2020年度未踏ターゲット事業

ストリームデータ処理を実現するアニーリングマシンアプリケーションの探索と実装
― フレームワーク「sawatabi」とデモアプリケーションの開発 ―

寺田 晃太朗、古山 慎悟、臼井 純哉、小野 和輝

背景

ビッグデータを扱う産業でストリーム処理が重要（例: 異常検知、リアルタイムレポート、リアルタイムレコメンデーション）
ストリーム処理とは連続で終わりのない入力データを永続的にデータ処理する計算のこと

目的

ストリームデータに対して継続的アニーリング計算を実行可能なアプリケーションを開発するためのフレームワークの設計・開発

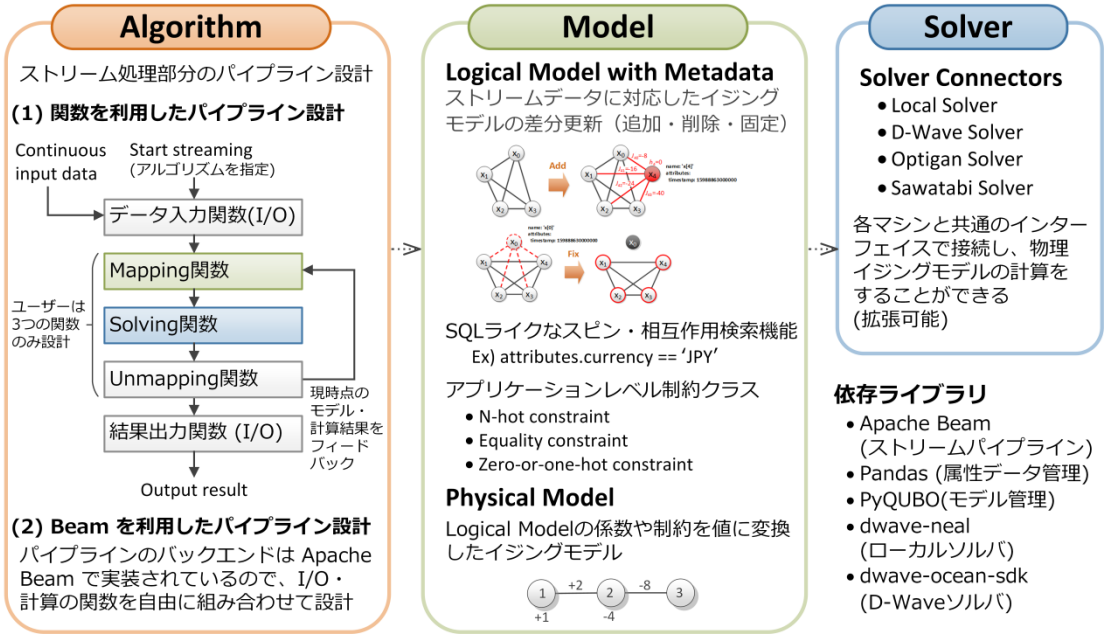
開発成果

開発フレームワーク「sawatabi」を作成
サンプルとして4つのデモアプリケーションを作成

- 異常検知
- ジョブショップスケジューリング問題
- 為替の最適裁定取引機会検出問題
- 巡回セールスパーソン問題

解決する課題

- これまで作成が容易でなかったストリームデータに対するアニーリング計算アプリケーションを開発可能
- 開発アプリはクラウド上のストリーム処理実行エンジン上(例: Google Cloud Dataflow) で実行可能
- ストリームデータの過去の答えをアニーリング計算の初期状態をして利用することで得られる解品質がよくなるケースあり

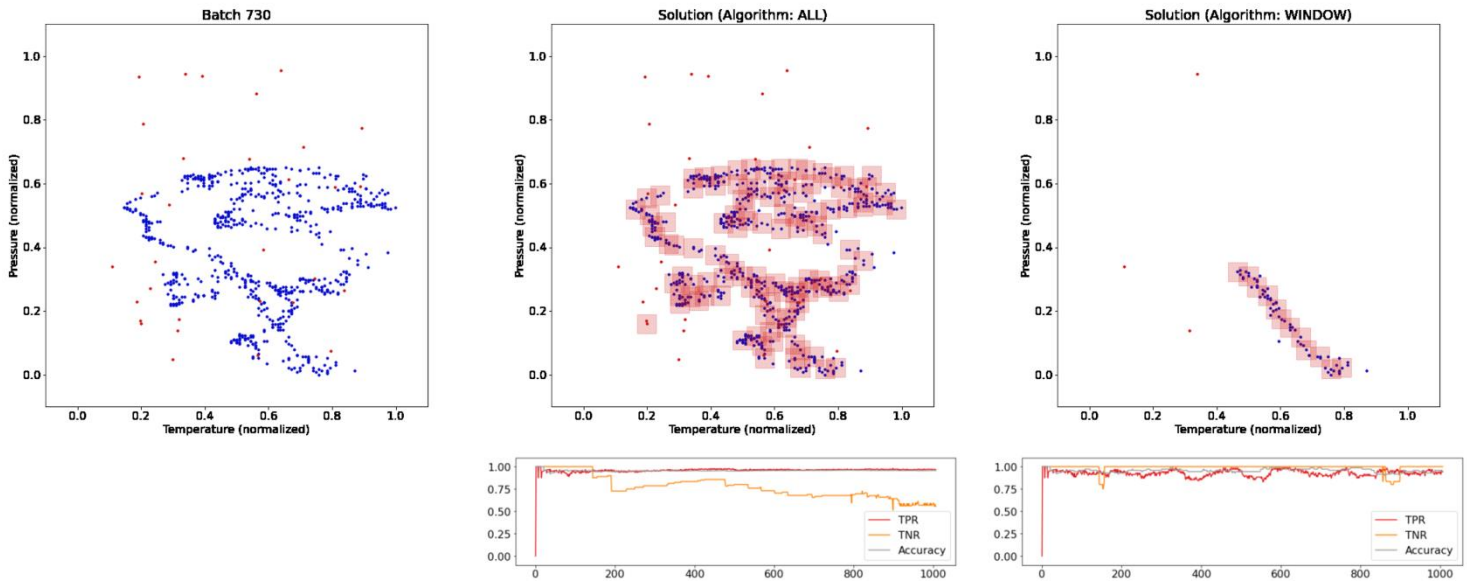


sawatabi OSSリポジトリ: https://github.com/kotarot/sawatabi

デモ：異常検知

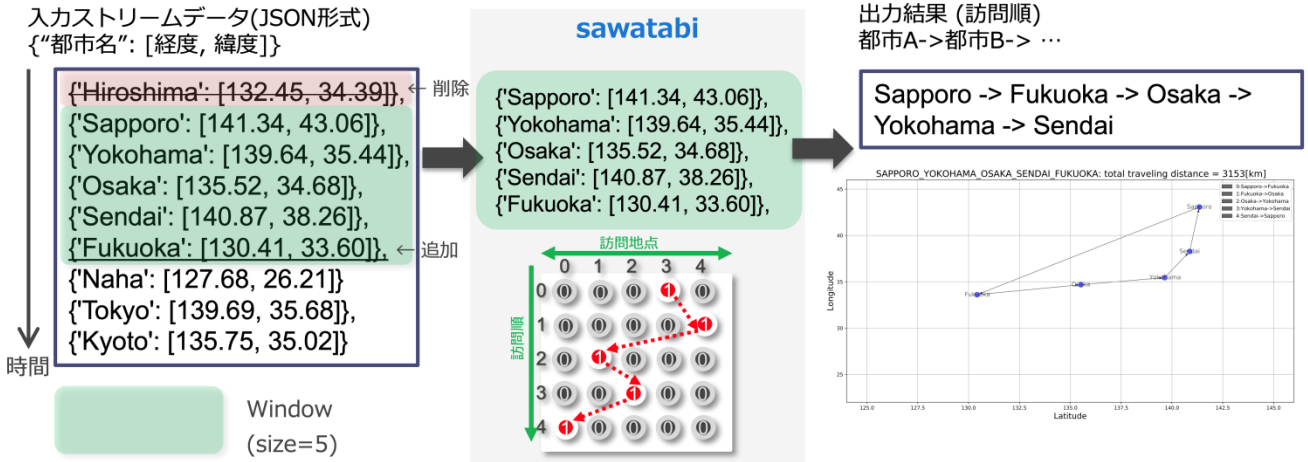
想定される応用先の例：
センサデータの異常検知
検証結果：
- 継続したアニーリング計算の実現
- データ更新アルゴリズムの選択により アプリケーションの結果が変わる
- 入力データの特徴やアプリケーションで 求めたい指標に適切なアルゴリズムを 選択することでより正確な異常検知の 予測モデルを構築できる

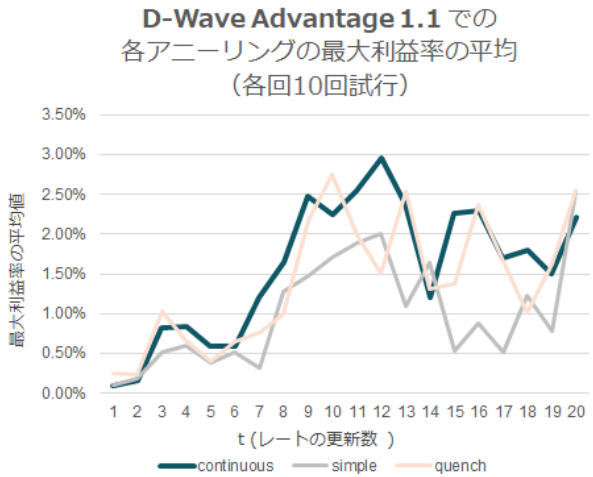
ストリームデータ　　アルゴリズム: Incremental　　アルゴリズム: Window



デモ：巡回セールスパーソン問題

想定される応用先の例：ライドシェアのルーティング最適化
検証結果：継続したアニーリング計算の実現
ストリーム処理実行エンジンとしてGoogle Cloud Dataflowで動作



デモ：為替の最適裁定取引機会検出問題

想定される応用先の例：為替などの取引
検証結果：過去の結果を初期状態として利用する ことで最大利益率が平均的に向上