# AutoML Challenge @ PAKDD 2019
## Team: ML Intelligence

Mengjiao Bao[1,2], Huixue[1], Yihuan Mao[1], Yujing Wang[1]

Microsoft Research Asia[1], Beihang University[2]

# Content

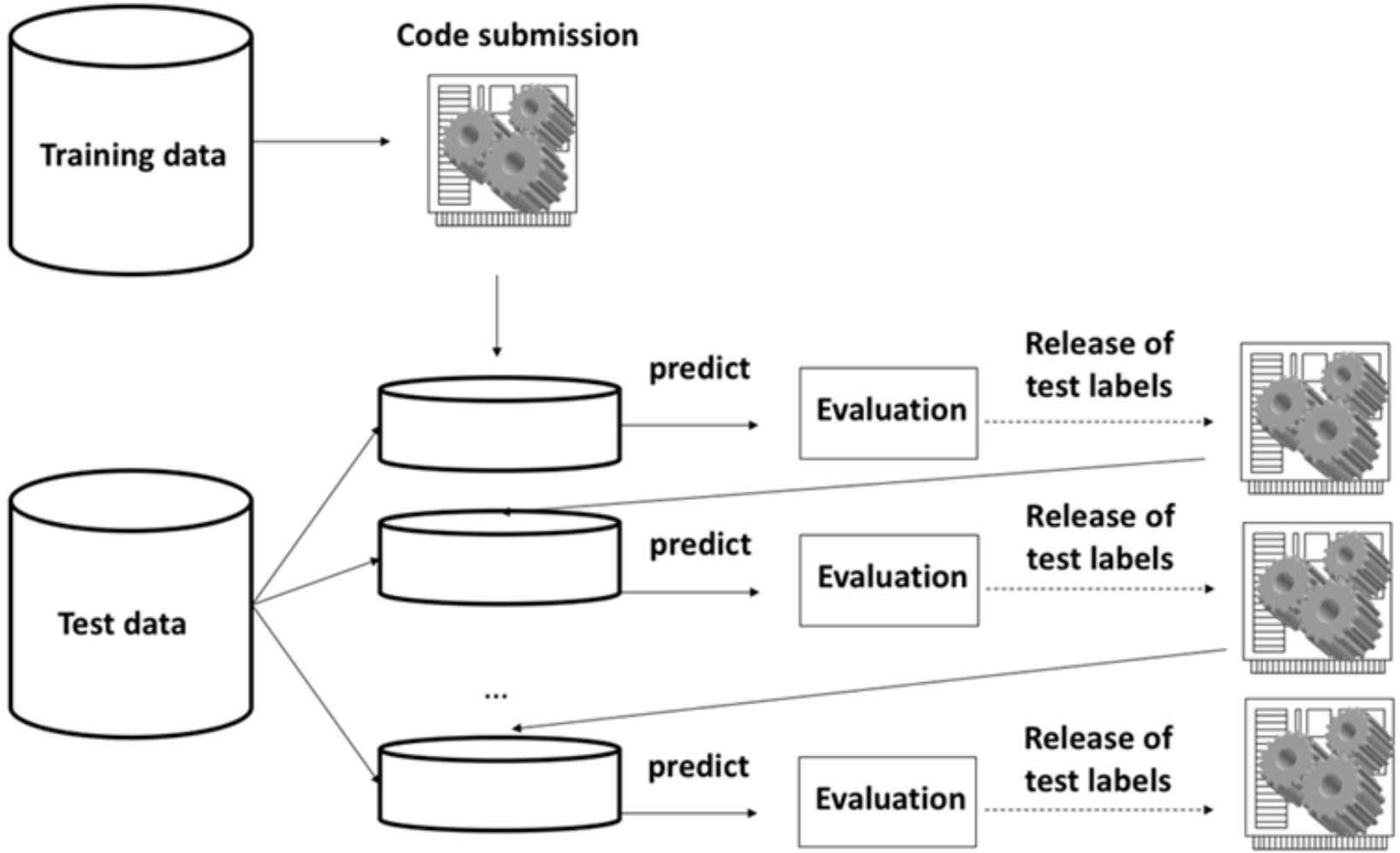
- Overview
- Key Challenge
- Pipeline
- Concept Drift
- Auto Part

# Overview

TIME

CAT/MV

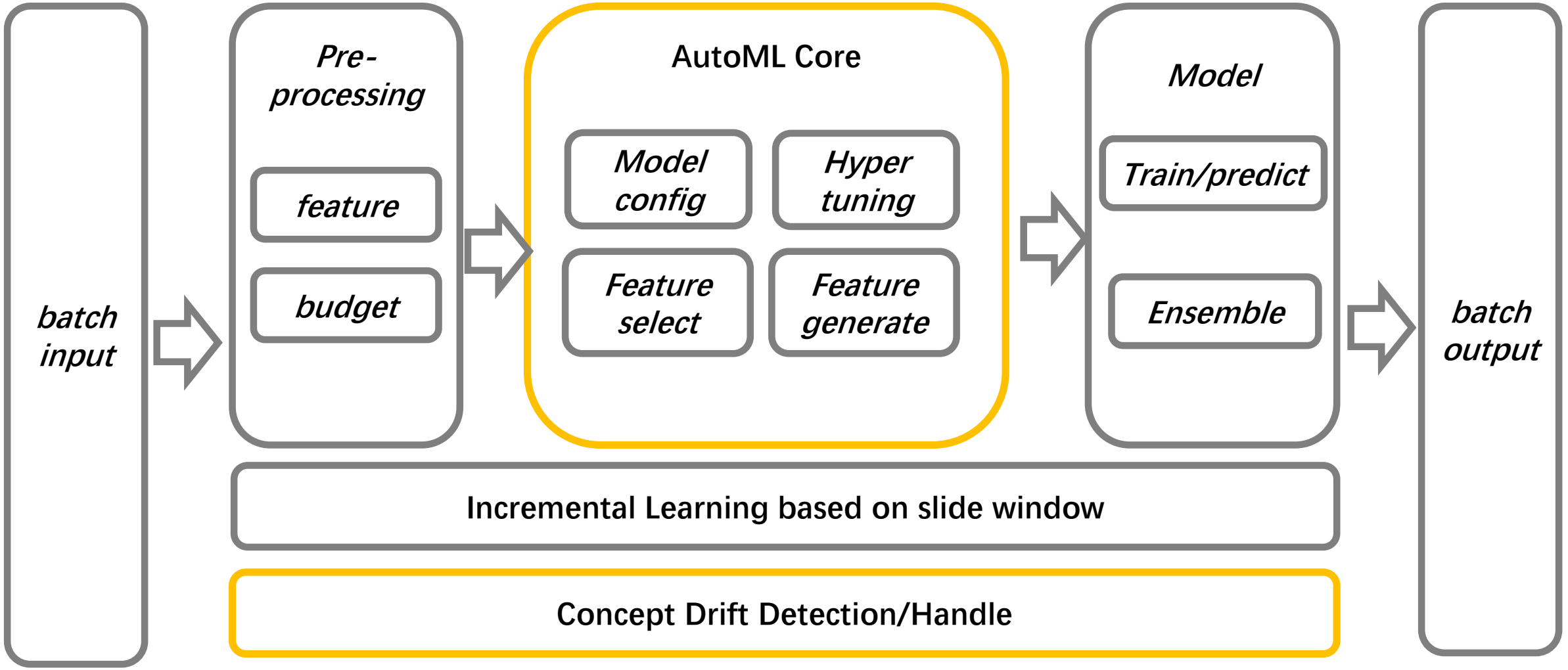
NUM



# Difficulties

- **Algorithm scalability:** large datasets(10^7)
- **Varied feature types:** continuous, binary, ordinal, categorical, multi-value categorical, temporal
- **Concept drift:** Distribution change over time
- **Lifelong setting:** Datasets chronologically -> 10 batches, test -> labels reveal.

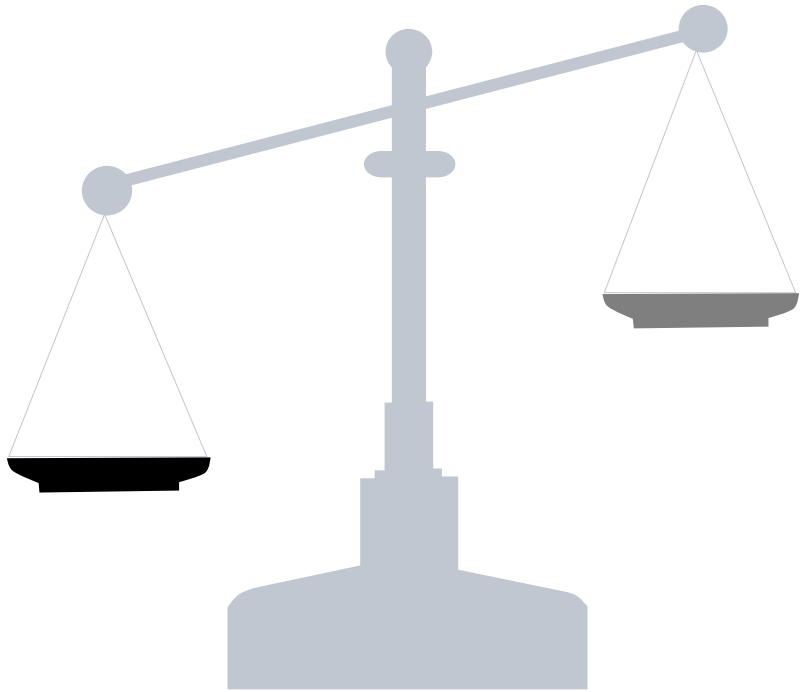
# Pipline


batch input
Pre-processing
feature
budget
AutoML Core
Model config
Hyper tuning
Feature select
Feature generate
Model
Train/predict
Ensemble
batch output
Incremental Learning based on slide window
Concept Drift Detection/Handle

# Auto Model Config

Performance

1.More Feature
2.More Data
3.More Parameter



Budget

1.Time Limit
2.RAM Limit

$$Estimate = len(Data) * (\alpha + \beta + \gamma) * len(Cat_{feature})$$

# Auto Feature Selection

- Rank methods(for selection and beam search)
  - Naive feature importance generated by LightGBM
  - Bagging methods: 5-kflods Feature importance average / Bagging with different LightGBM models
  - Feature selection with null importance

- Drop methods
  - time accumulation LightGBM importance

# Feature Encoding Methods for HD cat

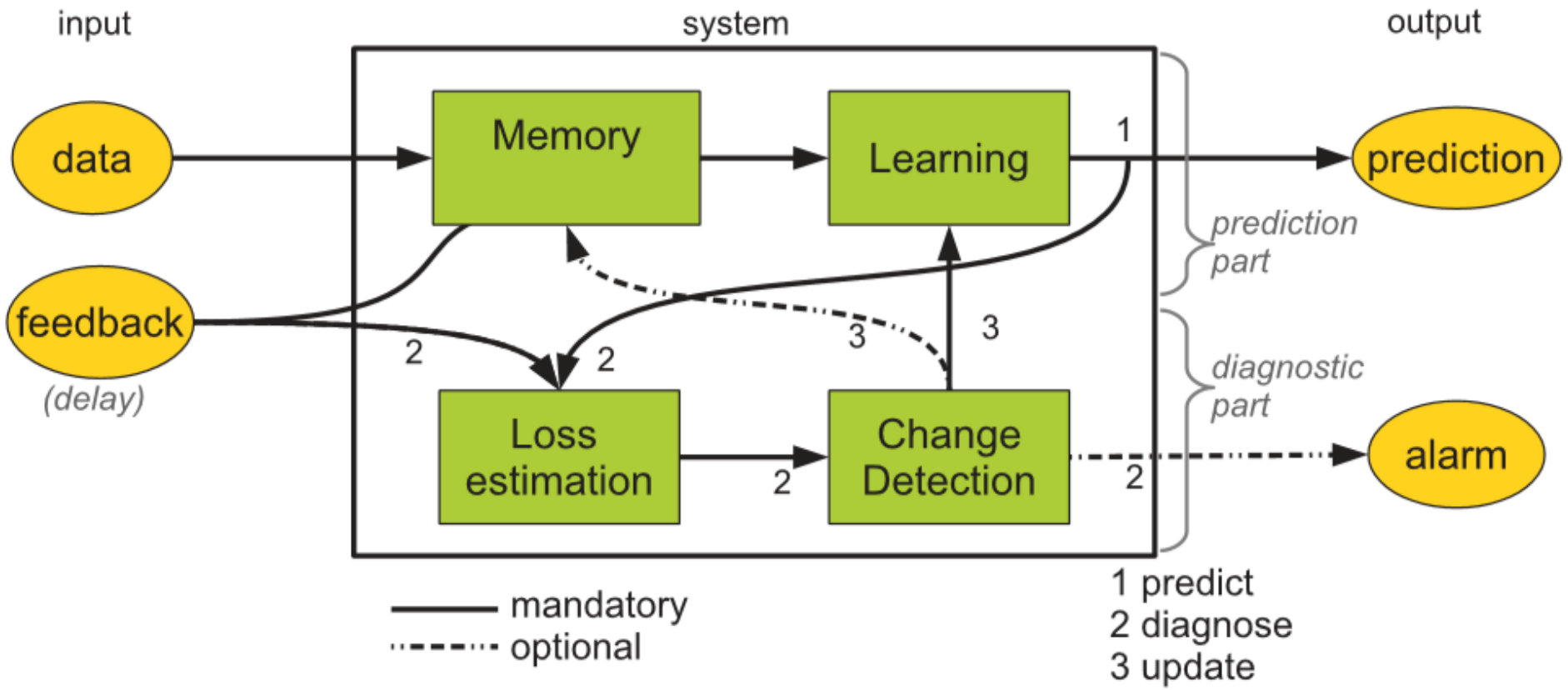
| | | | | | | |
|---|---|---|---|---|---|---|
| AutoGBT | Hash encode | 0.46 | 0.18 | 0.45 | 0.33 | 0.68 |
| | Incremental encode | 0.503 | 0.278 | 0.535 | 0.451 | 0.732 |
| | Hash + incremental | 0.506 | 0.277 | 0.505 | 0.450 | 0.729 |
| Our method | Hash encode | 0.491 | 0.281 | 0.480 | 0.451 | 0.730 |
| | Count encode | 0.513 | 0.328 | 0.484 | 0.569 | 0.737 |
| | Encode based on first batch | 0.533 | 0.333 | 0.490 | 0.571 | 0.740 |
| | **Incremental (slide window)** | **0.540** | **0.338** | **0.555** | **0.575** | **0.775** |

# Auto hyper parameter tuning （fix lr）

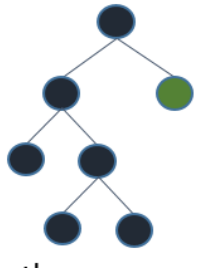
- a. Hyperband
- b. Tree of Parzen Estimators(implements by hyperopt)
- c. metis

| | A | B | C | D | E |
|---|---|---|---|---|---|
| metis | 0.812 | 0.666 | **0.848** | 0.780 | 0.885 |
| TPE | 0.808 | 0.669 | **0.884** | 0.784 | 0.883 |
| Hand min | 0.812 | 0.661 | **0.822** | 0.774 | 0.885 |
| Had max | 0.814 | 0.672 | **0.826** | 0.782 | 0.886 |

# Handle Concept Drift Overview



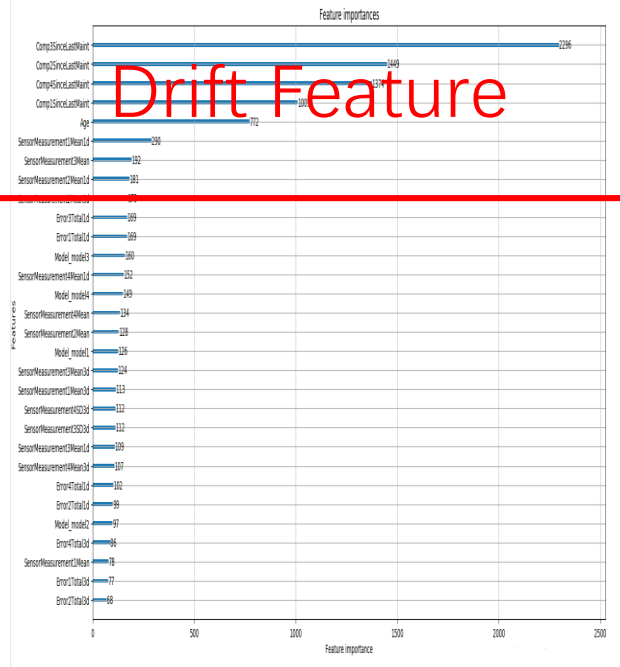
# Detection and Handle



TRAIN POSITIVE

TEST NEGATIVE

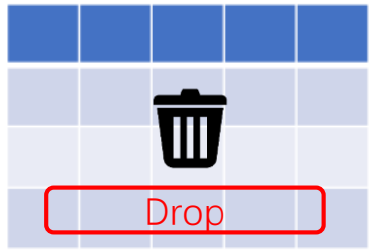
Simple GBDT split train and test




Decision Tree importance

**Detection**


Basic important feature




High-order feature

**handle**

# Detection and Handle

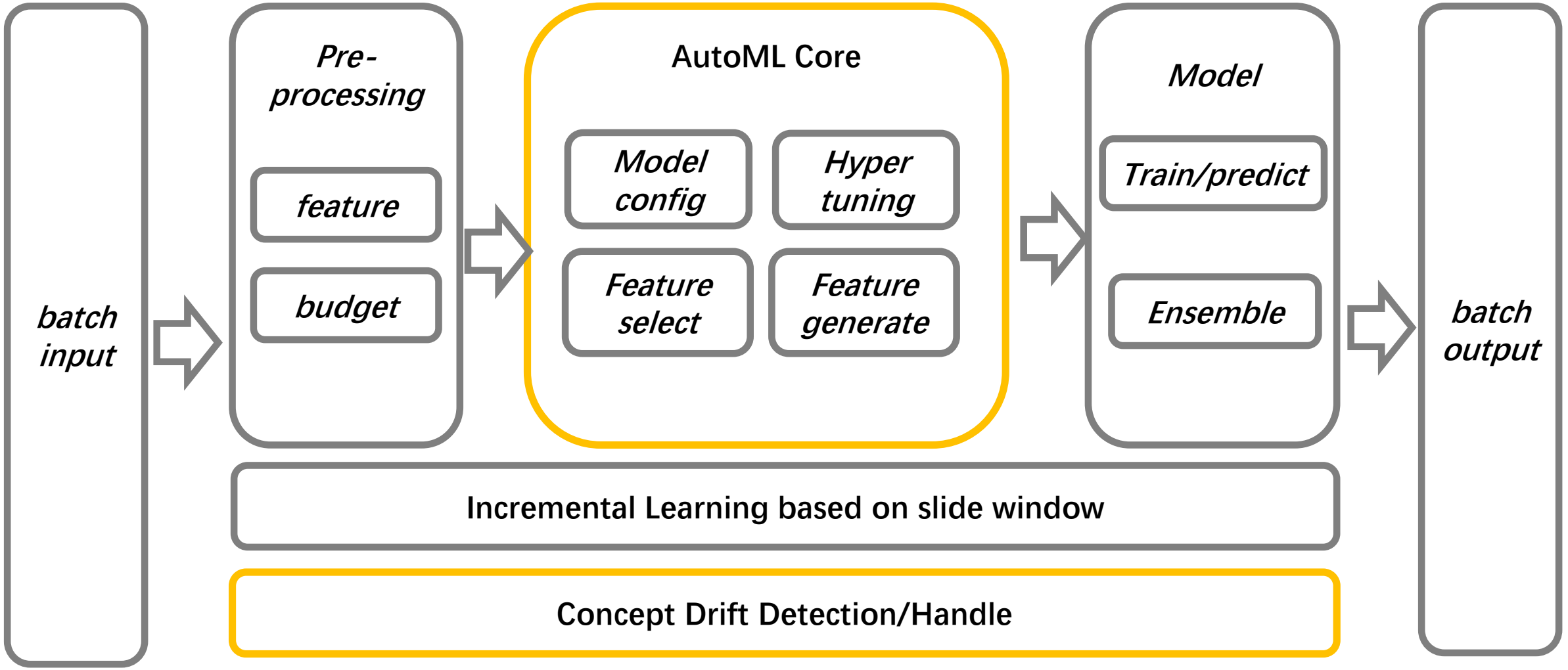
**Algorithm 2** Concept Detection and Handle

**Input:** TrainSet $D_{train}$, TestSet $D_{test}$

Set TrainSet $D_{train}$ as positive, TestSet $D_{test}$ as negative;
Train a binary classifier by GBDT to split train and test.
get the *auc* score in training procedure.
Calculate the feature importance and train set *auc* score.
if $auc > 0.65$, get the most $n$ largest gain feature as $F_{drift}$.
get concept drift feature set $F_{drift}$
**for** $f \in F_{drift}$: **do**
    **if** $f \notin F_{highorder}$: **then**
        delete $f$ from $F_{drift}$.
    **end if**
**end for**

**return** Out put $F'_{drift}$

# Conclusion


batch input
Pre-processing
feature
budget
AutoML Core
Model config
Hyper tuning
Feature select
Feature generate
Model
Train/predict
Ensemble
batch output
Incremental Learning based on slide window
Concept Drift Detection/Handle

# Q&A


https://github.com/Microsoft/nni