

# リアルタイムな機械学習を可能とする ストリーム処理アーキテクチャの実装と評価

富士通株式会社

石垣 弘哉

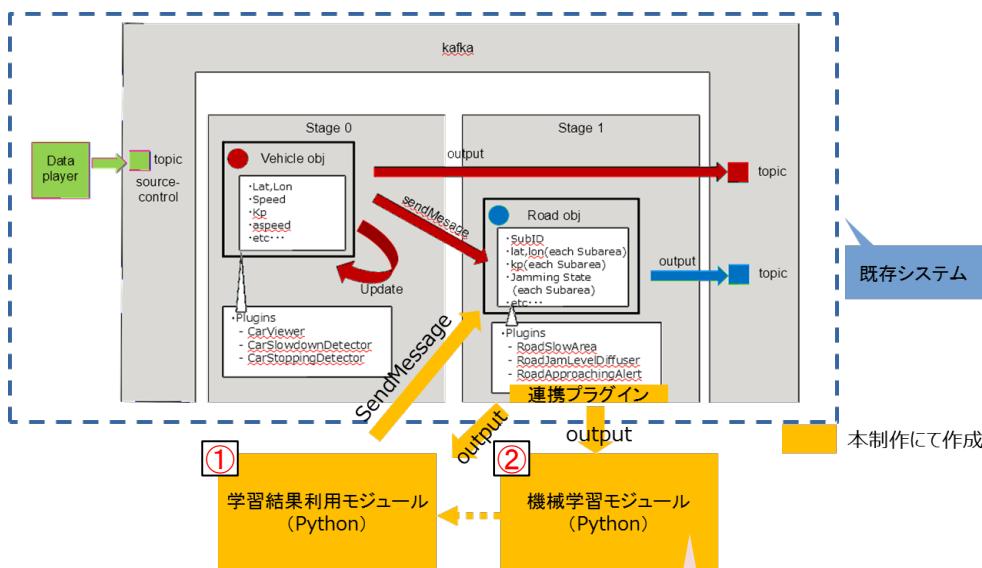
## 現行アーキテクチャの課題

- 大量のストリームデータをリアルタイムに処理することが求められる時代である
- その処理の中に重い処理を混ぜると遅延が発生し、積み重なると結果バックプレッシャーとして処理全体に大きな遅延が発生する

## 提案手法による解決

- 機械学習用のモジュールと橋渡しとなるロジックを実装し、リアルタイムな処理と機械学習を並列化する

## アーキテクチャの構成



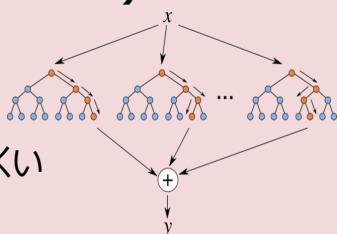
### 実装の詳細

- 個車の情報から区間の渋滞度合をリアルタイムに生成する既存システムに、数分後の渋滞度合を予測する機械学習モジュールを実装した
- 実装したモジュールのI/Fは以下の表の通り

	モジュール名	入力	出力
1	機械学習モジュール	JSON(200区間 * 100B/1min)	学習モデル
2	学習結果利用モジュール	学習モデル 現在から5-30分後 (5分毎) の時刻	現在から5-30分後 (5分毎) の時刻の渋滞予測値

### ERT (Extremely Randomized Trees)

- Random Forestの派生
- 決定木を作る際に、ランダムで決定
- 学習が高速
- 教師データが少なくても過学習しにくい
- 反面、精度は落ちる



[2017/9/13 10:00:00 , 12\_2\_2] → 学習モデル → [0.235]

出力例

```

import numpy.core.umath_tests as umath
import time
Predicted JamLevel: 0.2603982023505671
Predicted JamLevel: 0.2740551748187741
    
```

## 提案手法のメリット

### リアルタイム処理に対する機械学習

- データをリアルタイム処理の抽象度で扱うことを可能にする

### エンドユーザー視点

- 情報鮮度が向上することにより、適用可能なアプリケーションフィールドが拡大

### デベロッパー (サービス提供者)

- 視点リソースが不足した際のシステム再構成ポイントがわかりやすい

## まとめと今後の展望

### まとめ

- アーキテクチャの特性
  - ストリーム処理と機械学習のリアルタイムな並列処理
- 実行結果
  - 学習済みモデルを用いた予測

### 今後の展望

- アーキテクチャの改良
- 機械学習のバリエーションの選定