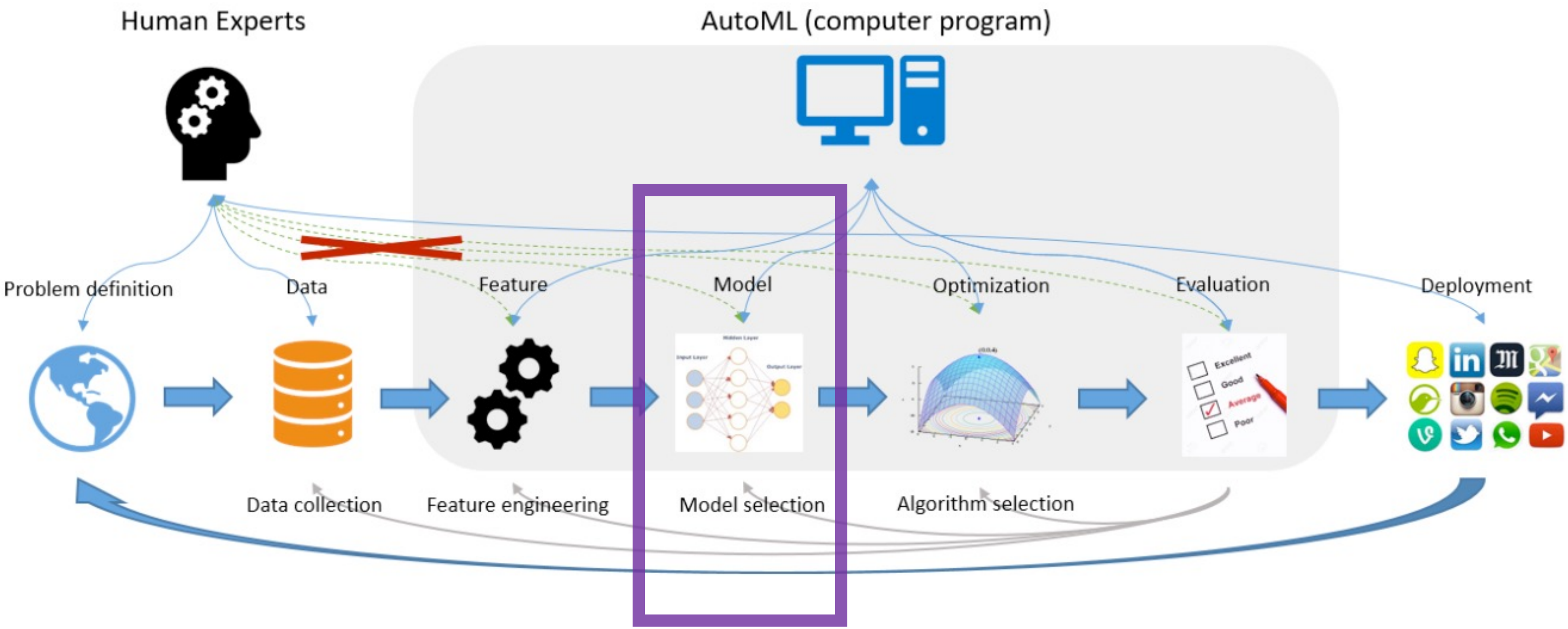


Feature Engineering

GOAL Automatically construct features from the data so that subsequent learning tools can have good performance.

Feature Selection	Build a feature subset based on the original feature set by reducing irrelevant or redundant features.
Feature Construction	Construct new features from the basic feature space or raw data to enhance the robustness and generalizability of the model.
Feature Extraction	Dimensionality-reduction process via some mapping functions

Pipeline of AutoML



Model Selection

GOAL

Automatically select classifiers and set their hyper-parameters so that good learning performance can be obtained.

1

Pick up some classifiers

- Traditional ML models
  - SVM, KNN etc..
- Deep neural network (DNN)

2

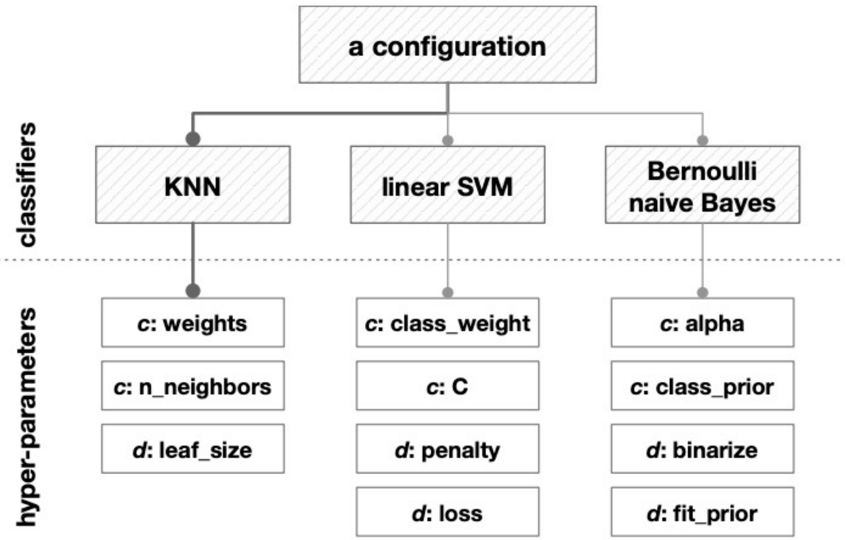
Set their corresponding hyper-parameters

Model Selection (Traditional ML)

Classification tools

	number of hyper-parameters		
	total	discrete	continuous
AdaBoost	4	1	3
Bernoulli naive Bayes	2	1	1
decision tree	4	1	3
gradient boosting	6	0	6
kNN	3	2	1
linear SVM	4	2	2
kernel SVM	7	2	5
random forest	5	2	3
logistic regression	10	4	6

Search space



Optimization algorithms

	number of hyper-parameters		
	total	discrete	continuous
GD	0	0	0
L-BFGS	1	1	0
SGD	4	1	3

Search space

Determined by configurations of optimization algorithms and the values of their hyper-parameters.

Model Selection (DNN)

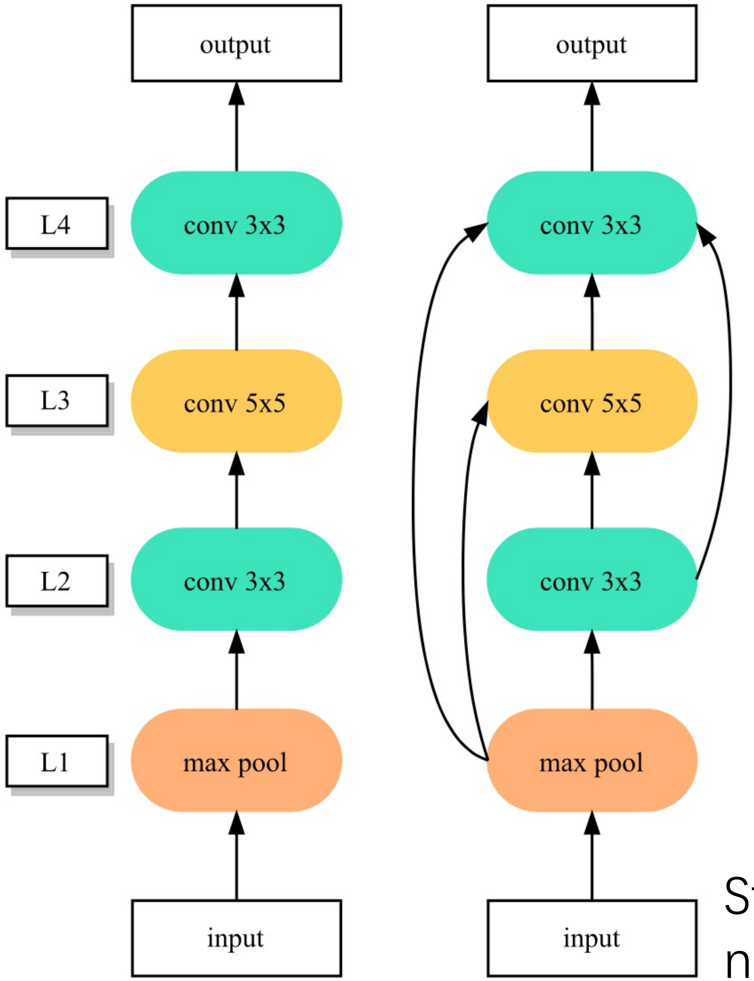
Search space

Entire-structured

Cell-based

Hierarchical

Morphism-based



✓ Intuitive and straightforward

✓ Explore more possibilities

✗ Searching deep network is onerous and computationally expensive.

Stacking a predefined number of nodes

Model Selection (DNN)

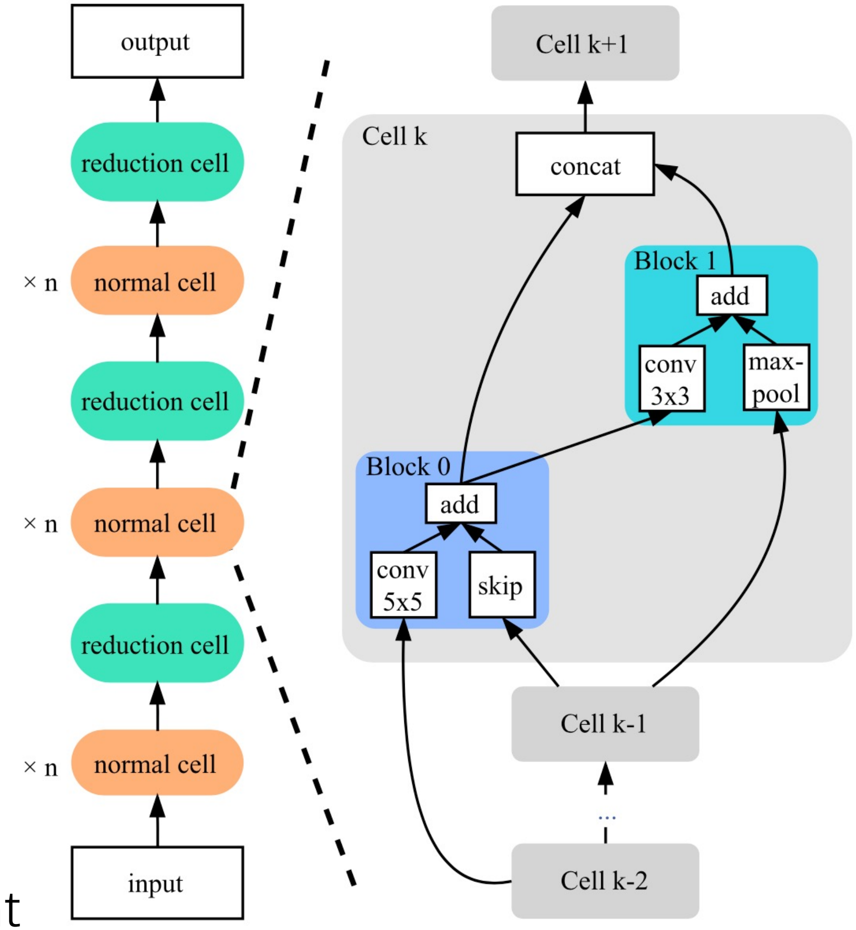
Search space

Entire-structured

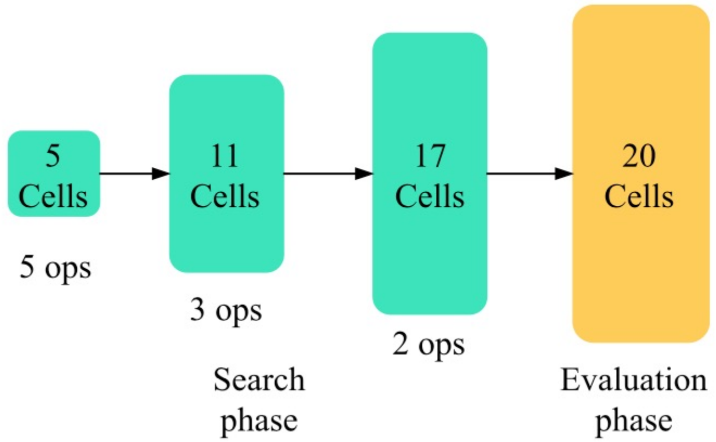
Cell-based

Hierarchical

Morphism-based



Reduce train-evaluate phase gap



P-DARTS

✓ Easy to expand

- Network consists different type of cells.
- Search the best cell structure

✓ Lower complexity

Model Selection (DNN)

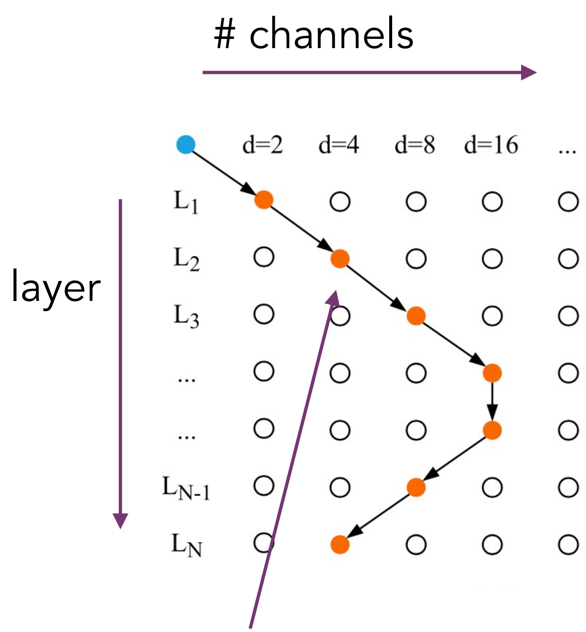
Search space

Entire-structured

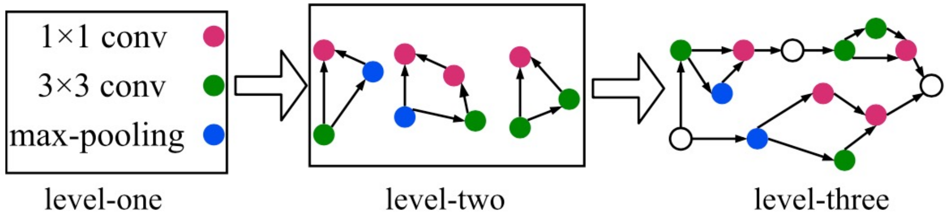
Cell-based

Hierarchical

Morphism-based



Final selected network-level structure



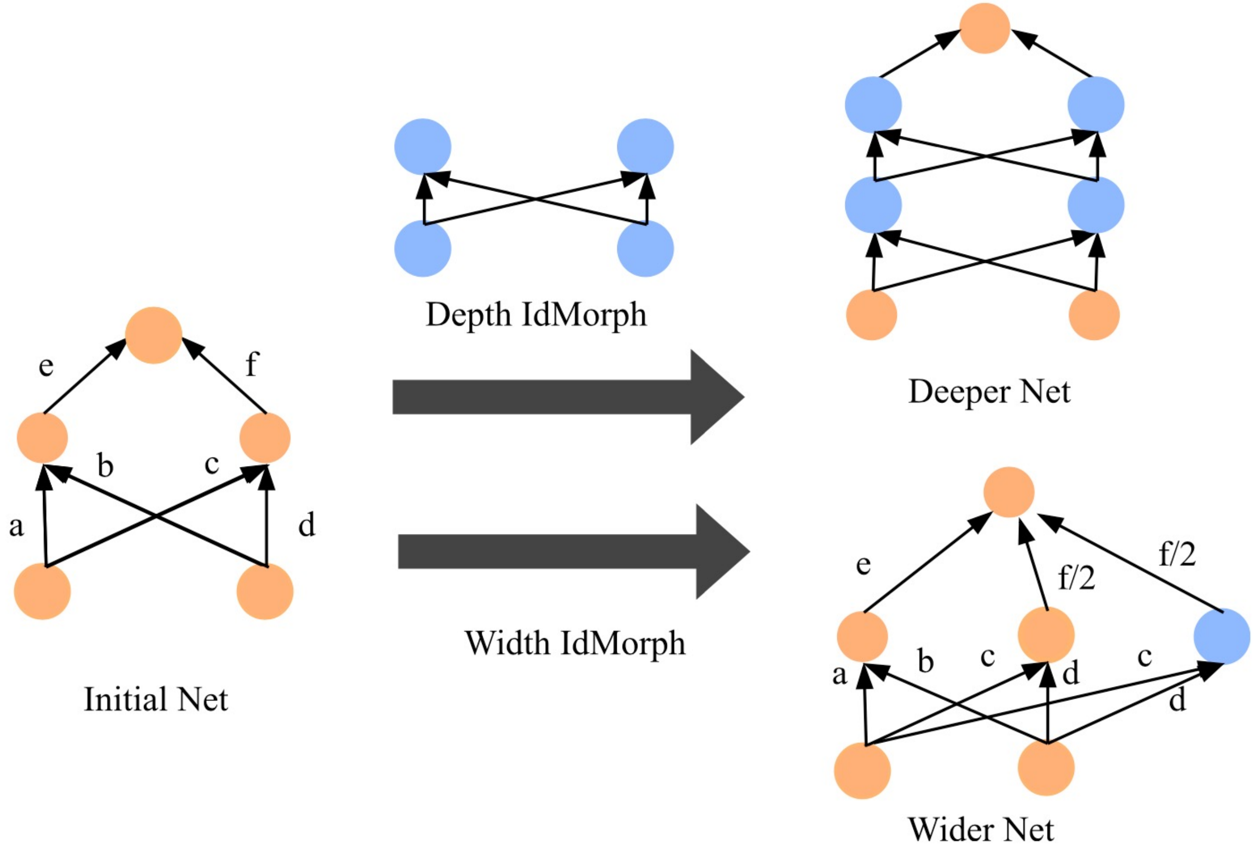
- Three-level hierarchical architecture representation.
- The level-one primitive operations are assembled into level-two cells. Then level-two cells are viewed as primitive operations and assembled into level-three cell.

✔ Network level

Model Selection (DNN)

- **Search space**
  - Entire-structured
  - Cell-based
  - Hierarchical
  - **Morphism-based**

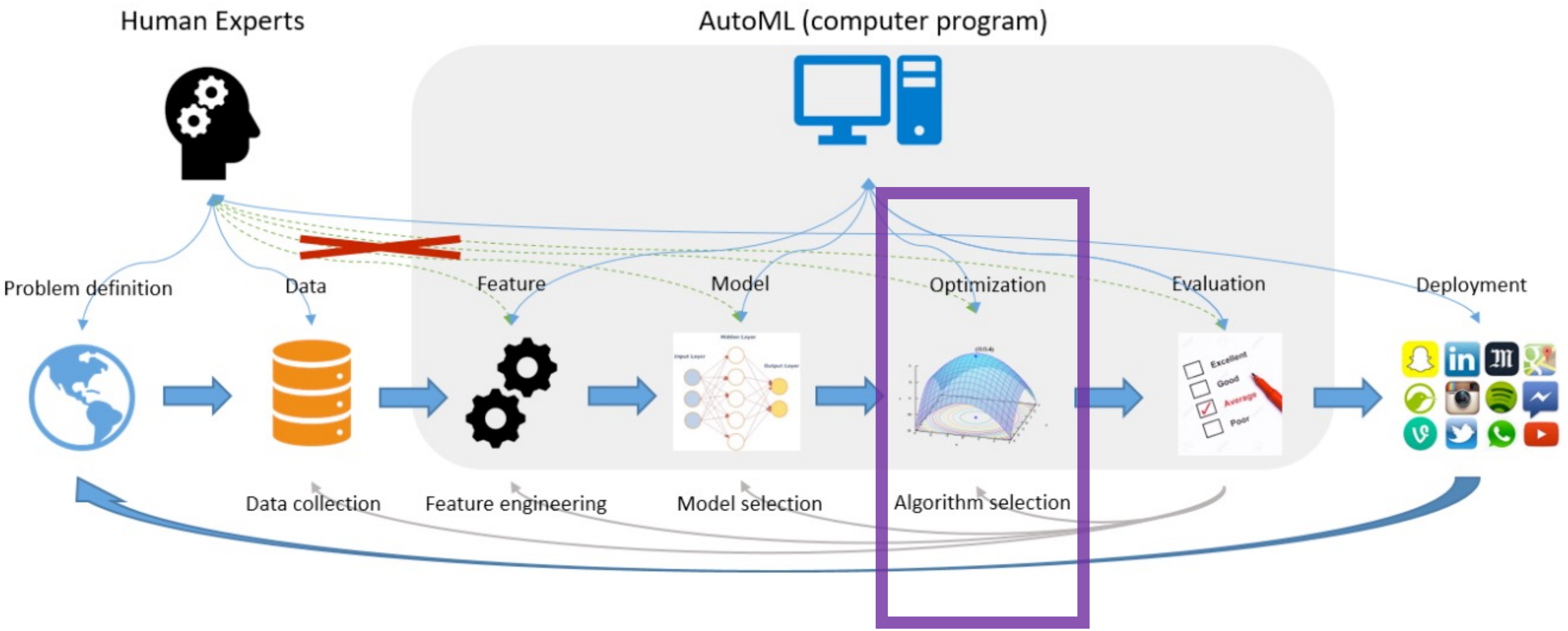
Previous work do not directly change the structures.



Deeper

Wider

Pipeline of AutoML

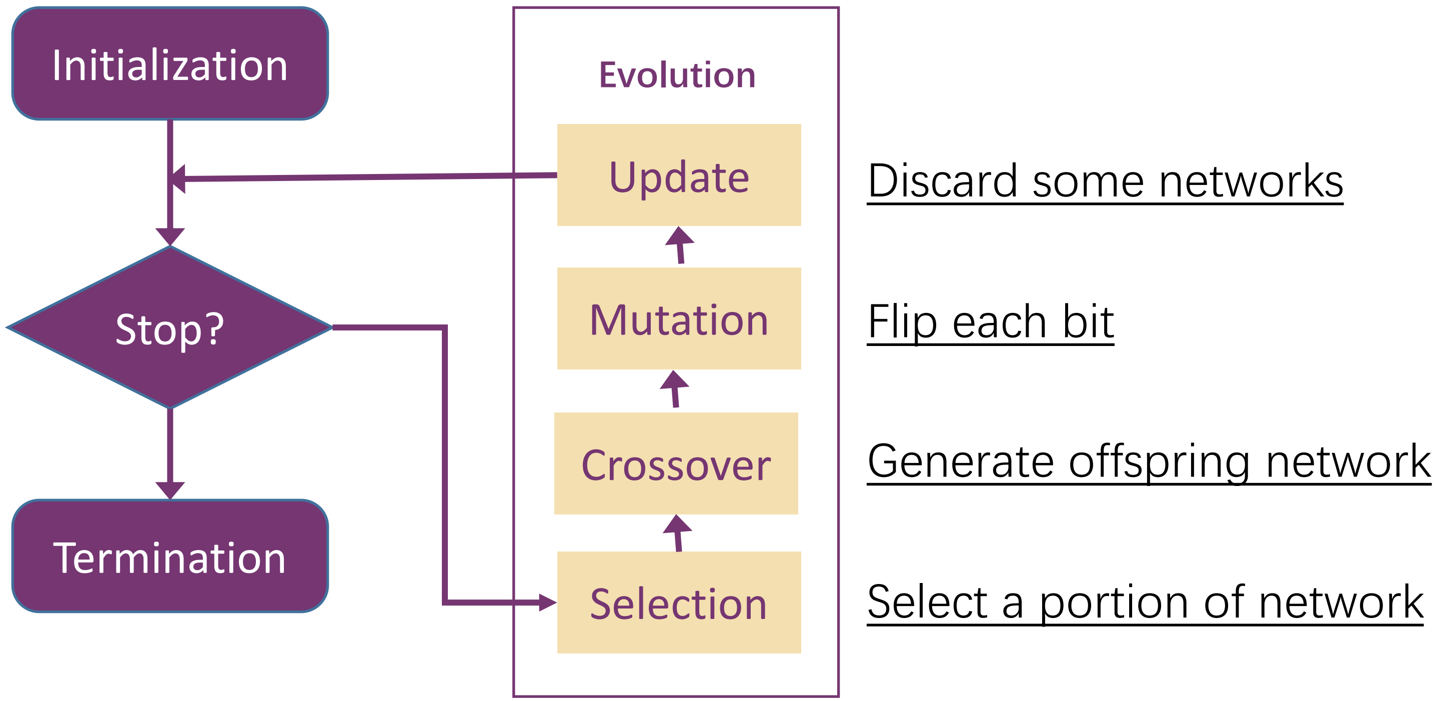


Architechture optimization

- Evolution algorithm
- Reinforcement learning
- Gradient descent
- Surrogate model-based optimization
- Grid and random search
- Hybrid optimization method

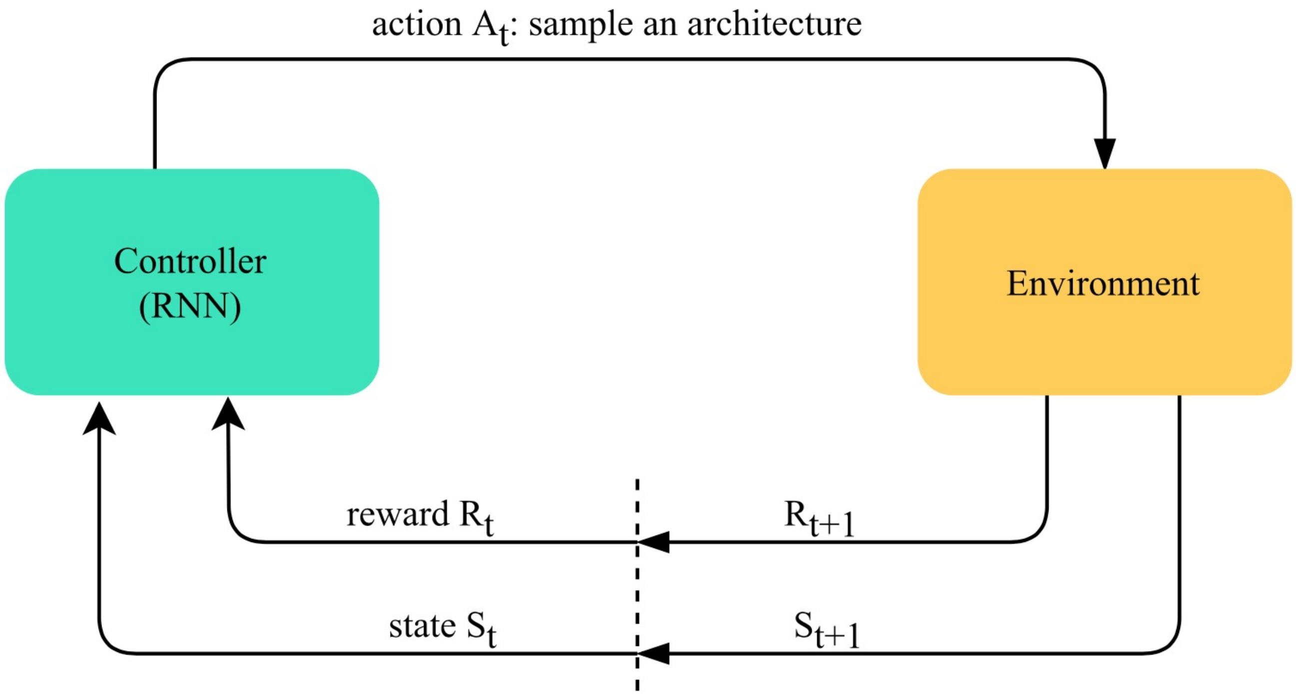
STEP1 Encode the architecture to a string of bits

STEP2 Apply the evolution algorithm



Architechture optimization

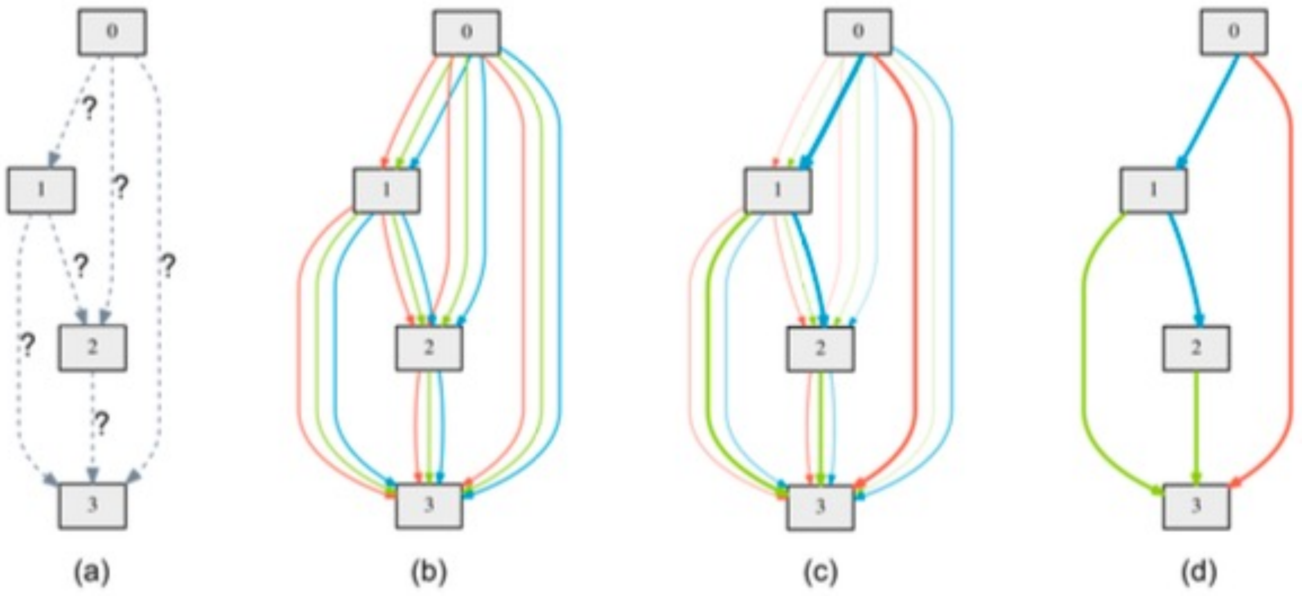
- Evoluation algorithm
- Reinforcement learning
- Gradient descent
- Surrogate model-based optimization
- Grid and random search
- Hybrid optimization method



1) Sample a new architecture.
2) Use reward and state to update the sampling strategy.

Architechture optimization

- Evoluation algorithm
- Reinforcement learning
- Gradient descent
- Surrogate model-based optimization
- Grid and random search
- Hybrid optimization method



Loss func
$$\min_{\alpha} \quad \mathcal{L}_{val}(w^*(\alpha), \alpha)$$
$$\text{s.t.} \quad w^*(\alpha) = \text{argmin}_w \; \mathcal{L}_{train}(w, \alpha)$$

Simultaneously learn the architecture and weights within all the mixed operations (e.g., weights of the convolution filters).

Architechture optimization

Evoluation algorithm

Reinforcement learning

Gradient descent

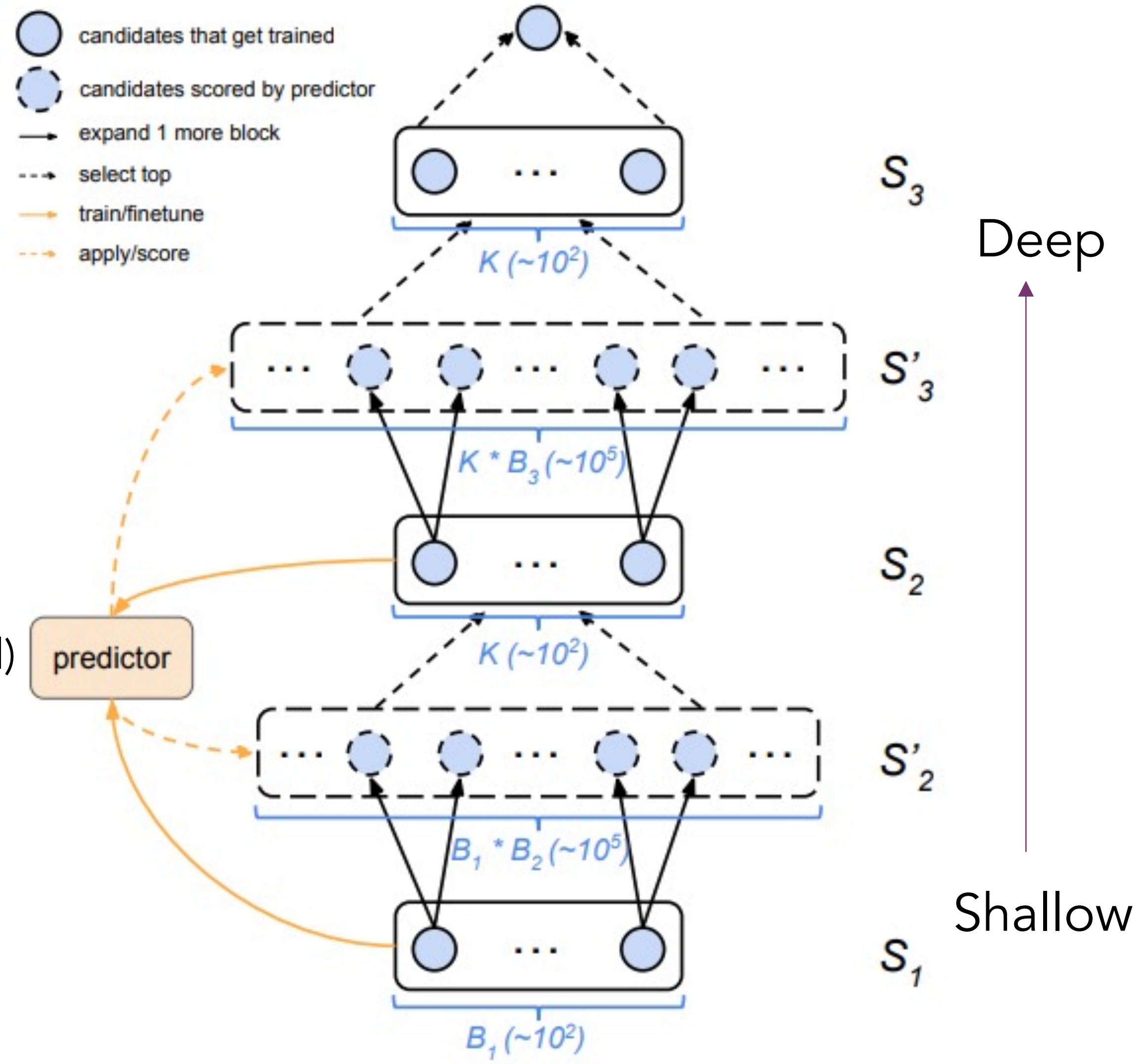
Surrogate model-based optimization

Grid and random search

Hybrid optimization

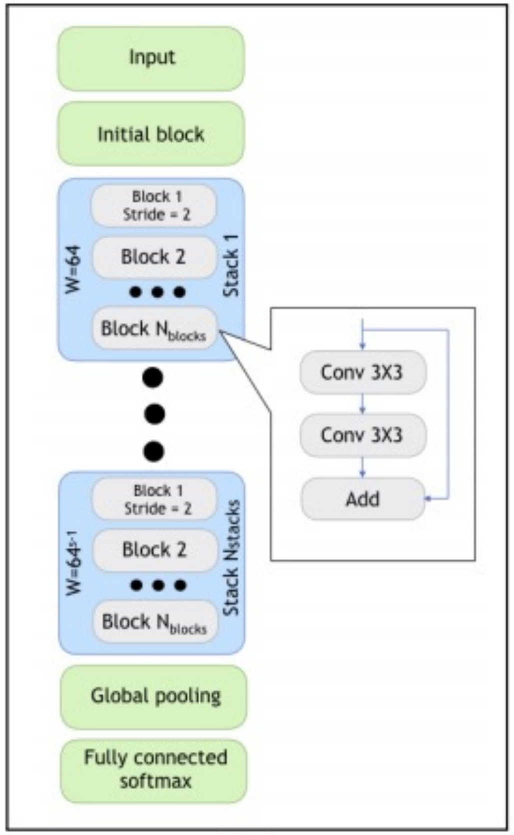
Surrogate model: mimic the behavior of the simulation model as closely as possible while being computationally cheaper to evaluate

(Surrogate model)

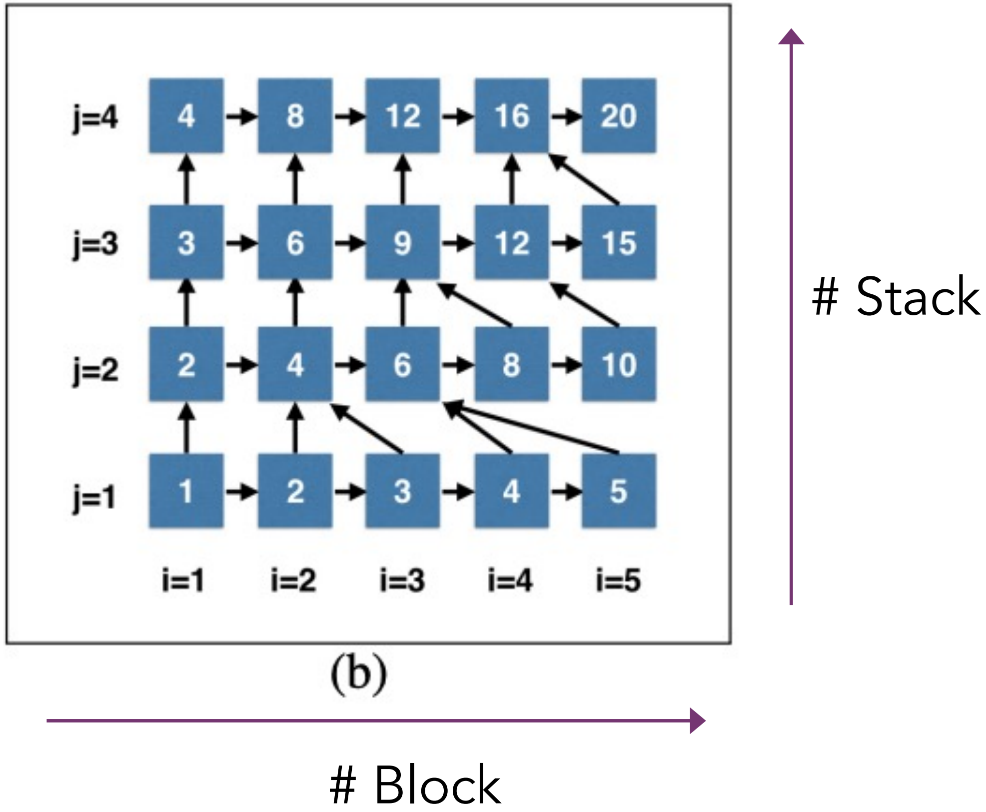


Architechture optimization

- Evoluation algorithm
- Reinforcement learning
- Gradient descent
- Surrogate model-based optimization
- Grid and random search
- Hybrid optimization method



(a)



(b)

Architechture optimization

Evoluation algorithm

Reinforcement learning

Gradient descent

Surrogate model-based optimization

Grid and random search

Hybrid optimization method

EA

- Mature global optimization method with robustness
- High computational resources
- Evolution operations perform randomly

RL

- Learn complex architectural patterns
- CANNOT guarantee searching efficiency and stability

GD

- Improve search efficiency
- Limit diversity

EA+RL

EA+GD

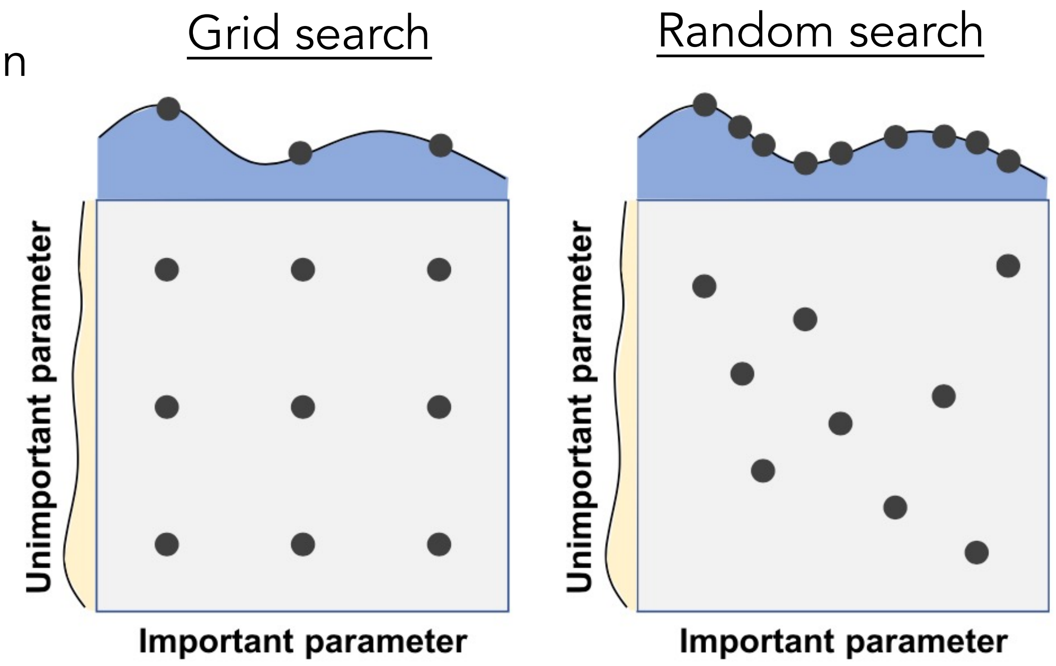
EA+SMBO

Hyperparameter optimization

- Grid and random search

  Bayesian optimization

  Gradient-based optimization



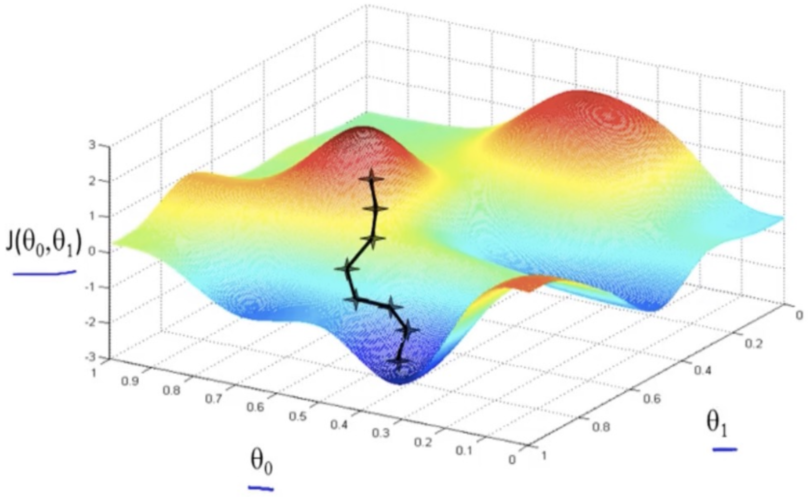
Nine trials cover only three important parameter values.

Explore nine distinct values.

Random search is more likely to find the optimal combination than grid search.

Hyperparameter optimization

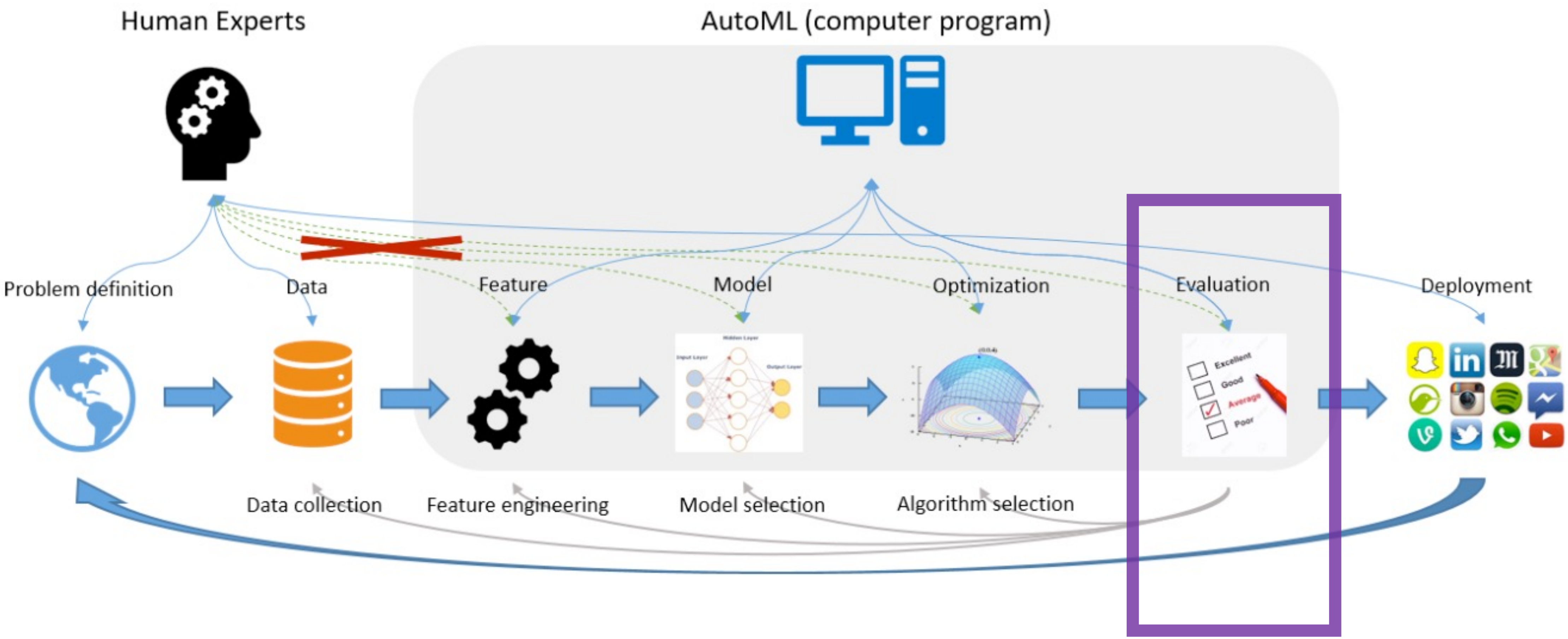
- Grid and random search
- Bayesian optimization
- Gradient-based optimization




```
SMBO(f, M, T, S)
O ← ∅
for t ← 1 to T
    z* ← arg max_z S(z; M)
    Evaluate f(z*)          ▷ Expensive step
    O ← O ∪ (z*, f(z*))
    Fit a new model M to O
return O
```

Core idea: Build a surrogate model to fit the record dataset. The dataset includes architecture parameters and their corresponding performance.

Pipeline of AutoML



Model evaluation

GOAL

Automatically measure the performance of a candidate network.

Challenges:

An intuitive method is to train the network to convergence and then evaluate its performance. However, this method requires extensive time and computing resources.

- How to provide *fast* evaluation?
- How to provide *accurate* evaluation?

Model evaluation

- Evaluator
- **Low-fidelity**
- Weight-sharing
- Surrogate
- Early-stopping

- The number of images or the resolution of images can be decreased.
- Low-fidelity model evaluation can be realized by reducing the model size.
- There is a weak correlation between performance after short and long training times, thus confirming that a prolonged search for network configurations is necessary.

Model evaluation

Evaluator

Low fidelity

Weight-sharing

Surrogate

Early-stopping

- Previously, once a network has been evaluated, it is dropped. Hence, the technique of weight sharing is used to accelerate the process of neural architecture search.

- It is possible to use knowledge from prior tasks to accelerate network design.

Model evaluation

Evaluator

Low-fidelity

Weight-sharing

Surrogate

Early-stopping

- Once a good approximation has been obtained, it is trivial to find the configurations that directly optimize the original expensive objective.
- Surrogate evaluators can predict not only the performance of learning tools, but also the training time and model parameters.

Model evaluation

- Evaluator
- Low-fidelity
- Weight-sharing
- Surrogate
- **Early-stopping**

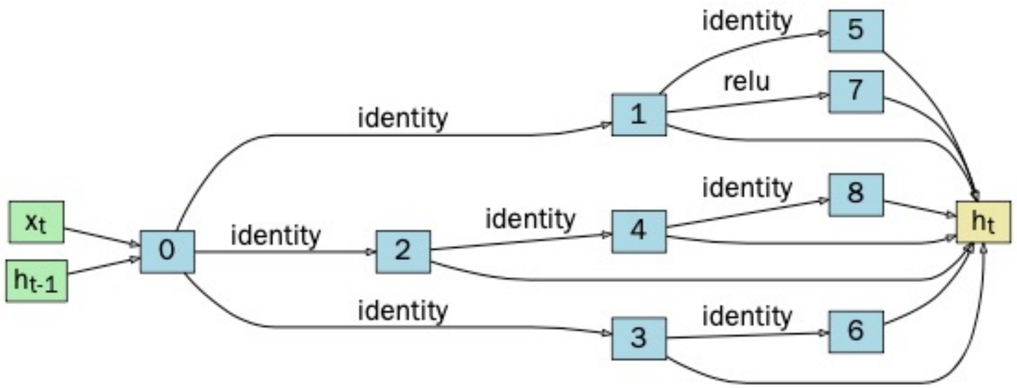
- It is usually used to cut down the training time for unpromising configurations.
- If a poor early-stage performance is observed, the evaluator can terminate the training and report a low performance to indicate that the candidate is unpromising.
- Early stop cuts down the total running time of AutoML, but also introduces noise and bias to the estimation.

Representative examples

AutoML in NLP

Search for an architecture:

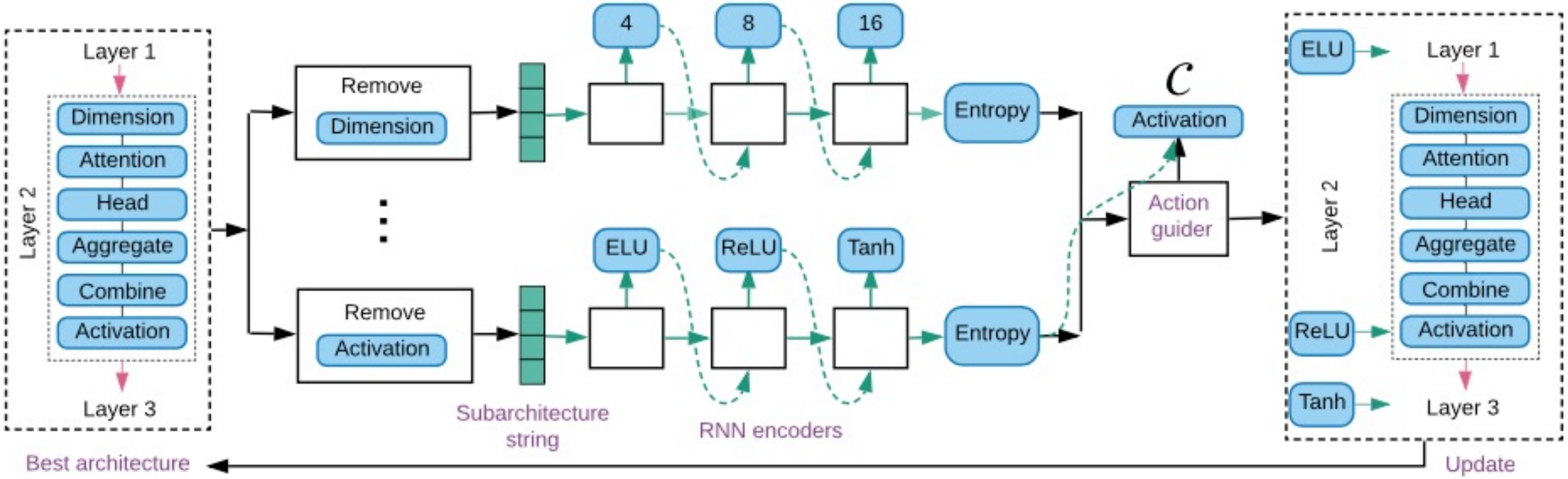
- Recurrent cell consists of 8 nodes.
- The candidate operation set of every edge contains 5 activation functions.



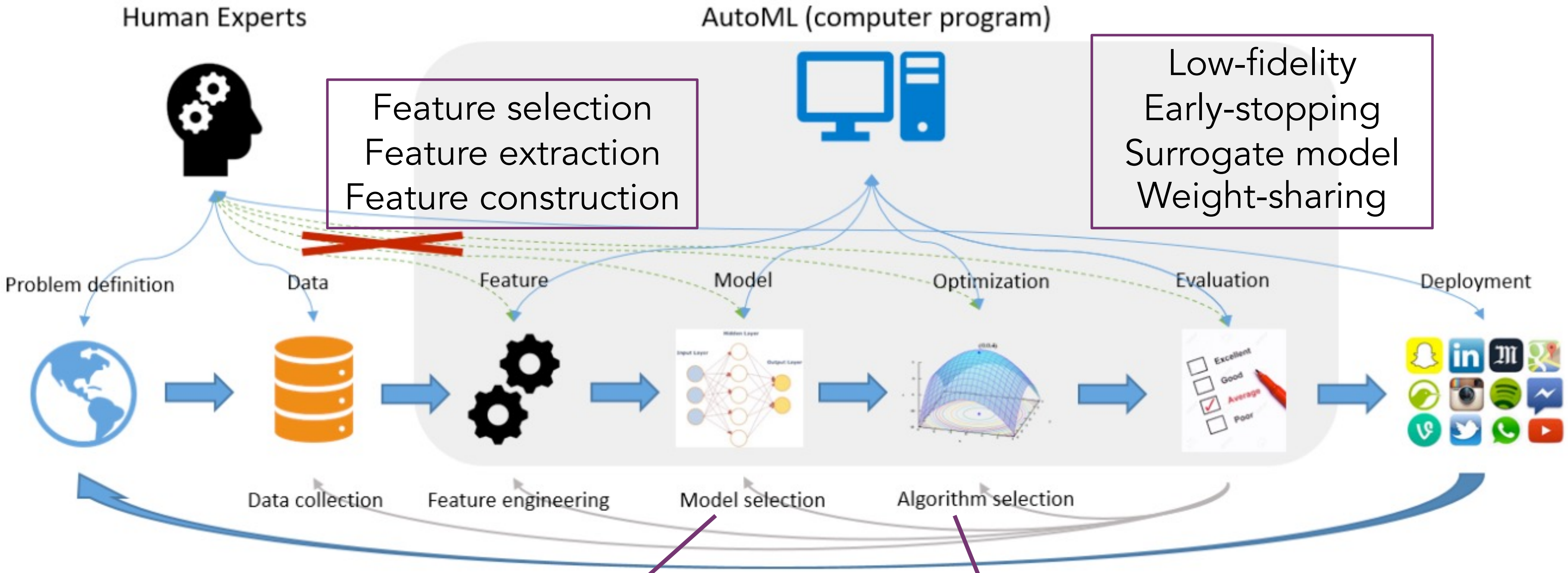
Model	F1
best published	
BiLSTM-CRF (Lample et al., 2016)	90.94
BiLSTM-CRF+ELMo (Peters et al., 2018)	92.22
BERT Base (Devlin et al., 2018)	92.40
BERT Large (Devlin et al., 2018)	92.80
BiLSTM-CRF+PCE (Akbik et al., 2019)	93.18
Random RNNs w/o pre-trained LM	90.64
DARTS w/o pre-trained LM	91.05
I-DARTS ($n = 2$) w/o pre-trained LM	90.96
I-DARTS ($n = 1$) w/o pre-trained LM	91.23
Random RNNs	92.89
DARTS	93.13
I-DARTS ($n = 2$)	93.14
I-DARTS ($n = 1$)	93.47

Representative examples

AutoML in GNN



Conclusion



Feature selection
Feature extraction
Feature construction

Low-fidelity
Early-stopping
Surrogate model
Weight-sharing

Traditional models
Deep neural networks

Hyperparameeter optimization
Architecture optimization

Classifiers, Entire-structured, Cell-based, Hierarchical, Morphism-based

Evoluation algorithm, Reinforcement learning, Gradient descent, Surrogate model-based optimization, Grid and random search, Hybrid optimization method

Thank you!

HUAWEI

香 港 中 文 大 學
The Chinese University of Hong Kong