Personalized Prediction of Bounded-Rational Bargaining Behavior in Network Resource Sharing

Haoran Yu (presenter) and Fan Li

School of Computer Science & Technology
Beijing Institute of Technology

May 2024 @ IEEE INFOCOM

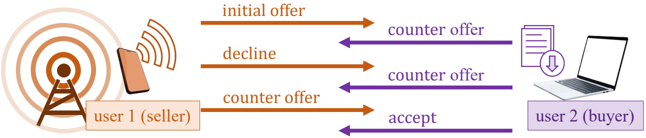
Background

Bargaining for Incentivizing Resource Sharing

- Bargaining is widely used to incentivize sharing of resources:
  - Internet access among mobile users: [G. Iosifidis et al. TON'17], [Y. Liu et al. TNSE'19]
  - Spectrum access among service providers: [H. Xu and B. Li TMC'12], [Q. Ni and C. Zarakovitis JSAC'11]
  - Network infrastructure among service providers: [L. Gao et al. JSAC'14], [H. Yu et al. TMC'16]

Bargaining for Incentivizing Resource Sharing

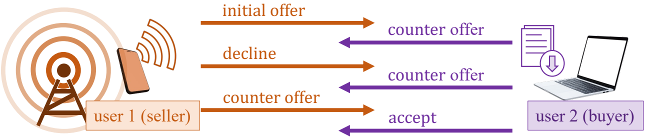
- Example of sharing Internet access



- Seller and buyer make decisions alternatively
- Decisions can be discrete ("A" or "D") or continuous (payment)
- An offer can be multi-dimensional (payment + speed)

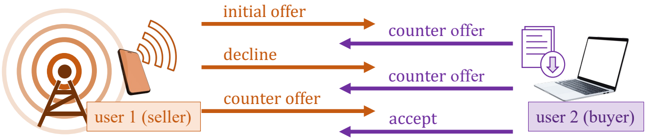
Bargaining for Incentivizing Resource Sharing

- Example of sharing Internet access



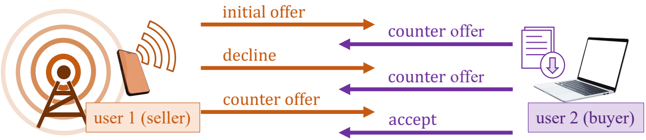
  - Seller and buyer make decisions alternatively
  - Decisions can be discrete (“A” or “D”) or continuous (payment)
  - An offer can be multi-dimensional (payment + speed)

Predicting Bargaining Behavior



- How to model bargaining behavior and predict the outcome?
- Most existing studies conducted game-theoretic analysis
  - Required strong informational and rationality assumptions
    - Assume the seller knows the buyer's gain from file downloading
    - Assume the buyer knows the seller's cost of sharing network
    - Assume their decisions maximize payoffs given information
  - Did not utilize real bargaining behavior data

Predicting Bargaining Behavior



- How to model bargaining behavior and predict the outcome?
- Most existing studies conducted game-theoretic analysis
  - Required strong informational and rationality assumptions
    - Assume the seller knows the buyer's gain from file downloading
    - Assume the buyer knows the seller's cost of sharing network
    - Assume their decisions maximize payoffs given information
  - Did not utilize real bargaining behavior data

Predicting Bargaining Behavior

- Q1: How to utilize data to predict bounded-rational bargaining behavior (including discrete and continuous decisions)?
- Q2: How to achieve a personalized behavior prediction that accounts for the heterogeneity of bargainers?
- We focus on predicting seller behavior in bilateral bargaining

Predicting Bargaining Behavior

- Q1: How to utilize data to predict bounded-rational bargaining behavior (including discrete and continuous decisions)?
- Q2: How to achieve a personalized behavior prediction that accounts for the heterogeneity of bargainers?
- We focus on predicting seller behavior in bilateral bargaining

Predicting Bargaining Behavior

- Q1: How to utilize data to predict bounded-rational bargaining behavior (including discrete and continuous decisions)?
- Q2: How to achieve a personalized behavior prediction that accounts for the heterogeneity of bargainers?
- We focus on predicting seller behavior in bilateral bargaining

Model

Bargaining Data

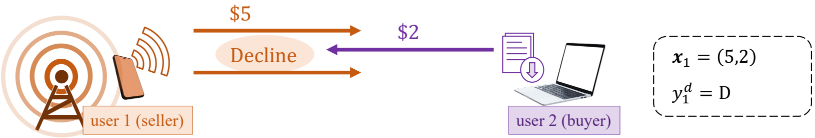
- Denote available training data as $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}}$
  - $\boldsymbol{x}_i$: history of the bargaining (between a seller and a buyer)
  - $\boldsymbol{y}_i = \left(y_i^d, \boldsymbol{y}_i^c\right)$: the seller's decision
    - Discrete decision y_i^d: "Accept", "Decline", or "Counter"
    - Continuous decision $\boldsymbol{y}_i^c$: offer details (e.g., payment and speed)
  - i: data point index

Bargaining Data

- Denote available training data as $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}}$
  - $\boldsymbol{x}_i$: history of the bargaining (between a seller and a buyer)
  - $\boldsymbol{y}_i = \left(y_i^d, \boldsymbol{y}_i^c\right)$: the seller's decision
    - Discrete decision y_i^d: "Accept", "Decline", or "Counter"
    - Continuous decision $\boldsymbol{y}_i^c$: offer details (e.g., payment and speed)
  - i: data point index

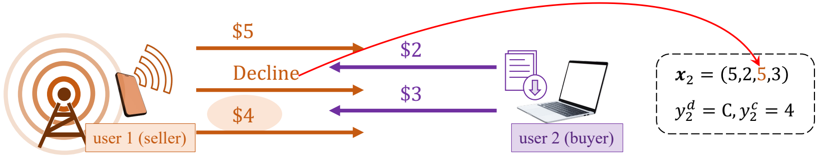
Bargaining Data

- Denote available training data as $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}}$
  - $\boldsymbol{x}_i$: history of the bargaining (between a seller and a buyer)
  - $\boldsymbol{y}_i = \left(y_i^d, \boldsymbol{y}_i^c\right)$: the seller's decision
    - Discrete decision y_i^d: "Accept", "Decline", or "Counter"
    - Continuous decision $\boldsymbol{y}_i^c$: offer details (e.g., payment and speed)
  - i: data point index



Bargaining Data

- Denote available training data as $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i\in\mathcal{I}}$
    - $\boldsymbol{x}_i$: history of the bargaining (between a seller and a buyer)
    - $\boldsymbol{y}_i = (y_i^d, \boldsymbol{y}_i^c)$: the seller's decision
        - Discrete decision y_i^d: "Accept", "Decline", or "Counter"
        - Continuous decision $\boldsymbol{y}_i^c$: offer details (e.g., payment and speed)
    - i: data point index



Personalized Behavior Prediction Problem

- Denote available training data as $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}}$
- We can split $\mathcal{I}$ into $\mathcal{I}_1, \ldots, \mathcal{I}_N$ according to the N sellers

Personalized Behavior Prediction Problem

Given $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}}$ and the information of $\mathcal{I}_1, \ldots, \mathcal{I}_N$, how to predict each seller n's future bargaining behavior?

Personalized Behavior Prediction Problem

- Denote available training data as $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i\in\mathcal{I}}$
- We can split $\mathcal{I}$ into $\mathcal{I}_1, \ldots, \mathcal{I}_N$ according to the N sellers

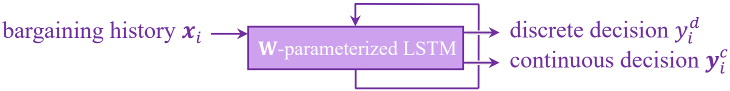
Personalized Behavior Prediction Problem

Given $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i\in\mathcal{I}}$ and the information of $\mathcal{I}_1, \ldots, \mathcal{I}_N$, how to predict each seller n's future bargaining behavior?

Solution

Behavior Prediction via Machine Learning

- We use a sequence model to learn the underlying pattern in seller behavior (which does not rely on rationality assumption)

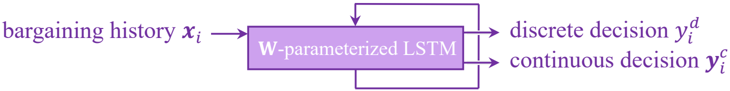


Sequence model can handle inputs with varying lengths

- We can train the LSTM to optimize $\boldsymbol{W}$ on data $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}}$
- How to utilize the information of $\mathcal{I}_1, \ldots, \mathcal{I}_N$?

Behavior Prediction via Machine Learning

- We use a sequence model to learn the underlying pattern in seller behavior (which does not rely on rationality assumption)

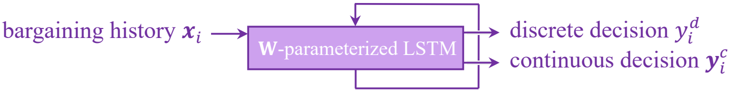


Sequence model can handle inputs with varying lengths

- We can train the LSTM to optimize $\boldsymbol{W}$ on data $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}}$
- How to utilize the information of $\mathcal{I}_1, \ldots, \mathcal{I}_N$?

Behavior Prediction via Machine Learning

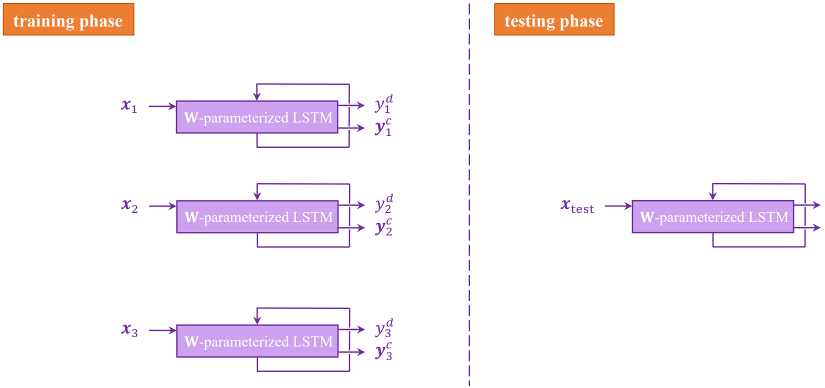
- We use a sequence model to learn the underlying pattern in seller behavior (which does not rely on rationality assumption)



Sequence model can handle inputs with varying lengths

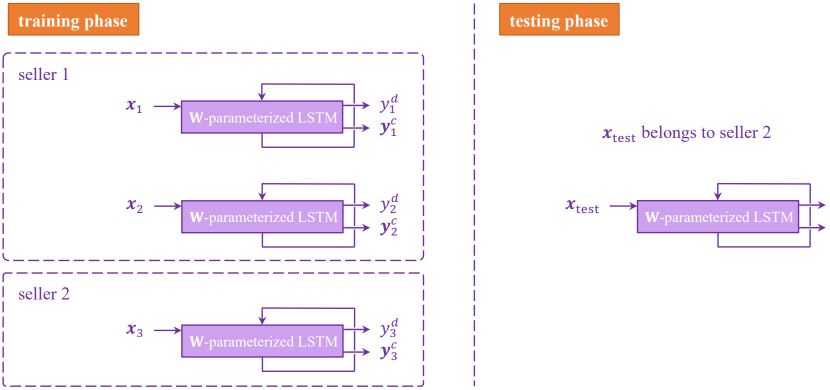
- We can train the LSTM to optimize $\boldsymbol{W}$ on data $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}}$
- How to utilize the information of $\mathcal{I}_1, \ldots, \mathcal{I}_N$?

Personalized Behavior Prediction



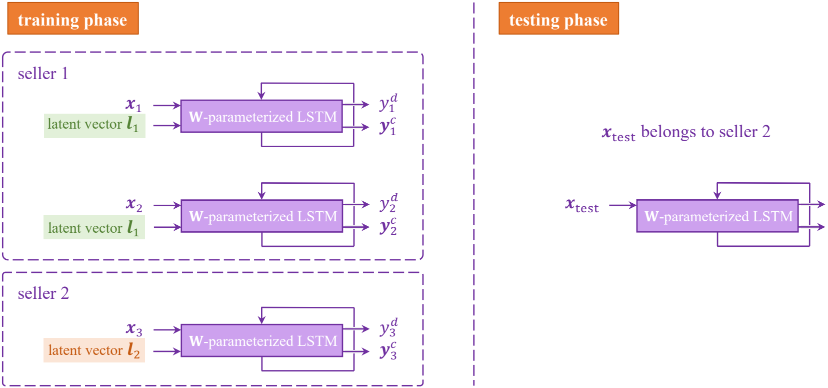
Standard behavior learning and prediction

Personalized Behavior Prediction



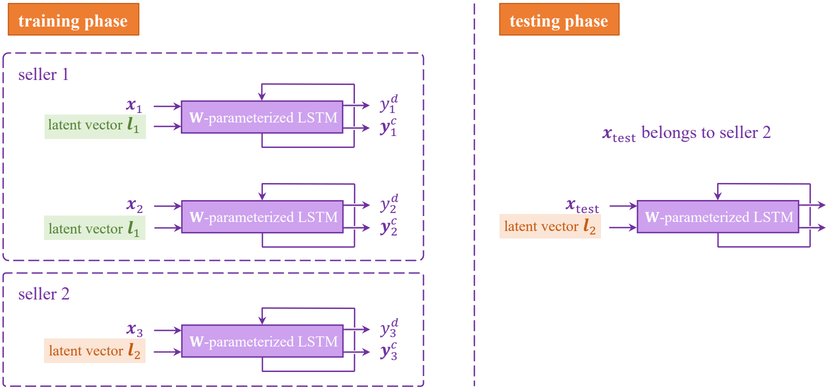
Personalized behavior learning and prediction

Personalized Behavior Prediction



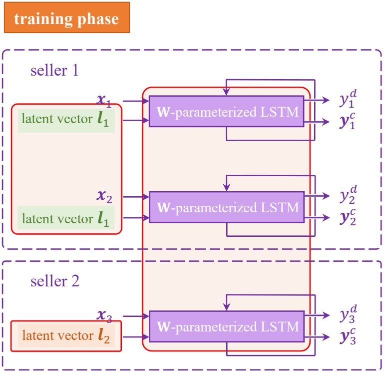
Define a latent vector l_n for each seller n to encode its decision-making preference

Personalized Behavior Prediction



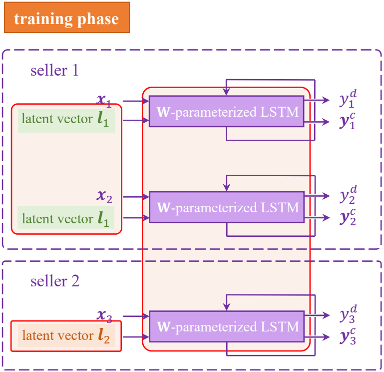
Use the trained LSTM and l_n to achieve a personalized prediction

Personalized Behavior Prediction



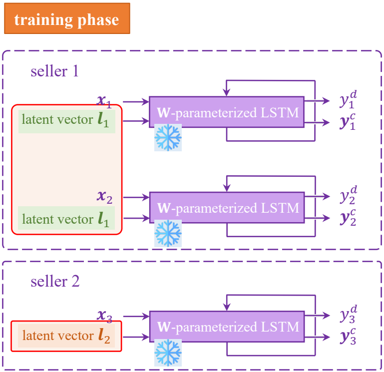
- How to learn both l_n and LSTM parameters W?
- Solution: Iteratively update l_n and W by maximizing data likelihood

Personalized Behavior Prediction



- How to learn both $\boldsymbol{l}_n$ and LSTM parameters $\boldsymbol{W}$?
- Solution: Iteratively update $\boldsymbol{l}_n$ and $\boldsymbol{W}$ by maximizing data likelihood

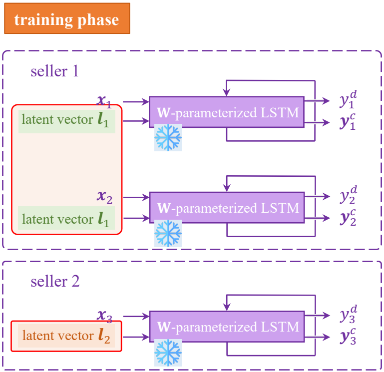
Iterative Updating of I_n and W



- This talk assumes prior distribution of I_n is a fixed uniform distribution
- In iteration k, we first update the posterior distribution of I_n:

$$\begin{aligned} &\Pr^k_{\text{post}}(I_n = j) \\ &= \Pr\left(I_n = j \middle| \{(x_i, y_i)\}_{i\in\mathcal{I}_n}; W^{k-1}\right) \\ &= \frac{\Pr\left(\{y_i\}_{i\in\mathcal{I}_n} \middle| I_n = j, \{x_i\}_{i\in\mathcal{I}_n}; W^{k-1}\right)}{\sum_{\hat{j}\in\mathcal{L}} \Pr\left(\{y_i\}_{i\in\mathcal{I}_n} \middle| I_n = \hat{j}, \{x_i\}_{i\in\mathcal{I}_n}; W^{k-1}\right)} \end{aligned}$$

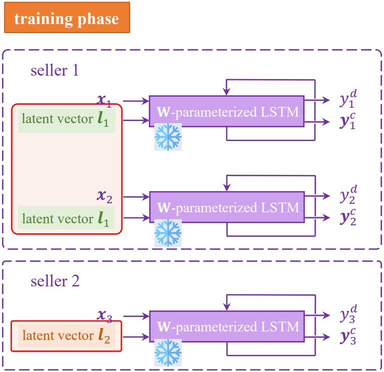
Iterative Updating of I_n and W



- This talk assumes prior distribution of $\boldsymbol{I}_n$ is a fixed uniform distribution
- In iteration k, we first update the posterior distribution of $\boldsymbol{I}_n$:

$$
\begin{aligned}
&\Pr^k_{\text{post}}(\boldsymbol{I}_n = \boldsymbol{j}) \\
&= \Pr\left(\boldsymbol{I}_n = \boldsymbol{j} \middle| \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}_n}; \boldsymbol{W}^{k-1}\right) \\
&= \frac{\Pr\left(\{\boldsymbol{y}_i\}_{i \in \mathcal{I}_n} \middle| \boldsymbol{I}_n = \boldsymbol{j}, \{\boldsymbol{x}_i\}_{i \in \mathcal{I}_n}; \boldsymbol{W}^{k-1}\right)}{\sum_{\hat{\boldsymbol{j}} \in \mathcal{L}} \Pr\left(\{\boldsymbol{y}_i\}_{i \in \mathcal{I}_n} \middle| \boldsymbol{I}_n = \hat{\boldsymbol{j}}, \{\boldsymbol{x}_i\}_{i \in \mathcal{I}_n}; \boldsymbol{W}^{k-1}\right)}
\end{aligned}
$$

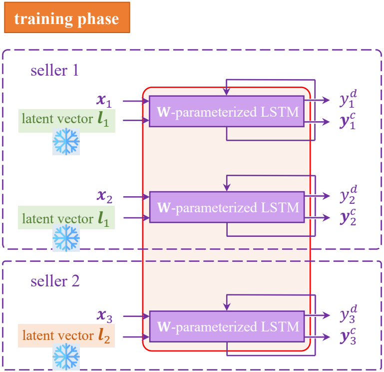
Iterative Updating of I_n and W



- This talk assumes prior distribution of $\boldsymbol{I}_n$ is a fixed uniform distribution
- In iteration k, we first update the posterior distribution of $\boldsymbol{I}_n$:

$$
\begin{aligned}
&\Pr^k_{\text{post}}\left(\boldsymbol{I}_n = \boldsymbol{j}\right) \\
&= \Pr\left(\boldsymbol{I}_n = \boldsymbol{j} \middle| \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i\in\mathcal{I}_n}; \boldsymbol{W}^{k-1}\right) \\
&= \frac{\Pr\left(\{\boldsymbol{y}_i\}_{i\in\mathcal{I}_n} \middle| \boldsymbol{I}_n = \boldsymbol{j}, \{\boldsymbol{x}_i\}_{i\in\mathcal{I}_n}; \boldsymbol{W}^{k-1}\right)}{\sum_{\hat{\boldsymbol{j}}\in\mathcal{L}} \Pr\left(\{\boldsymbol{y}_i\}_{i\in\mathcal{I}_n} \middle| \boldsymbol{I}_n = \hat{\boldsymbol{j}}, \{\boldsymbol{x}_i\}_{i\in\mathcal{I}_n}; \boldsymbol{W}^{k-1}\right)}
\end{aligned}
$$

Iterative Updating of l_n and W

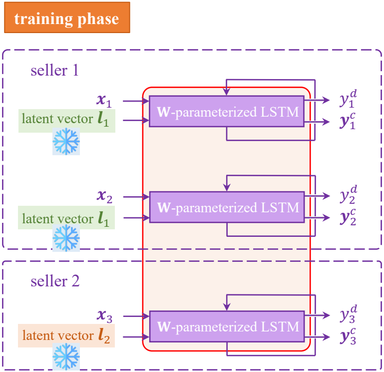


- In iteration k, we then update W^k:

$$\max_{W} \sum_n \mathbb{E}_{l_n} \left[\log \Pr \left(\{y_i\}_{i \in \mathcal{I}_n} \mid \{x_i\}_{i \in \mathcal{I}_n}, l_n; W \right) \right]$$

- We can approximately solve it by
  - sampling l_n according to posterior
  - training LSTM using data and l_n

Iterative Updating of l_n and W

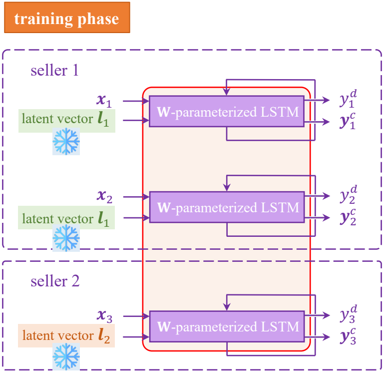


- In iteration k, we then update W^k:

$$\max_{W} \sum_{n} \mathbb{E}_{l_n} \left[\log \Pr \left(\{y_i\}_{i \in \mathcal{I}_n} \mid \{x_i\}_{i \in \mathcal{I}_n}, l_n; W \right) \right]$$

- We can approximately solve it by
  - sampling l_n according to posterior
  - training LSTM using data and l_n

Iterative Updating of I_n and W



- In iteration k, we then update $\boldsymbol{W}^k$:

$$\max_{\boldsymbol{W}} \sum_n \mathbb{E}_{\boldsymbol{I}_n} \left[\log \Pr \left(\{\boldsymbol{y}_i\}_{i \in \mathcal{I}_n} \mid \{\boldsymbol{x}_i\}_{i \in \mathcal{I}_n}, \boldsymbol{I}_n; \boldsymbol{W} \right) \right]$$

- We can approximately solve it by
  - sampling $\boldsymbol{I}_n$ according to posterior
  - training LSTM using data and $\boldsymbol{I}_n$

Iterative Updating of l_n and W

- Can this iterative updating process converge?

Main Proposition

If we optimally solve the expected log-likelihood maximization problem over $\boldsymbol{W}$ for each iteration, the iterative updating converges to a stationary point $\boldsymbol{W}^*$ of the data log-likelihood function, i.e.,

$$\nabla \log \Pr\left(\{\boldsymbol{y}_i\}_{i\in\mathcal{I}} \,\middle|\, \{\boldsymbol{x}_i\}_{i\in\mathcal{I}}; \boldsymbol{W}^*\right) = \mathbf{0}.$$

Iterative Updating of I_n and W

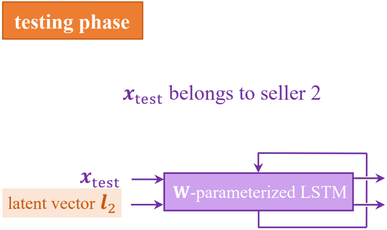
- Can this iterative updating process converge?

Main Proposition

If we optimally solve the expected log-likelihood maximization problem over $\boldsymbol{W}$ for each iteration, the iterative updating converges to a stationary point $\boldsymbol{W}^*$ of the data log-likelihood function, i.e.,

$$\nabla \log \Pr\left(\{\boldsymbol{y}_i\}_{i\in\mathcal{I}} \,\middle|\, \{\boldsymbol{x}_i\}_{i\in\mathcal{I}}\,; \boldsymbol{W}^*\right) = \boldsymbol{0}.$$

Iterative Updating of $\boldsymbol{l}_n$ and $\boldsymbol{W}$



- Predict $\boldsymbol{y}_{\text{test}}$ using the learned posterior distribution of $\boldsymbol{l}_n$ and $\boldsymbol{W}$:

$$\mathbb{E}_{\boldsymbol{l}_n}\left[\Pr\left(\boldsymbol{y}_{\text{test}} \middle| \boldsymbol{x}_{\text{test}}, \boldsymbol{l}_n; \boldsymbol{W}\right)\right]$$

Experiments

Experimental Settings

- eBay dataset: $240,000+$ data points covering 6,000 sellers
- Our Methods
  - Personalized Behavior Prediction (PBP)
  - Fast Personalized Behavior Prediction (FastPBP)
- Baselines
  - LSTM+FineTuning
    - Use all $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i \in \mathcal{I}}$ to train an LSTM
    - Fine-tune LSTM on $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i \in \mathcal{I}_n}$ to predict seller n's behavior
  - Clustering+LSTM
    - Partition sellers into different clusters based on their similarities
    - Train the LSTM for each cluster separately

Experimental Settings

- eBay dataset: $240,000+$ data points covering 6,000 sellers
- Our Methods
  - Personalized Behavior Prediction (PBP)
  - Fast Personalized Behavior Prediction (FastPBP)
- Baselines
  - LSTM+FineTuning
    - Use all $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i\in\mathcal{I}}$ to train an LSTM
    - Fine-tune LSTM on $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i\in\mathcal{I}_n}$ to predict seller n's behavior
  - Clustering+LSTM
    - Partition sellers into different clusters based on their similarities
    - Train the LSTM for each cluster separately

Experimental Settings

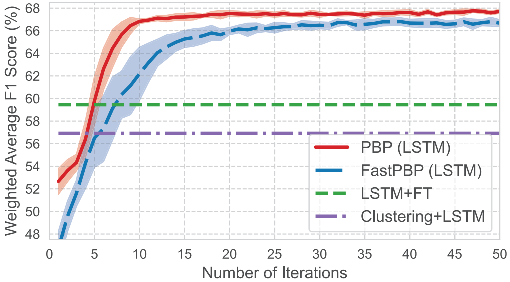
- eBay dataset: $240,000+$ data points covering 6,000 sellers
- Our Methods
  - Personalized Behavior Prediction (PBP)
  - Fast Personalized Behavior Prediction (FastPBP)
- Baselines
  - LSTM+FineTuning
    - Use all $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}}$ to train an LSTM
    - Fine-tune LSTM on $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}_n}$ to predict seller n's behavior
  - Clustering+LSTM
    - Partition sellers into different clusters based on their similarities
    - Train the LSTM for each cluster separately

Experimental Settings

- eBay dataset: $240,000+$ data points covering 6,000 sellers
- Our Methods
  - Personalized Behavior Prediction (PBP)
  - Fast Personalized Behavior Prediction (FastPBP)
- Baselines
  - LSTM+FineTuning
    - Use all $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}}$ to train an LSTM
    - Fine-tune LSTM on $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i \in \mathcal{I}_n}$ to predict seller n's behavior
  - Clustering+LSTM
    - Partition sellers into different clusters based on their similarities
    - Train the LSTM for each cluster separately

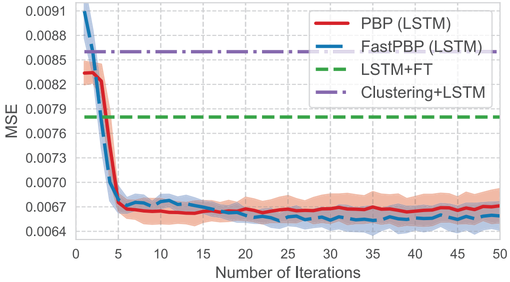
Experimental Results

- Comparison in predicting $\boldsymbol{y}^d$



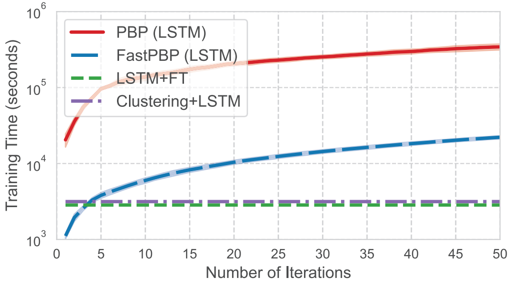
Experimental Results

- Comparison in predicting $\boldsymbol{y}^c$



Experimental Results

- Comparison in training time



Conclusion

- We proposed methods for personalized prediction of bounded rational bargaining behavior in network resource sharing
- Some contents are not covered:
  - Iteratively learn prior distribution of latent vector $\boldsymbol{I}_n$
  - Accelerate the iterations via sampling and early termination
  - Consider continuous latent vector $\boldsymbol{I}_n$

Conclusion

- We proposed methods for personalized prediction of bounded rational bargaining behavior in network resource sharing
- Some contents are not covered:
  - Iteratively learn prior distribution of latent vector $\boldsymbol{I}_n$
  - Accelerate the iterations via sampling and early termination
  - Consider continuous latent vector $\boldsymbol{I}_n$

THANK YOU