Learning Multiple Stock Trading Patterns with Temporal Routing Adaptor and Optimal Transport

KDD 2021

Hengxu Lin*, Dong Zhou*, Weiqing Liu, Jiang Bian
Sun Yat-sen University, Microsoft Research Asia

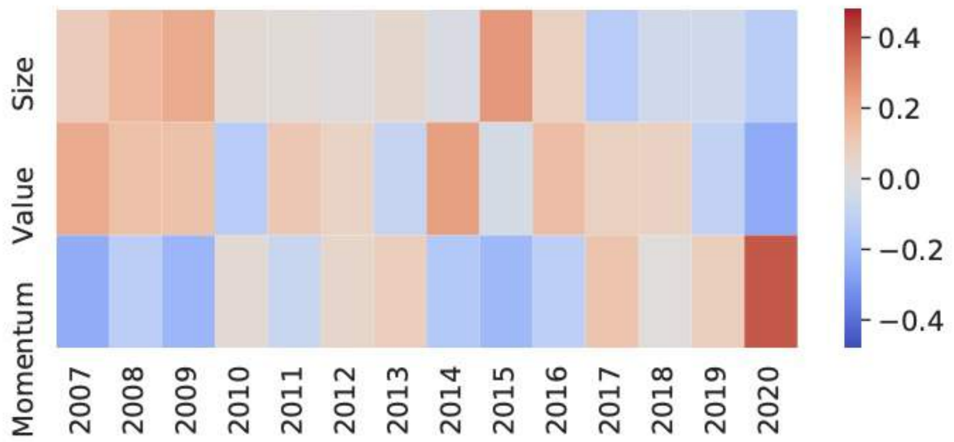
Motivation

- Investors hold various beliefs and strategies, and multiple trading patterns* in the stock market could be observed.

** A trading pattern means the causal relation between the available information at the current time (i.e., feature) and the stock price movement in the future (i.e., label).*

Motivation

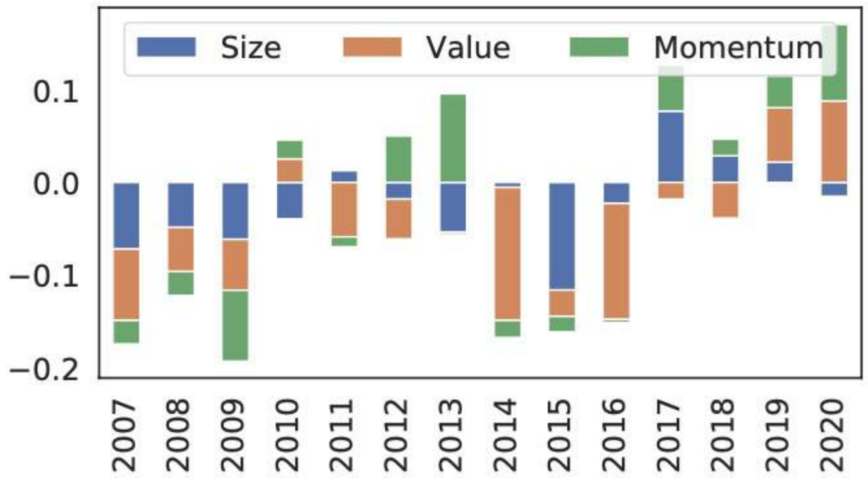
- Investors hold various beliefs and strategies, and multiple trading patterns* in the stock market could be observed.
  - e.g., excess return relative to market of three common trading strategies (size, value, momentum) changed and dominated different period



** A trading pattern means the causal relation between the available information at the current time (i.e., feature) and the stock price movement in the future (i.e., label).*

Motivation

- But simply following the identical distribution assumption fails to capture different trading patterns.
  - e.g., linear coefficients between excess return and strategy-related feature

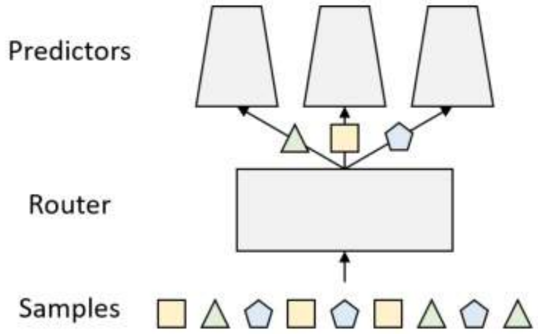


Methodology

- Temporal Routing Adaptor (TRA)
  - Empower existing NN based models with the ability to model multiple trading patterns.

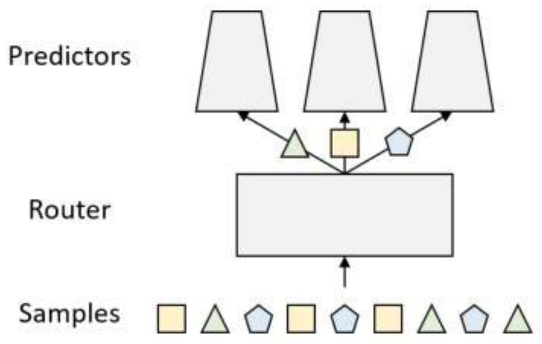
Methodology

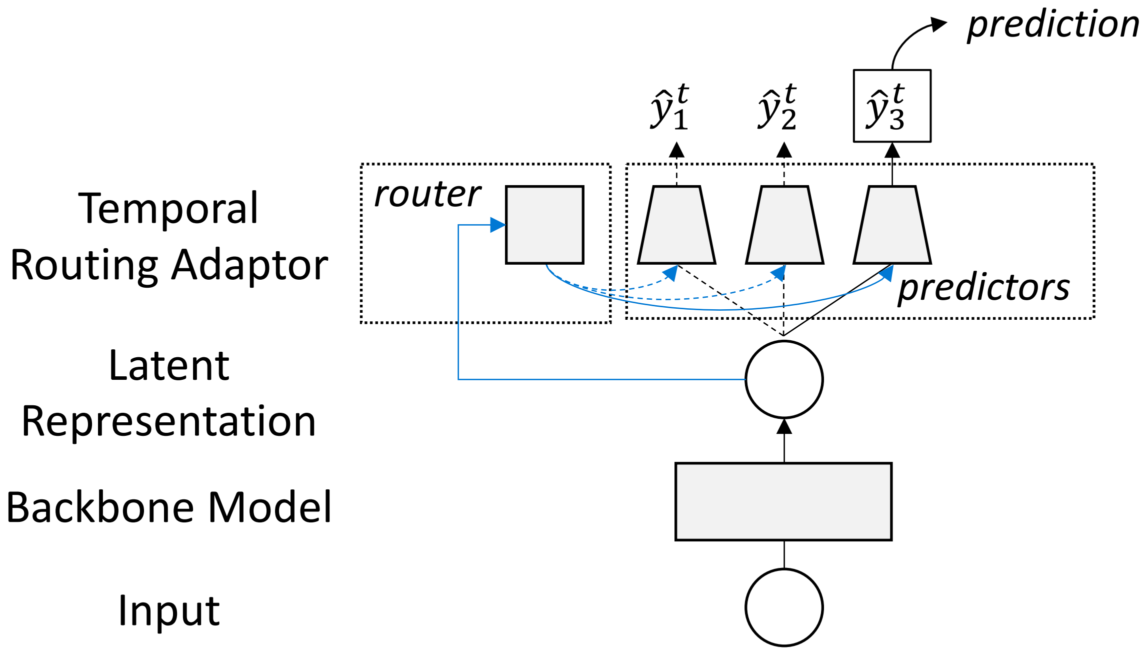
- Temporal Routing Adaptor (TRA) = predictors + router
  - Empower existing NN based models with the ability to model multiple trading patterns.
  - Predictors: Responsible for representing multiple patterns
  - Router: Responsible for detecting a sample's underlying pattern and its assignment



Methodology

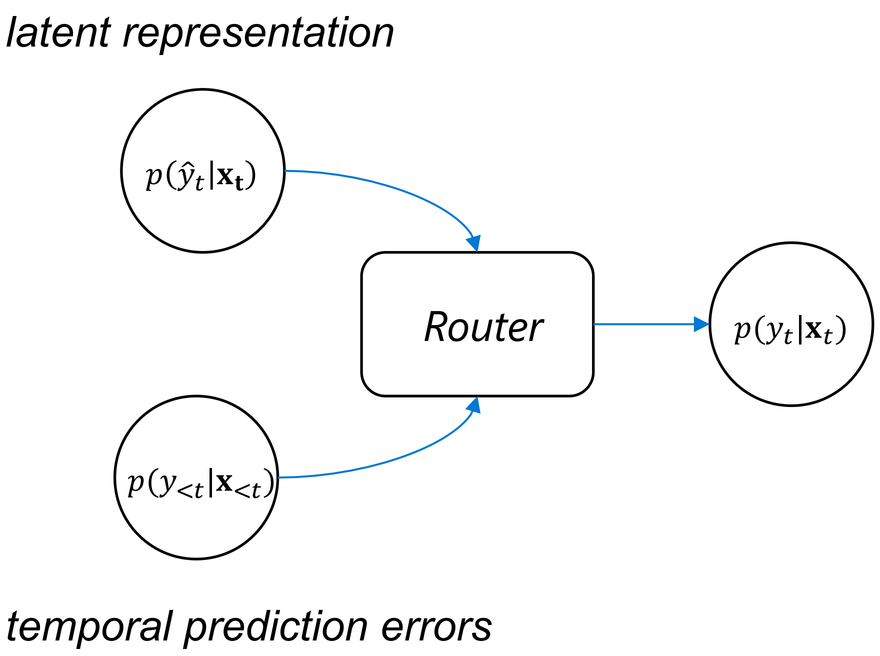
- Temporal Routing Adaptor (TRA) = predictors + router
  - Predictors: Responsible for representing multiple patterns
  - Extending our TRA on a backbone model avoid heavy parameters and provide fair comparison with backbone model.

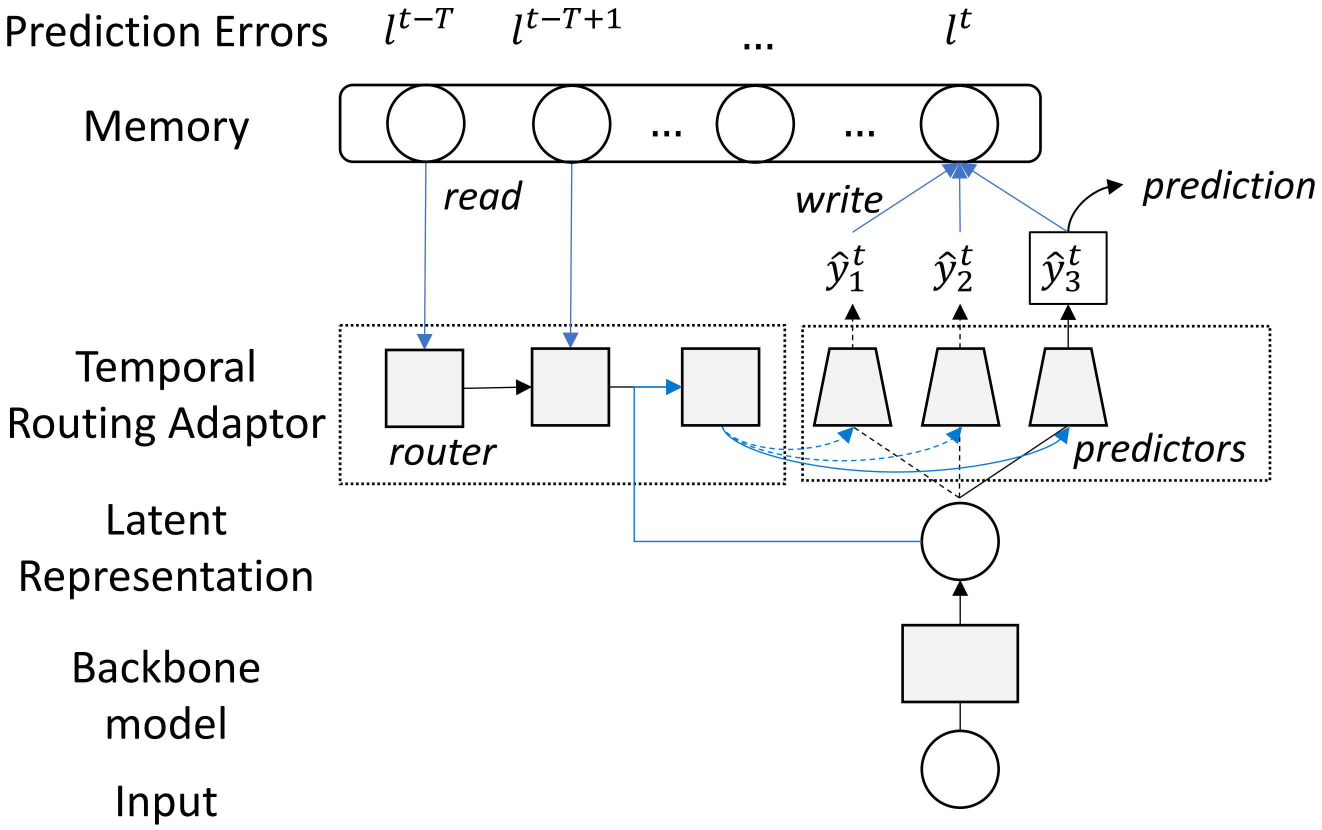




Methodology

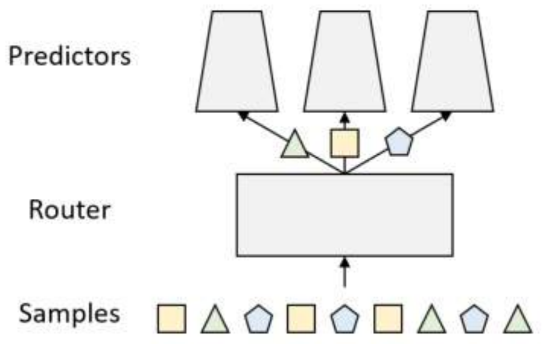
- Temporal Routing Adaptor (TRA) = predictors + router
  - Router: Responsible for detecting a sample's underlying pattern and its assignment

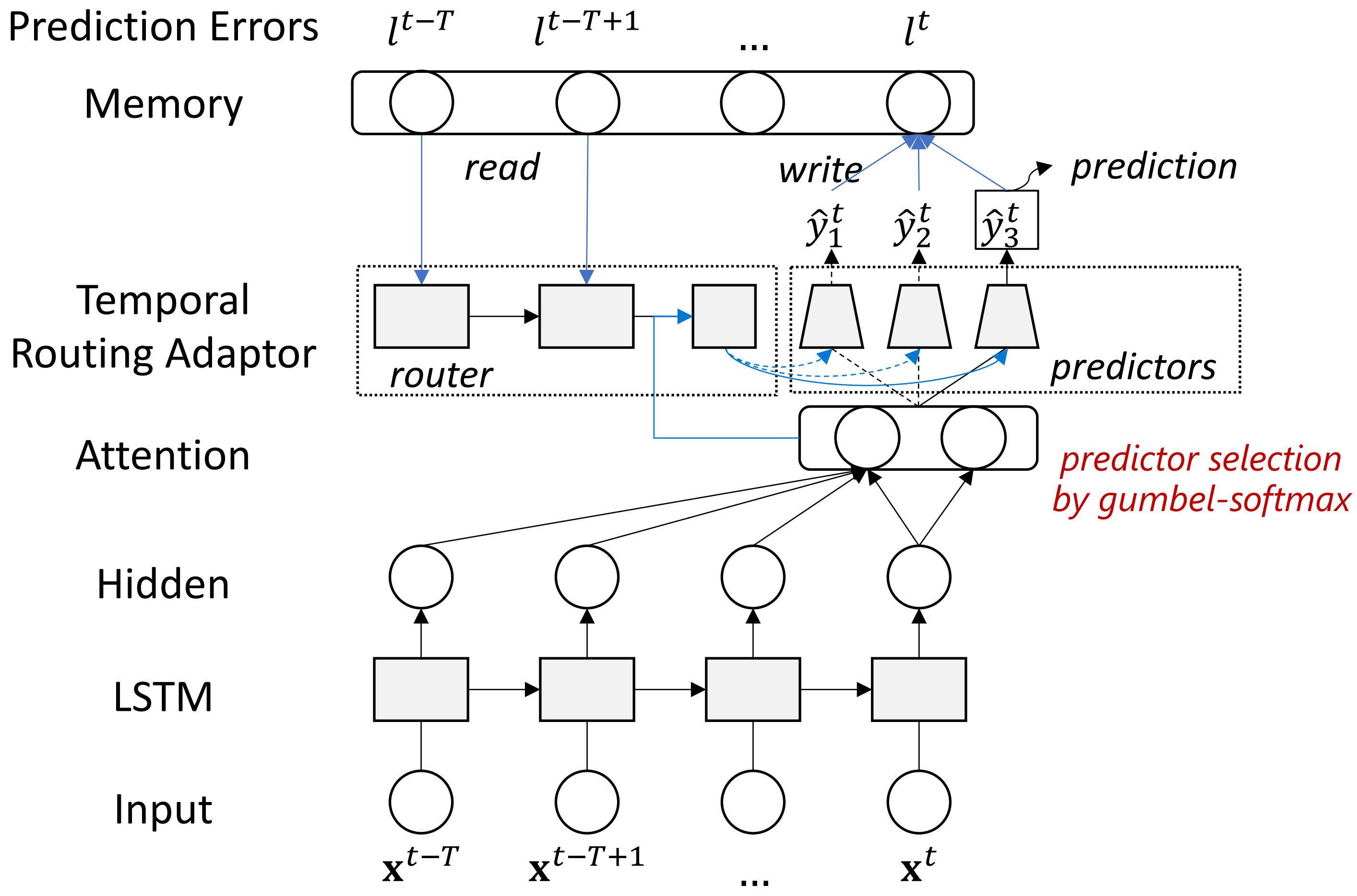




Methodology

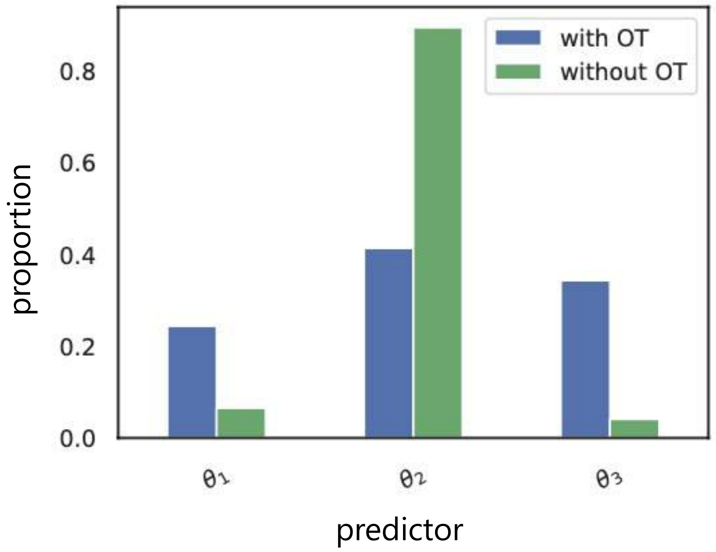
- Temporal Routing Adaptor (TRA) = predictors + router
  - An implementation of TRA on RNN:

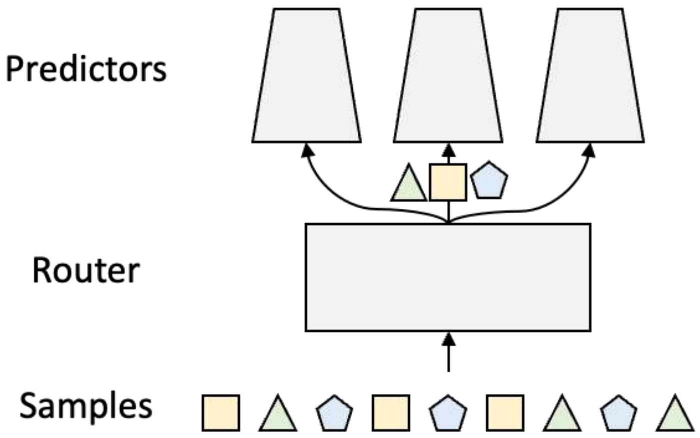




Methodology

- Optimal Transport (OT):
  - While TRA has the capacity to model multiple patterns, it can easily fall to some trivial solutions like selecting only one:





Methodology

- Optimal Transport (OT):
  - While TRA has the capacity to model multiple patterns, it can easily fall to some trivial solutions like selecting only one.
  - Introducing constraint to make assigned samples more balanced?

Methodology

- Optimal Transport (OT):
  - *+Objective*: assign samples to predictors to minimize the overall loss
  - *Constraint*: keep the assigned samples proportional to the relative share of the pattern

$$
\begin{aligned}
&\min_{\mathbf{P}} \langle \mathbf{P}, \mathbf{L} \rangle, \\
&s.t. \sum_{i=1}^{N} \mathbf{P}_{ik} = v_k * N, \ \forall k = 1 \ldots K \\
&\quad \sum_{k=1}^{K} \mathbf{P}_{ik} = 1, \ \forall i = 1 \ldots N \\
&\quad \mathrm{P}_{ik} \in \{0, 1\}, \ \forall i = 1 \ldots N, k = 1 \ldots K,
\end{aligned}
$$

where P_{ik} is the router assignment matrix, L is the cost matrix.

Methodology

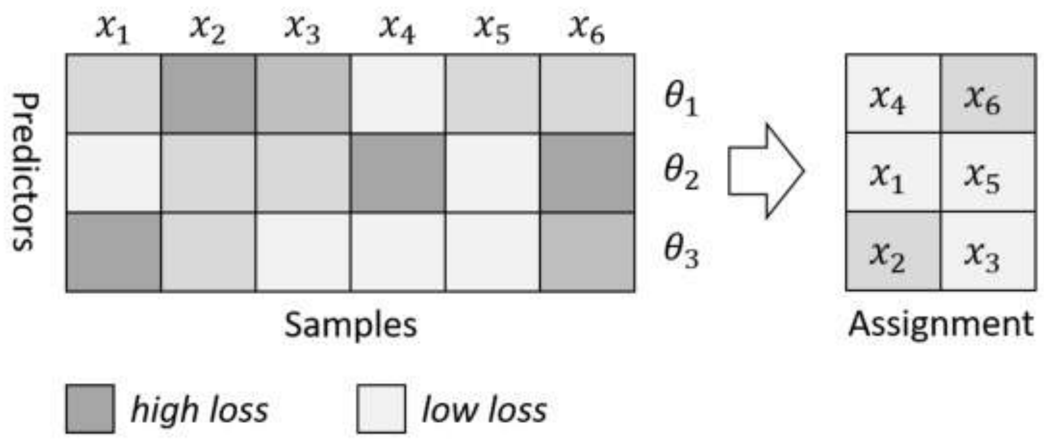
- Optimal Transport (OT):
  - *+Objective*: assign samples to predictors to minimize the overall loss
  - *Constraint*: keep the assigned samples proportional to the relative share of the pattern

$$\min_{\mathbf{P}} \langle \mathbf{P}, \mathbf{L} \rangle,$$

$$s.t. \sum_{i=1}^{N} \mathbf{P}_{ik} = v_k * N, \ \forall k = 1...K$$

$$\sum_{k=1}^{K} \mathbf{P}_{ik} = 1, \ \forall i = 1...N$$

$$\mathrm{P}_{ik} \in \{0, 1\}, \ \forall i = 1...N, k = 1...K,$$

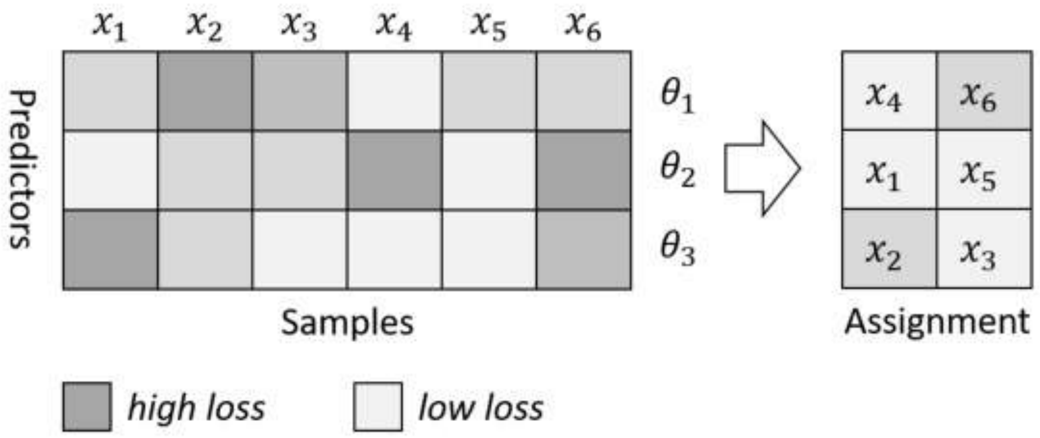


where P_{ik} is the router assignment matrix, L is the cost matrix.

Methodology

- Optimal Transport (OT):
  - *+Objective*: assign samples to predictors to minimize the overall loss
  - *Constraint*: keep the assigned samples proportional to the relative share of the pattern

$$
\begin{aligned}
&\min_{\mathbf{P}} \langle \mathbf{P}, \mathbf{L} \rangle, \\
&s.t. \sum_{i=1}^{N} \mathbf{P}_{ik} = \nu_k * N, \ \forall k = 1 \ldots K \\
&\sum_{k=1}^{K} \mathbf{P}_{ik} = 1, \ \forall i = 1 \ldots N \\
&\mathrm{P}_{ik} \in \{0, 1\}, \ \forall i = 1 \ldots N, k = 1 \ldots K,
\end{aligned}
$$



  - To emulate OT assignment during testing, we guide the learning of the router through adding an auxiliary regularization term into the objective:

$$
\min_{\Theta,\pi,\psi} \mathbb{E}_{(\mathbf{x}_i,\mathrm{y}_i)\in\mathcal{D}^{\text{train}}} \left[\ell(\mathbf{x}_i, \mathrm{y}_i; \Theta, \pi, \psi) \boxed{- \lambda \textstyle\sum_{k=1}^{K} \mathrm{P}_{ik} \log(q_{ik})} \right]
$$

Experiments

- Task: Stock Ranking Prediction
- Dataset: CSI800 Stocks on China A-share Market
- Metrics:
  - Regression metrics: MSE, MAE
  - Finance ranking metrics: Information Coefficients (IC), Information Ratio of IC (ICIR).
  - Portfolio metrics: Annual Return (AR), Annual Volatility (AV), Sharpe Ratio (SR), Max Drawdown
- Baselines:
  - Linear, LightGBM, MLP, SFM (KDD 17), ALSTM (IJCAI 17, IJCAI 19), Transformer (IJCAI 20)
  - Heuristic methods for learning multiple patterns: ALSTM+TS, Transformer+TS

* *Our dataset is publicly available at* https://github.com/microsoft/qlib

Experiments

- Heuristic methods for learning multiple patterns harm the model performance:

Method	Ranking Metrics				Portfolio Metrics			
	MSE (↓)	MAE (↓)	IC (↑)	ICIR (↑)	AR (↑)	AV (↓)	SR (↑)	MDD (↓)
Linear	0.163	0.327	0.020	0.132	-3.2%	16.8%	-0.191	32.1%
LightGBM	0.160(0.000)	0.323(0.000)	0.041	0.292	7.8%	15.5%	0.503	25.7%
MLP	0.160 (0.002)	0.323 (0.003)	0.037	0.273	3.7%	15.3%	0.264	26.2%
SFM	0.159 (0.001)	0.321 (0.001)	0.047	0.381	7.1%	14.3%	0.497	22.9%
ALSTM	0.158 (0.001)	0.320 (0.001)	0.053	0.419	12.3%	**13.7%**	0.897	**20.2%**
Trans.	0.158 (0.001)	0.322 (0.001)	0.051	0.400	14.5%	14.2%	1.028	22.5%
ALSTM+TS	0.160 (0.002)	0.321 (0.002)	0.039	0.291	6.7%	14.6%	0.480	22.3%
Trans.+TS	0.160 (0.004)	0.324 (0.005)	0.037	0.278	10.4%	14.7%	0.722	23.7%
ALSTM+TRA (Ours)	**0.157 (0.000)**	**0.318 (0.000)**	**0.059**	**0.460**	12.4%	14.0%	0.885	20.4%
Trans.+TRA (Ours)	**0.157 (0.000)**	0.320 (0.000)	0.056	0.442	**16.1%**	14.2%	**1.133**	23.1%

Experiments

- With both TRA and OT, our method can give consistent gains:

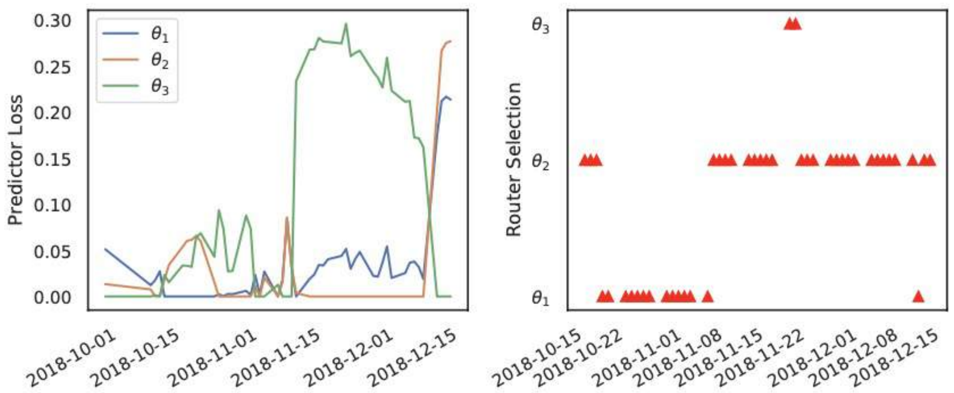
Method	Ranking Metrics				Portfolio Metrics			
	MSE (↓)	MAE (↓)	IC (↑)	ICIR (↑)	AR (↑)	AV (↓)	SR (↑)	MDD (↓)
Linear	0.163	0.327	0.020	0.132	-3.2%	16.8%	-0.191	32.1%
LightGBM	0.160(0.000)	0.323(0.000)	0.041	0.292	7.8%	15.5%	0.503	25.7%
MLP	0.160 (0.002)	0.323 (0.003)	0.037	0.273	3.7%	15.3%	0.264	26.2%
SFM	0.159 (0.001)	0.321 (0.001)	0.047	0.381	7.1%	14.3%	0.497	22.9%
ALSTM	0.158 (0.001)	0.320 (0.001)	0.053	0.419	12.3%	**13.7%**	0.897	**20.2%**
Trans.	0.158 (0.001)	0.322 (0.001)	0.051	0.400	14.5%	14.2%	1.028	22.5%
ALSTM+TS	0.160 (0.002)	0.321 (0.002)	0.039	0.291	6.7%	14.6%	0.480	22.3%
Trans.+TS	0.160 (0.004)	0.324 (0.005)	0.037	0.278	10.4%	14.7%	0.722	23.7%
ALSTM+TRA (Ours)	**0.157 (0.000)**	**0.318 (0.000)**	**0.059**	**0.460**	12.4%	14.0%	0.885	20.4%
Trans.+TRA (Ours)	**0.157 (0.000)**	0.320 (0.000)	0.056	0.442	**16.1%**	14.2%	**1.133**	23.1%

Incremental Analysis

- During test, TRA selects predictor with smaller loss:

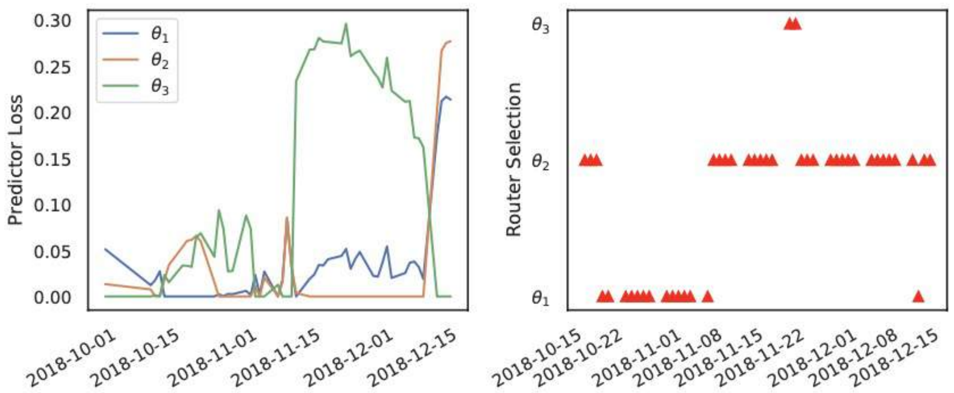
Incremental Analysis

- During test, TRA selects predictor with smaller loss:



Incremental Analysis

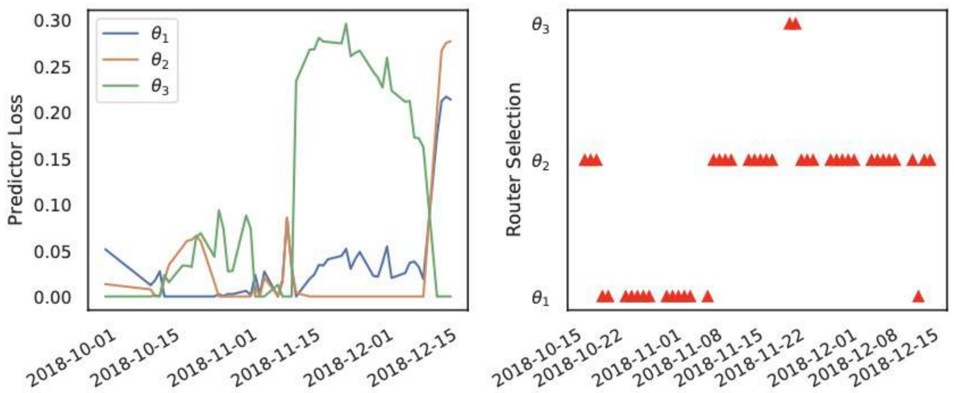
- During test, TRA selects predictor with smaller loss:



- TRA benefits most from using both latent representation and temporal prediction error as input:

Incremental Analysis

- During test, TRA selects predictor with smaller loss:



- TRA benefits most from using both latent representation and temporal prediction error as input:

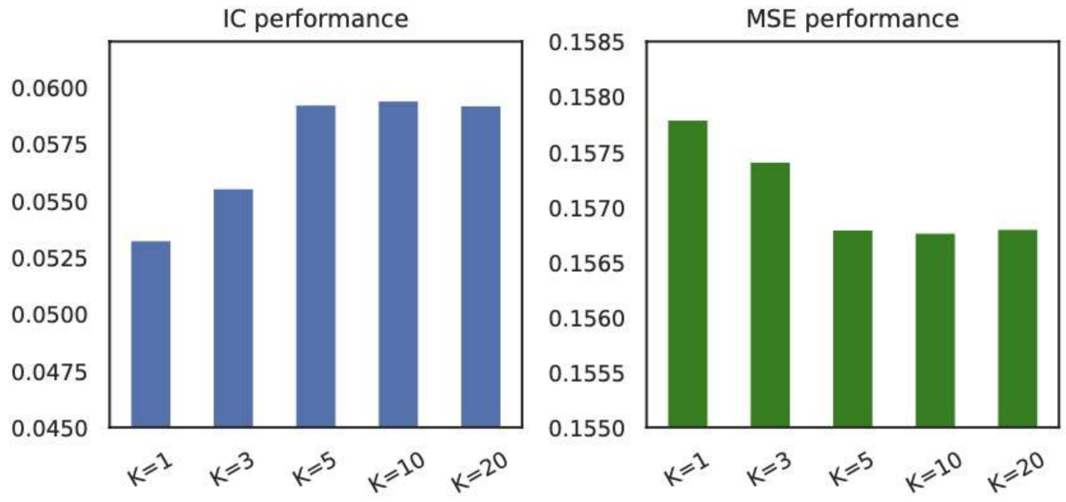
Information	MSE (↓)	MAE (↓)	IC (↑)	ICIR (↑)
Random	0.159 (0.001)	0.321 (0.002)	0.048	0.362
LR	0.158 (0.001)	0.320 (0.001)	0.053	0.409
TPE	0.158 (0.001)	0.321 (0.001)	0.049	0.381
LR+TPE	**0.157 (0.000)**	**0.318 (0.000)**	**0.059**	**0.460**

Incremental Analysis

- A moderate selection of # predictors (i.e., trading patterns) could give desirable performance gains:

Incremental Analysis

- A moderate selection of # predictors (i.e., trading patterns) could give desirable performance gains:



Thank you!

GitHub: https://github.com/microsoft/qlib
E-mail: HeLin23@gsb.columbia.edu
Zhou.Dong@microsoft.com