Collective Learning and Model Fusion in Large Multi-Agent Systems

Bryan Kian Hsiang Low[3] *in collaboration with*
Trong Nghia Hoang[1], Quang Minh Hoang[2], Carl Kingsford[2], and Jonathan How[4]

[1]MIT-IBM Watson AI Lab, IBM Research
[2]Department of Computer Science, Carnegie Mellon University (CMU)
[3]Department of Computer Science, National University of Singapore (NUS)
[4]Laboratory for Information and Decision Systems, MIT

MIT-IBM
Watson AI Lab

Carnegie Mellon University

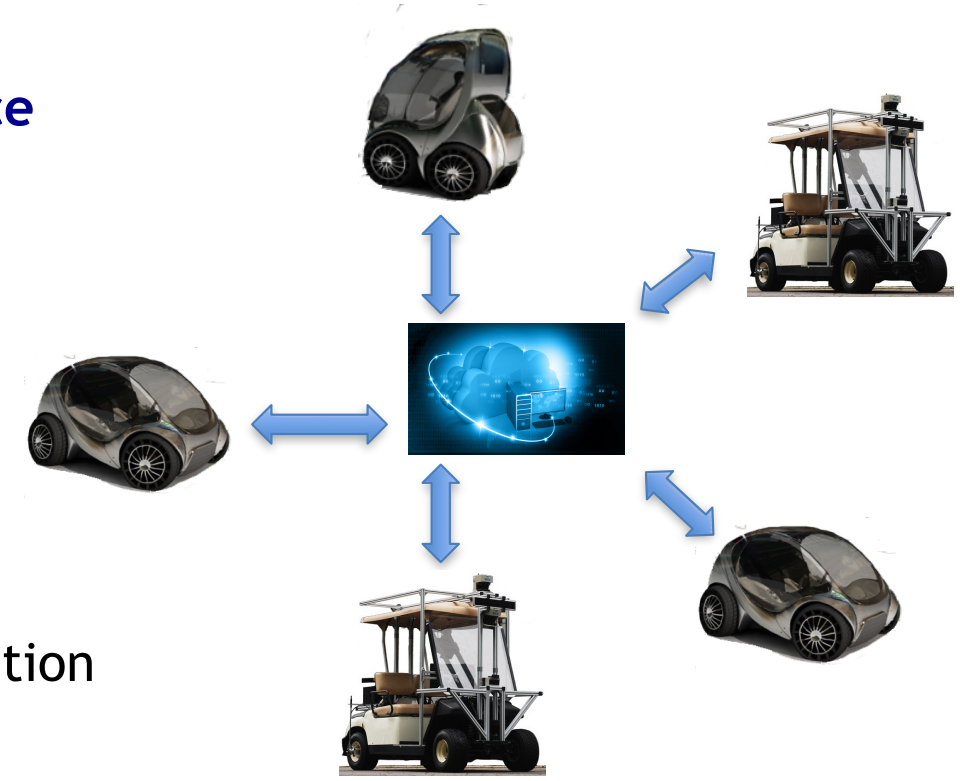
Collective Learning: Motivation

Distributed Learning [13]: Cloud Intelligence

+ Local workers: upload local statistics
+ Central server: broadcast intelligence

Limitations:

+ Choke point of operation failure
+ Communication bottlenecks on server
+ High latency due to centralized communication



Cloud Intelligence Architecture

Cloud Intelligence: Centralized risk of **operational failure** + communication & computational **bottlenecks** imposed by the central server

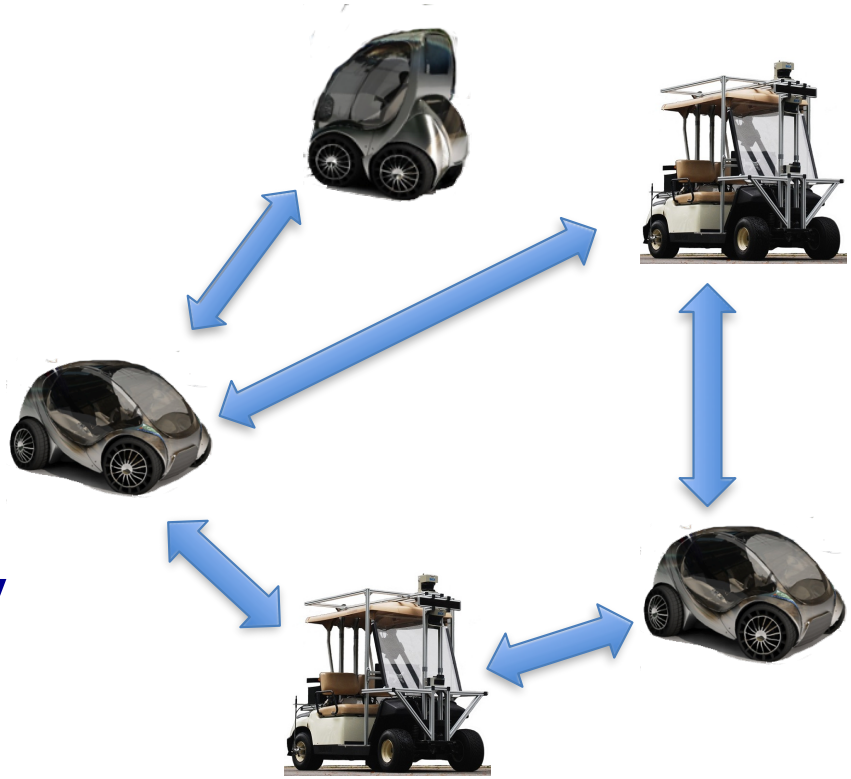
Collective Learning: Motivation

Collective Learning: Edge Intelligence

+ Edge devices engineer **local intelligence**
  no choke point

+ Intelligence fusion via **local communication**
  reduced latency

+ Collective computation via **mass productivity**
  and local **message exchanges**
  preserved predictive quality



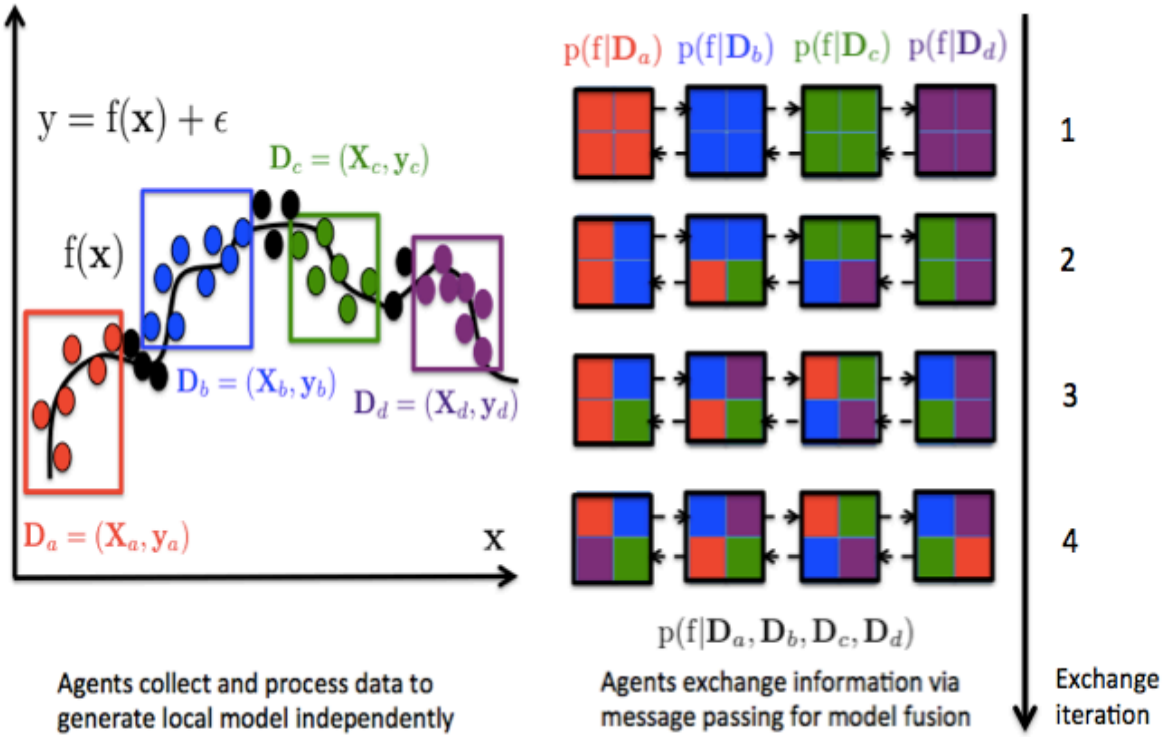
Edge Intelligence Architecture

Edge Intelligence: **Mass productivity of edge devices** for computation and **preserved predictive quality** through **on-demand fusion of knowledge**

Collective Learning: Issues

Model representation scale poorly with increasing data **(expensive communications)**

Merging model requires sharing data & re-training on shared data **(inefficient computations)**



Incomplete information: Agents only communicate with local neighbors (vulnerable to **system changes**)

Agents collect local data of different behaviors **(cross-domain communications)**

Collective Learning: Research Objectives

Objectives: Develop collective learning framework that

- Generates **local** models independently
- Encodes these models into **effective representations** for model fusion
- Assembles **global** models accurately via **distributed communication**

Challenges:

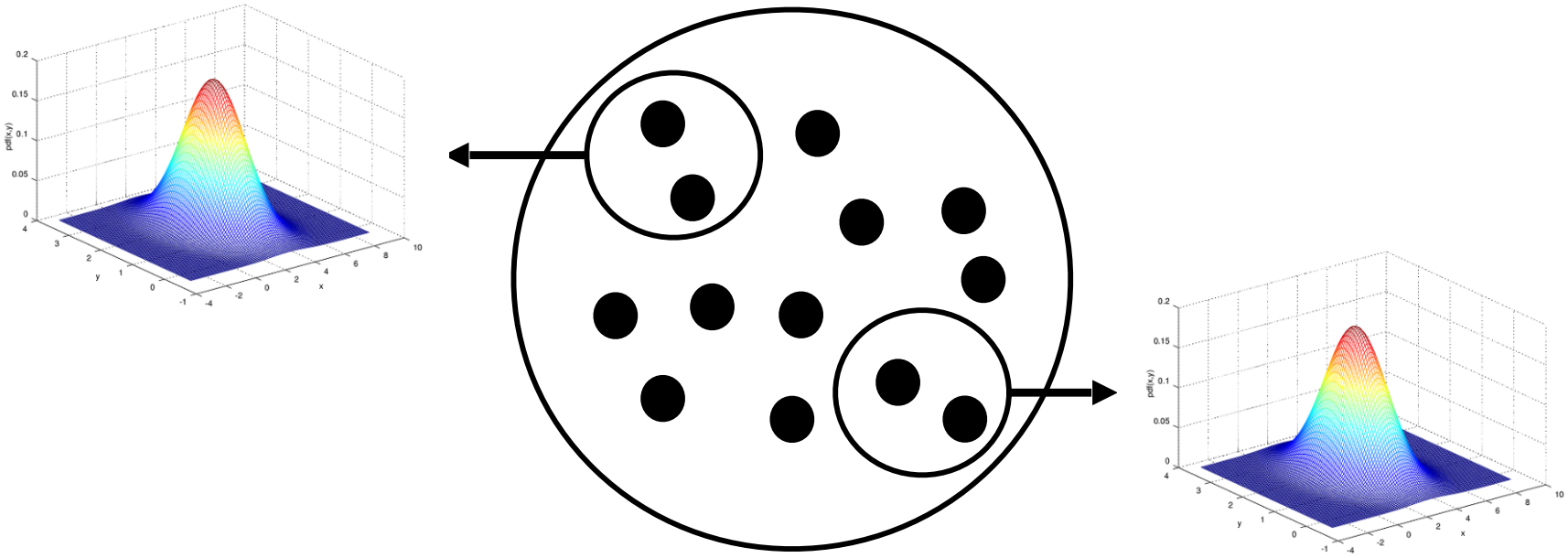
- Limited **computation**: Inefficient to re-train model when new data arrives
- Limited **communication**: Exchanging data directly is undesirable
- **Distributed** communication: **No central server** to coordinate

Research goals – Develop:

- Representation amenable to **online update** with streaming data
- **Communication-efficient** operator for **model fusion**
- **Peer-to-peer message passing algorithm** for decentralized fusion

Gaussian Process

f(.) is distributed by a **Gaussian process (GP)** if its evaluations at **any finite subset of inputs** are distributed by a **multivariate Gaussian distribution**



GP allows analytic inference ☺

but incurs cubic processing cost ☹

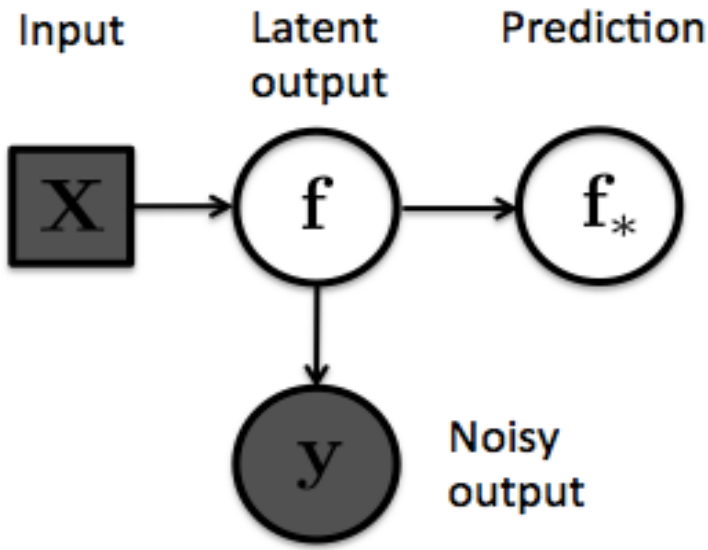
Local Model via Gaussian Process

Objective: Develop efficient local model representation for online update with data $\mathbf{D} = (\mathbf{X}, \mathbf{y})$

Gaussian process (GP) – expressive representation with **"streaming" prior**

However, not efficient

- Computation: **cubic**
- Representation: **quadratic**
- Update: **cubic**
- Communication: **not possible** across different domains



$p(\mathbf{f}_*|\mathbf{D})$ incurs $O(|\mathbf{D}|^3)$ computation

$O(|\mathbf{D}|^2)$ memory

Idea: Exploit **sparse encodings** of GP representation [2-5, 16]

Local Model via Sparse Gaussian Process

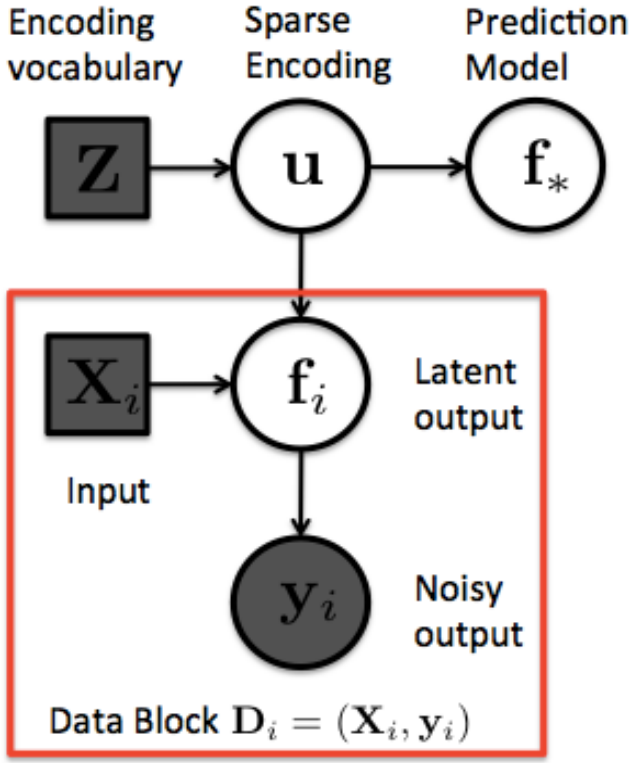
Objective: Develop efficient local model representation for online update with **streaming** data $\mathbf{D} = \{\mathbf{D}_i\}$ with $\mathbf{D}_i = \{\mathbf{X}_i, \mathbf{y}_i\}$

Solution: Exploit sparse encoding $\mathbf{u} = u(\mathbf{Z})$ where u(.) is distributed by a **parameter-free** GP [16] – the resulting model can:

Encode local model into a **common** structure by learning a transformation from u(.) to f(.)

Separate effect of each data block on predictive model: $\mathbf{D}_i$'s are independent given **u**

Generate additive model summary that allows **online update** – unlike GPs [1-5, 16]



$p(\mathbf{u}|\mathbf{D})$ incurs $O(|\mathbf{D}_i||\mathbf{Z}|^2)$ update

$O(|\mathbf{Z}|^2)$ memory

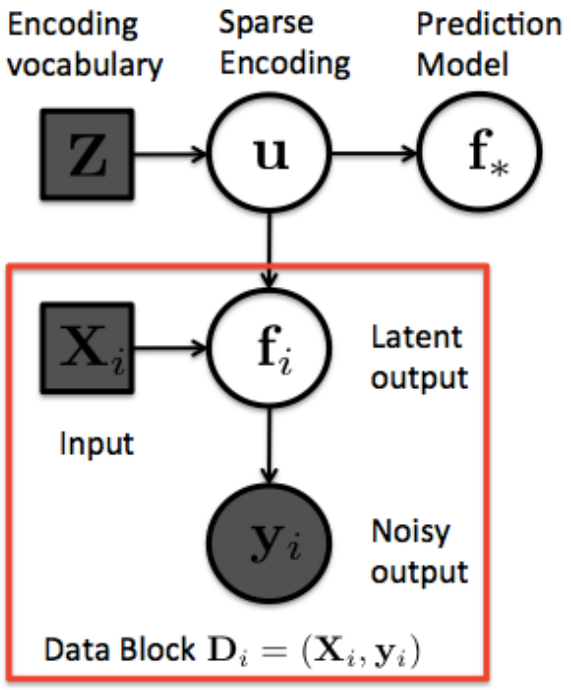
Local Model via Sparse Gaussian Process

Objective: Develop efficient local model representation for online update with **streaming** data $\mathbf{D} = \{\mathbf{D}_i\}$ with $\mathbf{D}_i = \{\mathbf{X}_i, \mathbf{y}_i\}$

Advantages over existing GPs [1-5, 16]:

Online update cost scales **linearly** in size of data block $|\mathbf{D}_i|$

Novel communication mechanism using $\mathrm{p}(\mathbf{u}|\mathbf{D})$ across agents operating in different (but correlated) domains



$\mathrm{p}(\mathbf{u}|\mathbf{D})$ incurs $\mathrm{O}(|\mathbf{D}_i||\mathbf{Z}|^2)$ update

$\mathrm{O}(|\mathbf{Z}|^2)$ memory

Local Model via Sparse Gaussian Process

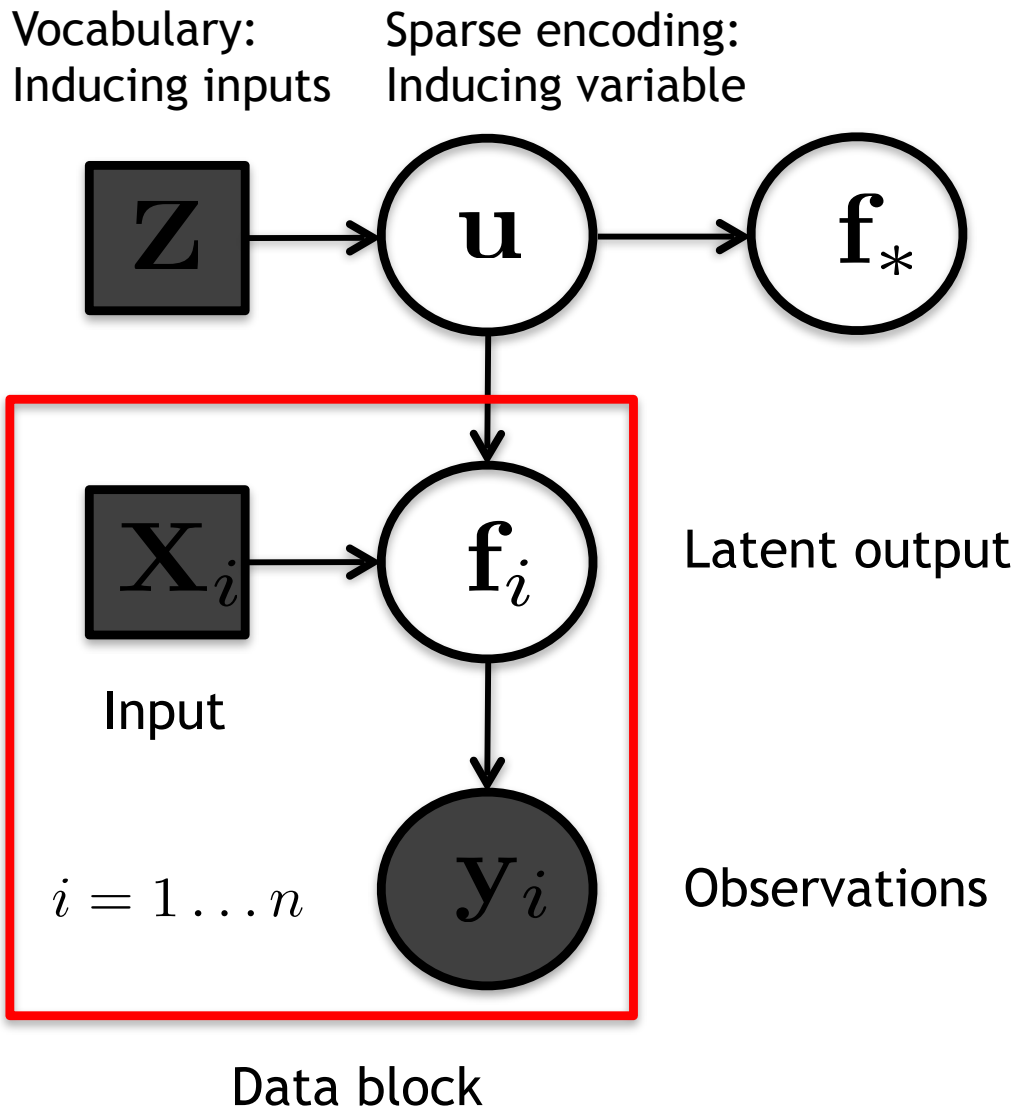
Embedding Domain Knowledge:
Prior Belief

$$\mathbf{u} \sim \mathbb{N}(\mathbf{m}_0, \mathbf{V}_0)$$

Local learning

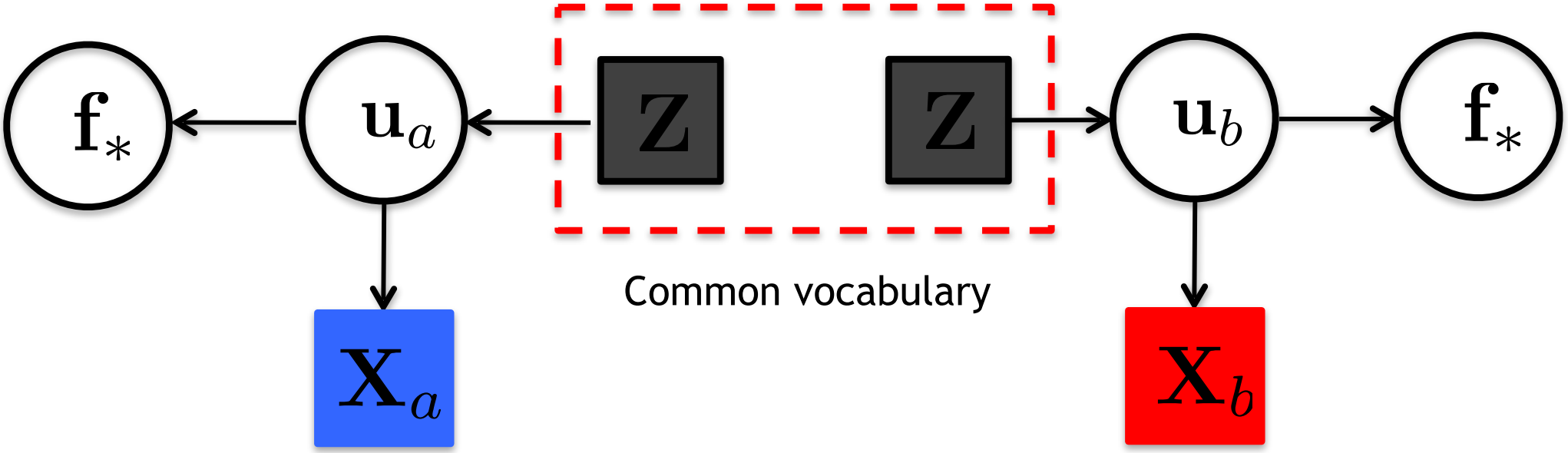
Local Model:
Posterior Belief

$$\mathbf{u} \sim \mathbb{N}(\mathbf{m}_n, \mathbf{V}_n)$$



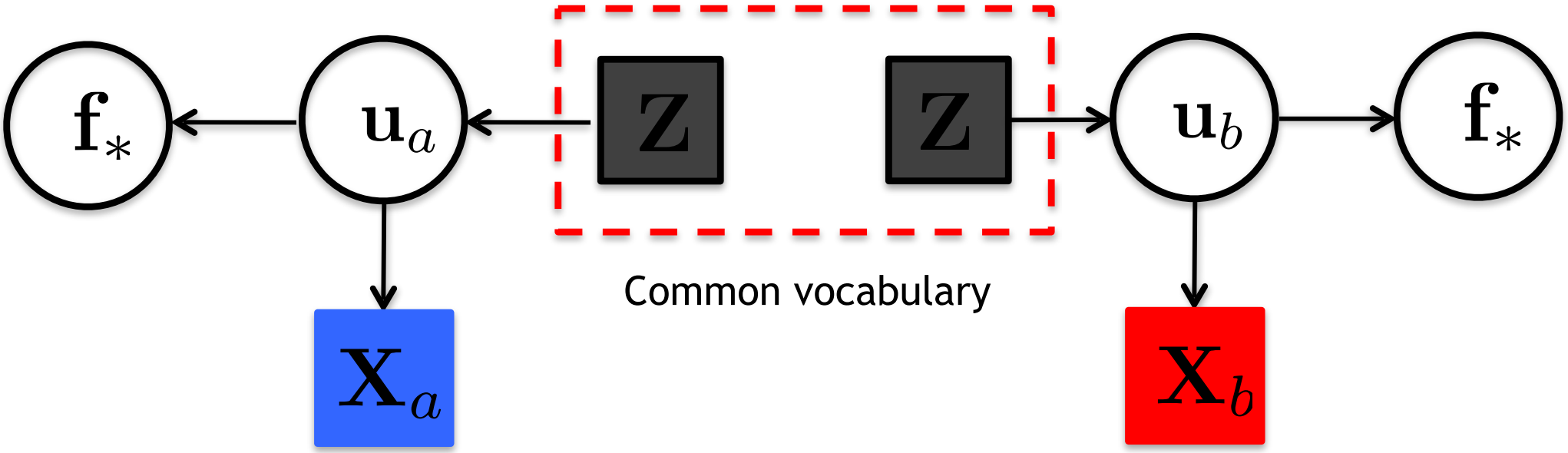
Cost-Efficient Model Fusion

Objective: Develop **cost-efficient** operator that enables agents to share models without communicating raw data (costly)



$$\mathbf{r}_a \triangleq \begin{bmatrix} \mathbf{V}_a^{-1}\mathbf{m}_a \\ \mathbf{V}_a^{-1} \end{bmatrix} \qquad \mathbf{r}_b \triangleq \begin{bmatrix} \mathbf{V}_b^{-1}\mathbf{m}_b \\ \mathbf{V}_b^{-1} \end{bmatrix}$$

Cost-Efficient Model Fusion



$$\mathbf{r}_{ab} = \mathbf{r}_a + \mathbf{r}_b - \mathbf{r}_0$$

fused representation

agent a's and b's local representation

prior, common representation

Benefit: Fusion cost does not depend on data size → **computation- & communication-efficient**

Collective Model Fusion via Decentralized Message Passing

Objective: Develop **peer-to-peer** message passing algorithm to achieve global fusion **without** communicating via **central server**

Solution: Exploit **additive structure** of fusion operator to construct global representation from **distributed, local messages** combining local representations from different agents' neighborhoods

Message content: $\mathbf{m}_{i\to j}^{(t+1)}$ sent from agent i to j at time t + 1 encapsulates local representations in i's local neighborhood $\mathbb{A}(i) \setminus \{j\}$

$$\mathbf{m}_{i\to j}^{(t+1)} = \left(\sum_{k\in\mathbb{A}(i)\setminus\{j\}} \mathbf{m}_{k\to i}^{(t)} \right) + \mathbf{r}_{a_i} - \mathbf{r}_0$$

Collective Model Fusion via Decentralized Message Passing

Objective: Develop **peer-to-peer** message passing algorithm to achieve global fusion **without** communicating via **central server**

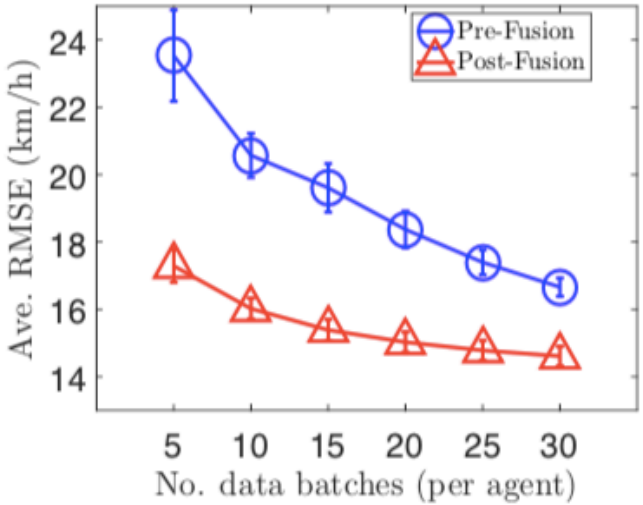
Solution: Exploit **additive structure** of fusion operator to construct global representation from **distributed, local messages** combining local representations from different agents' neighborhoods

Key result: Global model representation $\mathbf{r}$ can be constructed after convergence at $t \geq d$ (d denotes network diameter)

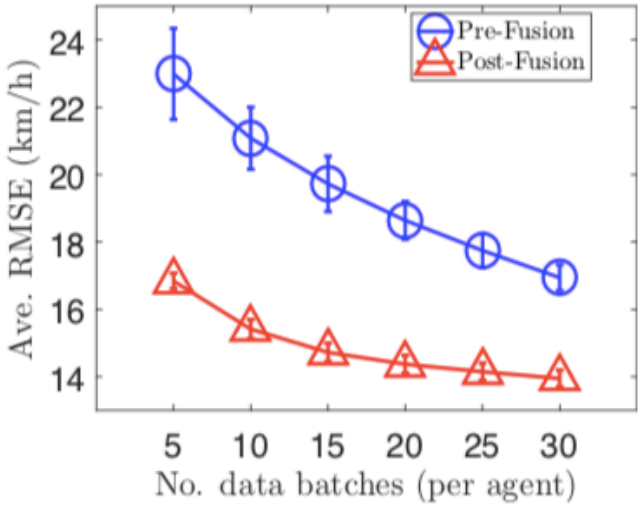
$$\mathbf{r} = \left(\sum_{k \in \mathbb{A}(i)} \mathbf{m}_{k \to i}^{(d)} \right) + \mathbf{r}_0$$

Empirical Studies: Performance Gain

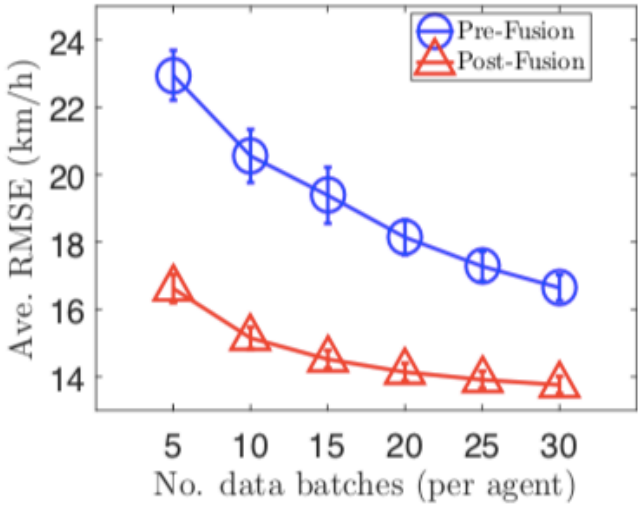
Traffic dataset (5D input) [13] features a traffic phenomenon over an urban road network with 775 road segments. 10K batches/blocks of data are streamed in random order to 100 agents. Each agent is evaluated on a separate set of 2K data points.



(a) $|\mathbf{Z}| = 50$



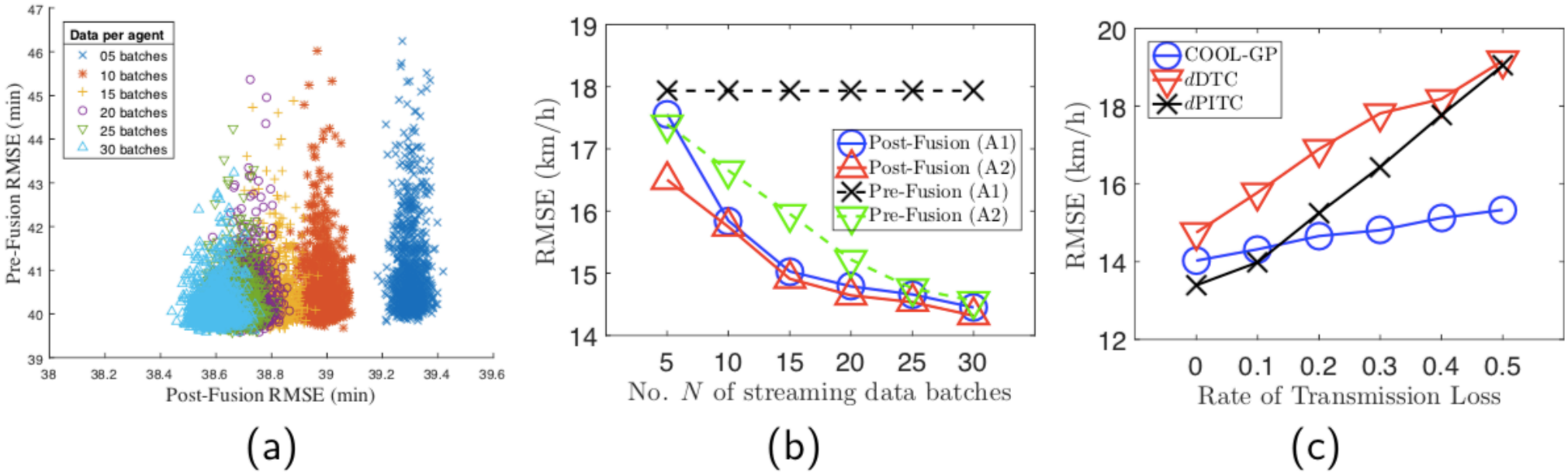
(b) $|\mathbf{Z}| = 100$



(c) $|\mathbf{Z}| = 200$

Observations: (I) Post-fusion prediction shows significant performance gain, and (II) performance gap reduces with more data **(diminishing gain)**

Empirical Studies: Stability

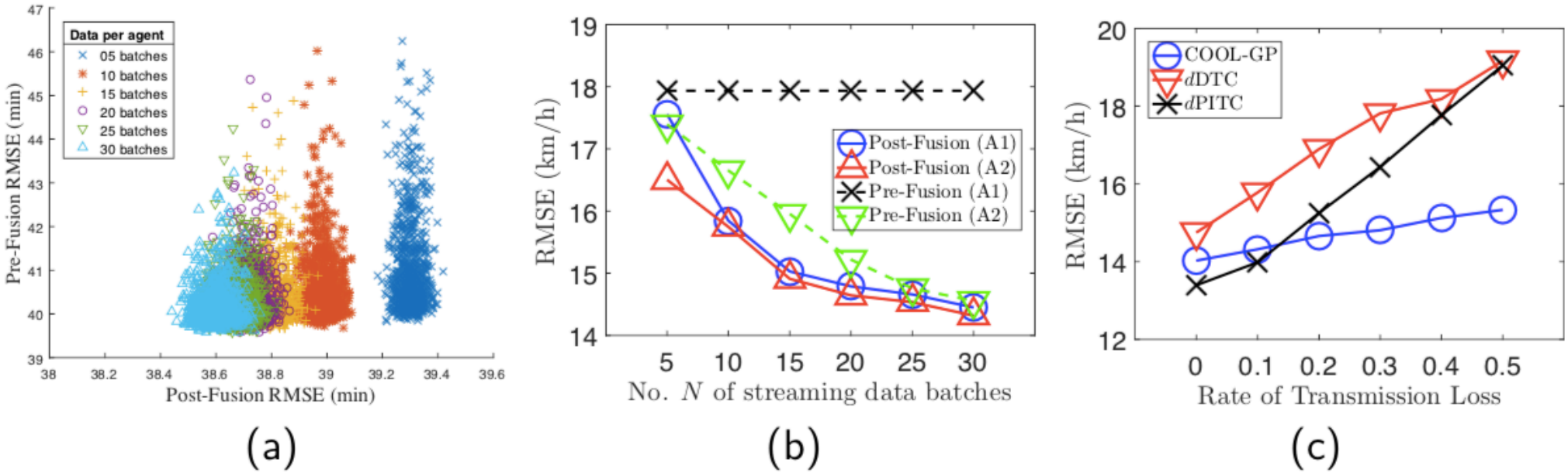


(a) Individual performance profiles (pre- vs. post-fusion RMSE) of a 1000-agent system evaluated in the data-intensive AIRLINE domain [6, 7]

Observations: (I) Clusters of performance profiles shift towards regions with better pre- and post-fusion accuracy with more data, (II) shift slows down with more data, (III) high variance for pre-fusion and low variance for post-fusion

→ **stability:** post-fusion consensus with small variation

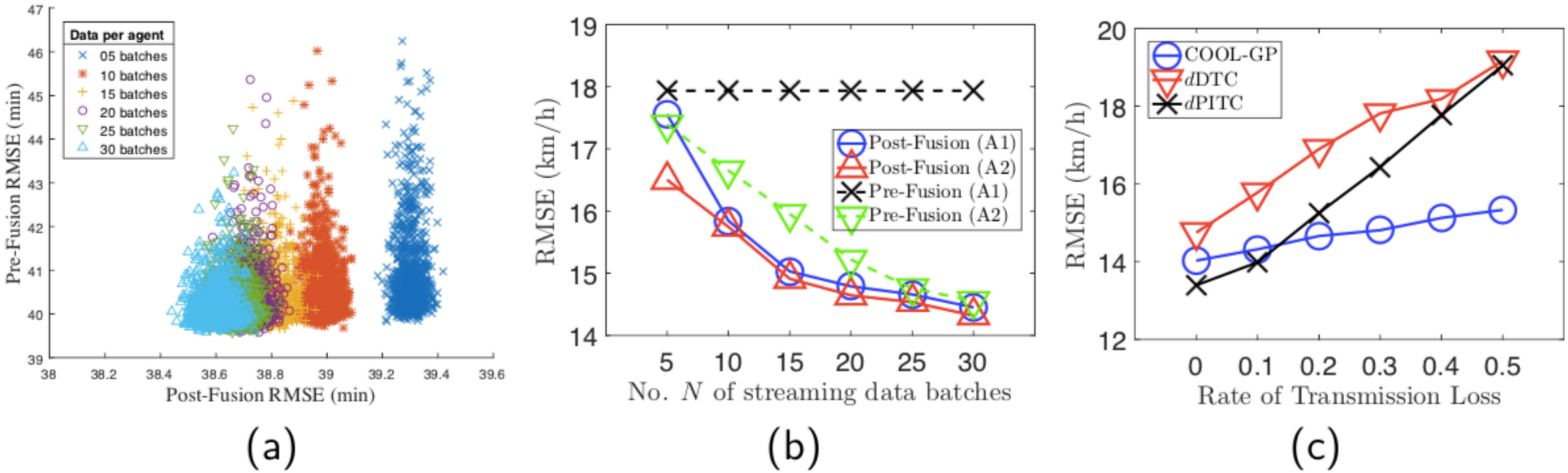
Empirical Studies: Resiliency



(b) Pre- and post-fusion performance of 2 agents with different learning capabilities: A1 with fixed amount of data & A2 with continuous supply of data

Observations: (I) Without fusion, A1 cannot improve its performance, (II) with fusion: A1 performs similar to A2 and far exceeds its original accuracy, (III) A2 marginally improves upon fusion with A1 → fusion benefits agents with lesser capabilities & is **resilient** to information disparity

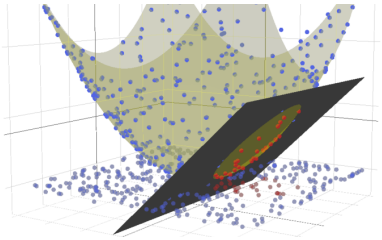
Empirical Studies: Robustness

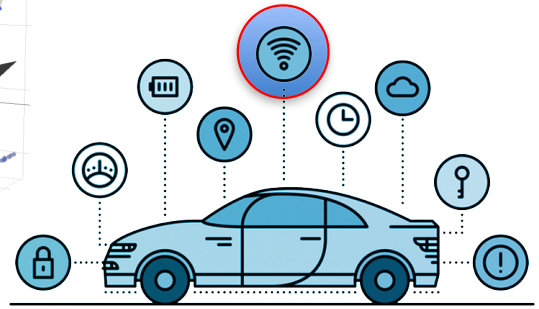


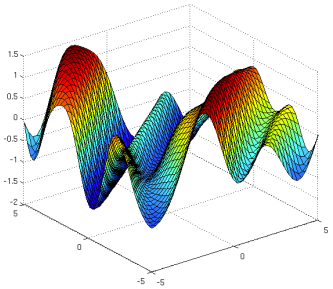
(c) Post-fusion performance of our COOL-GP compared to existing distributed GPs (dDTC [9] & dPITC [13]) vs. rate of transmission loss in traffic domain

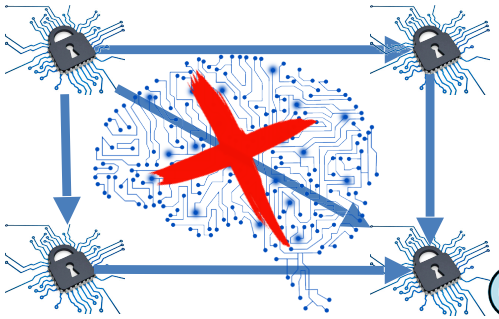
Observations: With higher transmission loss, post-fusion performance of our COOL-GP degrades more gracefully. **Reason:** dDTC & dPITC require all agents to transmit messages to a master server & failing to achieve this leads to irrecoverable information loss, while COOL-GP allows agents to propagate messages to multiple neighbors & lower risk of losing information -- more **robust** -- for more detail: https://arxiv.org/abs/1805.09266

Model Fusion with Black Boxes

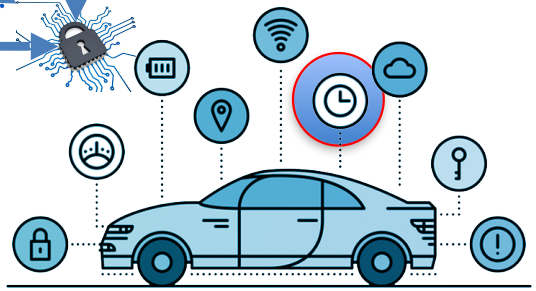


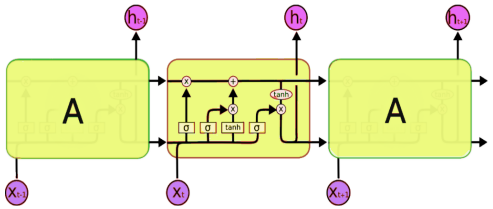


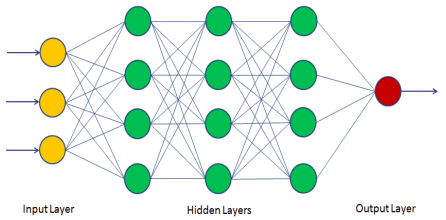




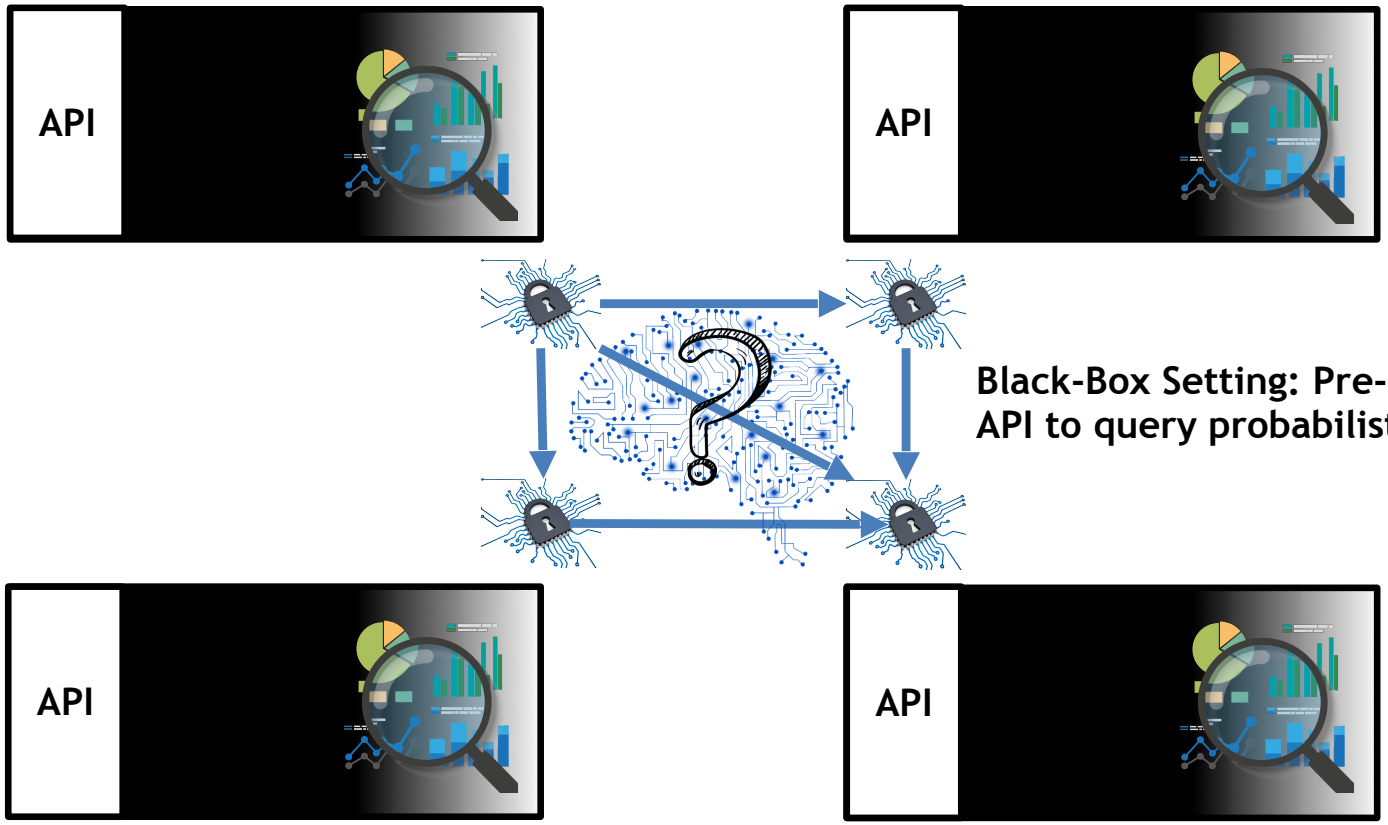
Challenge:
Private, heterogeneous model architecture
Prev. works assume white-box architecture





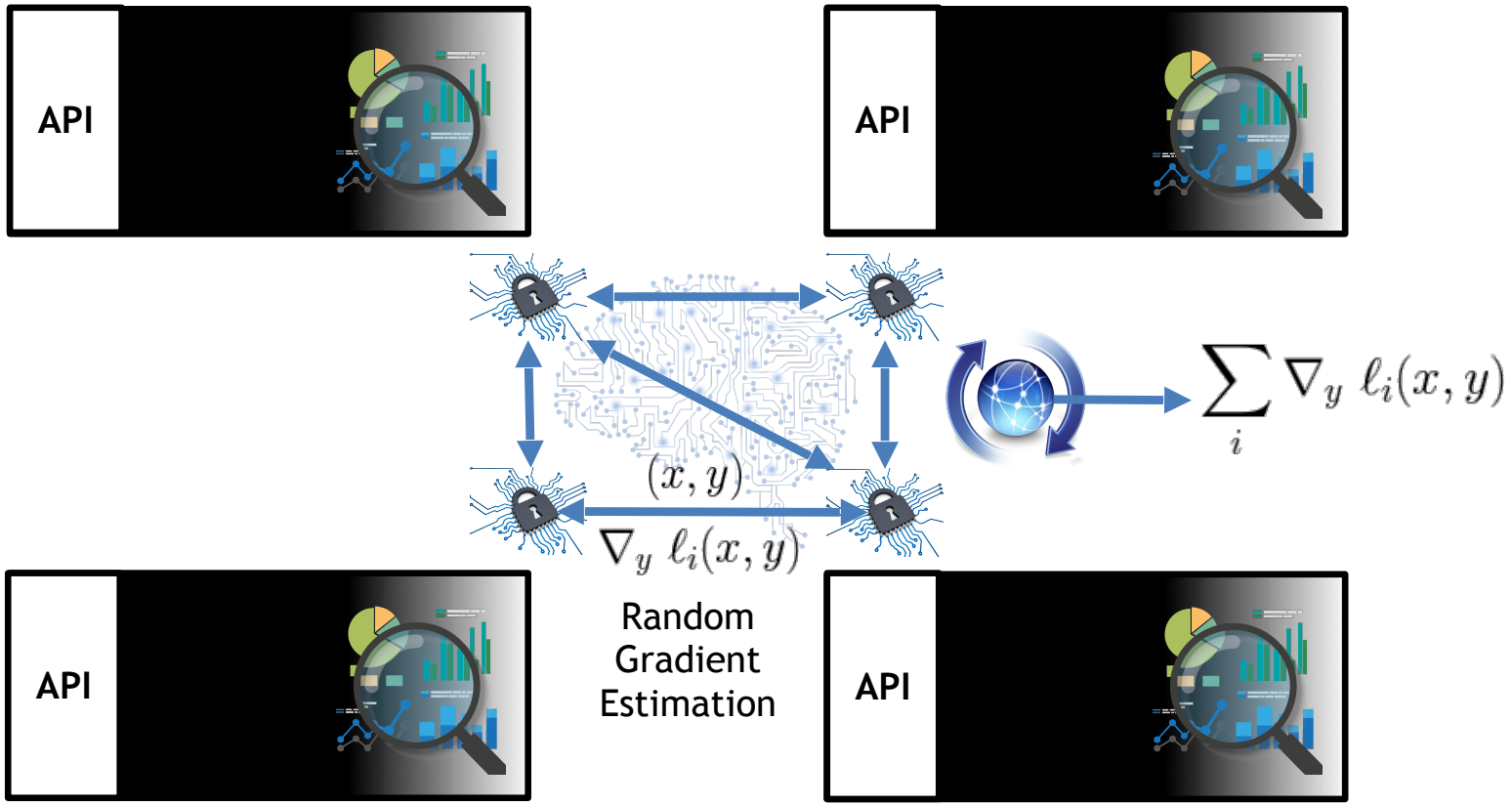


Model Fusion with Black Boxes



Black-Box Setting: Pre-trained model API to query probabilistic prediction

Model Fusion via Gradient Aggregation

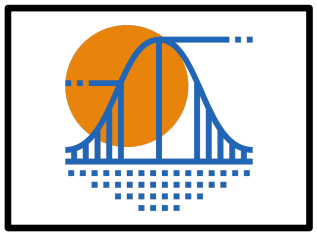


Light-Weight Fusion

Model Fusion via Gradient Aggregation

Guarantee: Disagreement rate is **upper-bounded by a constant** given sufficient training data

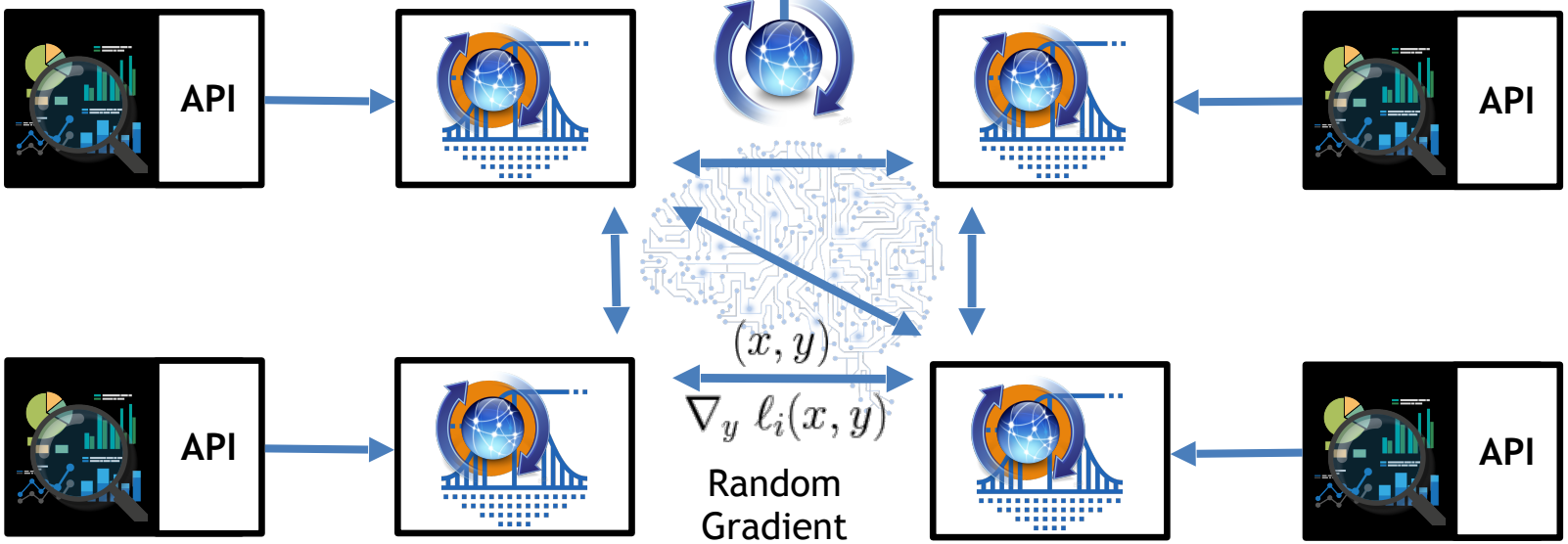
$$\underset{w_i}{\text{minimize}} \; \mathbf{D}_{\text{KL}}(q_i \;||\; p_i(y|x; w_i))$$



Gradient Aggregation

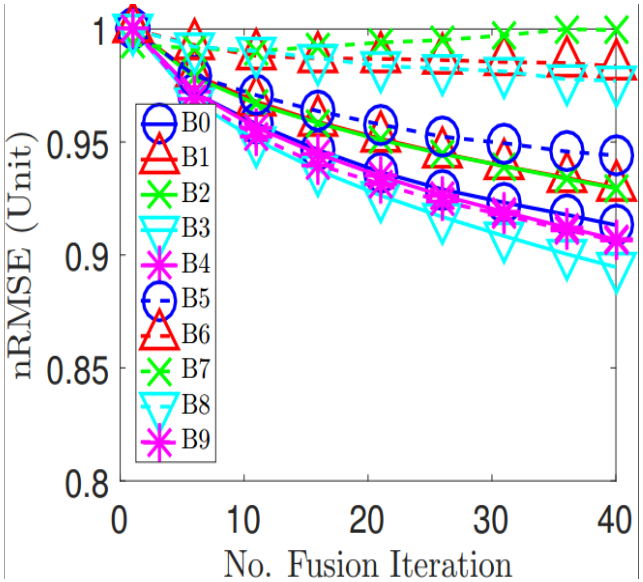
Persistent Fusion
Robust Imitation

For more details:
http://proceedings.mlr.press/v97/hoang19a/hoang19a.pdf



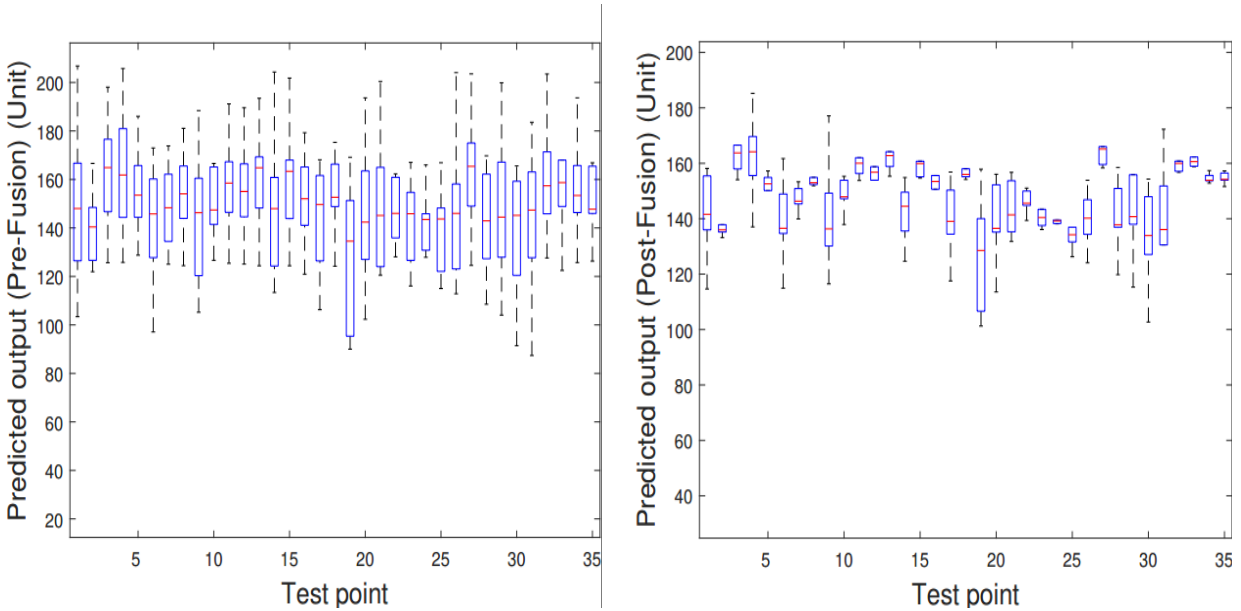
Model Fusion (via Gradient) Improves Performance

More accurate prediction with more fusion iterations



Up to 10% decrease in error for all black-box experts

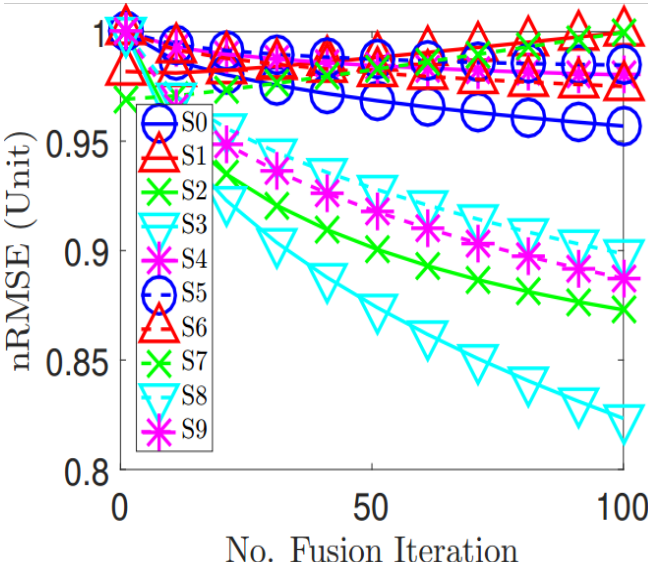
High prediction variance PRE-FUSION → Low prediction variance POST-FUSION



Before: Poor agreement
After: Better consensus

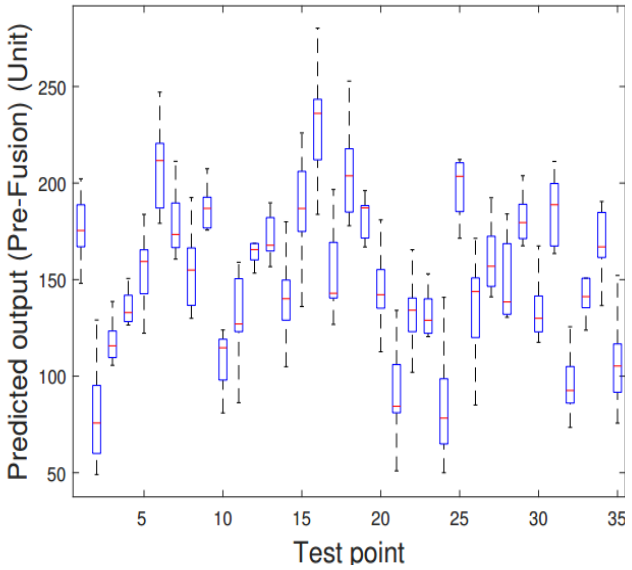
Model Fusion (via Imitation) Improves Performance

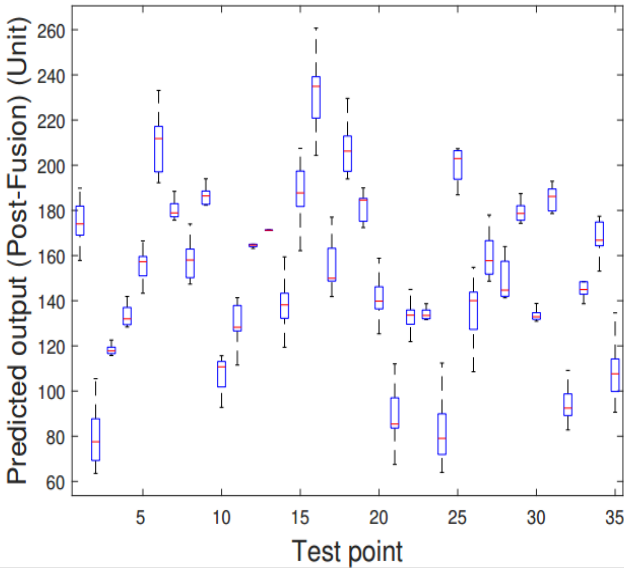
More accurate prediction with more fusion iterations



Up to 18% decrease in error for all black-box experts

High prediction variance PRE-FUSION → Low prediction variance POST-FUSION





Before: Poor agreement
After: Better consensus

Our Preliminary Work

[1] Collective Online Learning of Gaussian Processes in Massive Multi-Agent Systems.
Nghia Hoang, Minh Hoang, Bryan K. H. Low & Jonathan How.
In Proc. AAAI, 2019.
– Model fusion for Gaussian processes (for probabilistic regression)

[2] Collective Model Fusion with Multiple Black-Box Experts.
Minh Hoang, Nghia Hoang, Bryan K. H. Low & Carleton Kingsford.
In Proc. ICML, 2019.
– Model fusion for black-box AI (but restricted to probabilistic regression/classification)

Conclusion and Future Works

Contribution: Collective (Black-Box) Model Fusion Framework

- Advocate **edge intelligence** instead of cloud intelligence architecture (intelligence distributed on edge devices, not centralized on the cloud)
- **Communication-efficient** pairwise fusion operator between two agents, also amenable to cross-domain fusion and online learning
- Effective decentralized **message-passing** algorithm: resilient to system changes, while maintaining **consistent** global intelligence assimilation

Future Challenges:

- Agent might learn **different concepts** (each corresponds to a black box) concurrently (**component identification:** align concepts between two agents)
- Automatically **grow** encoding/surrogate model complexity to match that of streaming data (**durable** implementation for lifelong missions)
- **Extension** to incorporate decentralized **decision making** (completing the loop between inference & planning/decision making)

Thank You

References

[1] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[2] J. Quiñonero-Candela and C. E. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005.

[3] E. L. Snelson. *Flexible and efficient Gaussian process models for machine learning*. Ph.D. Thesis, University College London, London, UK, 2007.

[4] M. K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *Proc. AISTATS*, pages 567–574, 2009.

[5] M. Lázaro-Gredilla, J. Quiñonero-Candela, C. E. Rasmussen, and A. R. Figueiras-Vidal. Sparse spectrum Gaussian process regression. *Journal of Machine Learning Research*, pages 1865–1881, 2010.

[6] J. Hensman, N. Fusi, and N. D. Lawrence. Gaussian processes for big data. In *Proc. UAI*, pages 282–290, 2013.

[7] T. N. Hoang, Q. M. Hoang, and K. H. Low. A unifying framework of anytime sparse Gaussian process regression models with stochastic variational inference for big data. In *Proc. ICML*, pages 569–578, 2015.

[8] J. Chen, N. Cao, K. H. Low, R. Ouyang, C. K.-Y. Tan, and P. Jaillet. Parallel Gaussian process regression with low-rank covariance matrix approximations. In *Proc. UAI*, pages 152–161, 2013.

[9] Y. Gal, M. van der Wilk, and C. Rasmussen. Distributed variational inference in sparse Gaussian process regression and latent variable models. In *Proc. NIPS*, pages 3257–3265, 2014.

[10] K. H. Low, J. Yu, J. Chen, and P. Jaillet. Parallel Gaussian process regression for big data: Low-rank representation meets Markov approximation. In *Proc. AAAI*, pages 2821–2827, 2015.

[11] T. Campbell, J. Straub, J. W. Fisher III, and J. P. How. Streaming, distributed variational inference for Bayesian nonparametrics. In *Proc. NIPS*, pages 280–288, 2015.

[12] M. P. Deisenroth and J. W. Ng. Distributed Gaussian processes. In *Proc. ICML*, 2015.

[13] T. N. Hoang, Q. M. Hoang, and K. H. Low. A distributed variational inference framework for unifying parallel sparse Gaussian process regression models. In *Proc. ICML*, pages 382–391, 2016.

[14] J. Chen, K. H. Low, and C. K.-Y. Tan. Gaussian process-based decentralized data fusion and active sensing for mobility-on-demand system. In *Proc. RSS*, 2013.

[15] R. Allamraju and G. Chowdhary. Communication efficient decentralized Gaussian process fusion for Multi-UAS path planning. In *Proc. ACC*, 2017.

[16] M. K. Titsias and M. Lázaro-Gredilla. Variational inference for Mahalanobis distance metrics in Gaussian process regression. In *Proc. NIPS*, 2013.