

移動で
人を幸せに。



現場課題に向き合い MLOps成熟度を高める道

2023/6/21

自己紹介

GO株式会社

スマートドライビング事業本部

システム開発部 AI基盤グループ

森本 淳司

2021年 GO株式会社に入社

AI基盤グループで次世代AIドラレコサービス『DRIVE CHART』の
MLOps業務に従事



移動で人を幸せに。

2020年4月、JapanTaxi株式会社とDeNAのMOV/DRIVE CHART事業などが統合して誕生。

2023年4月、社名を「株式会社Mobility Technologies」から「GO株式会社」に変更しました。

「移動で人を幸せに。」をミッションに、ソフトウェアとハードウェアの両軸から移動の未来を創造しています。

<https://goinc.jp/>

GO Inc. の事業紹介

配車関連事業



個人様向け
タクシーが呼べるアプリのサービス運営・開発

GO BUSINESS

企業様向け
タクシー配車や経費精算などを簡単効率化
働くすべての人の移動を支えるサービスの
運営・開発



広告・決済事業



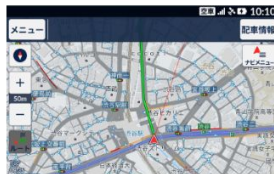
タクシーアプリのキャッシュレス決済機能

TOKYO PRIME



タクシー/ヘリ・サイネージメディア (動画広告)

乗務員向け ソリューション事業



乗務員向けアプリの開発・運営



乗務員向けナビゲーション端末



多言語表示、音声通訳機能のタブレット

スマート ドライビング事業



画像解析技術による地図情報の取得



持続可能な安全運転のための画像解析技術



自社開発によるドライブレコーダー

次世代向け R&D事業



CO₂削減・タクシー EV化



データビジネス



タクシー乗務員採用支援

※記載されている会社名や商品名などは、各社の商標または登録商標です。(出願中含む)

本日は話すこと

1. 次世代AIドラレコサービス『DRIVE CHART』について
2. MLOps成熟度モデルについて
3. 機械学習プロダクト・サービスでの現場課題と解決方法について
4. MLOps成熟度を高める道

01

AIドラレコ DRIVE CHART

次世代AIドラレコサービス『DRIVE CHART』について

AIを搭載したドライブレコーダーによる事故防止支援サービス

AIが様々な危険シーンを検知して、普段の運転行動を分析

安全運転をあたりまえの習慣に

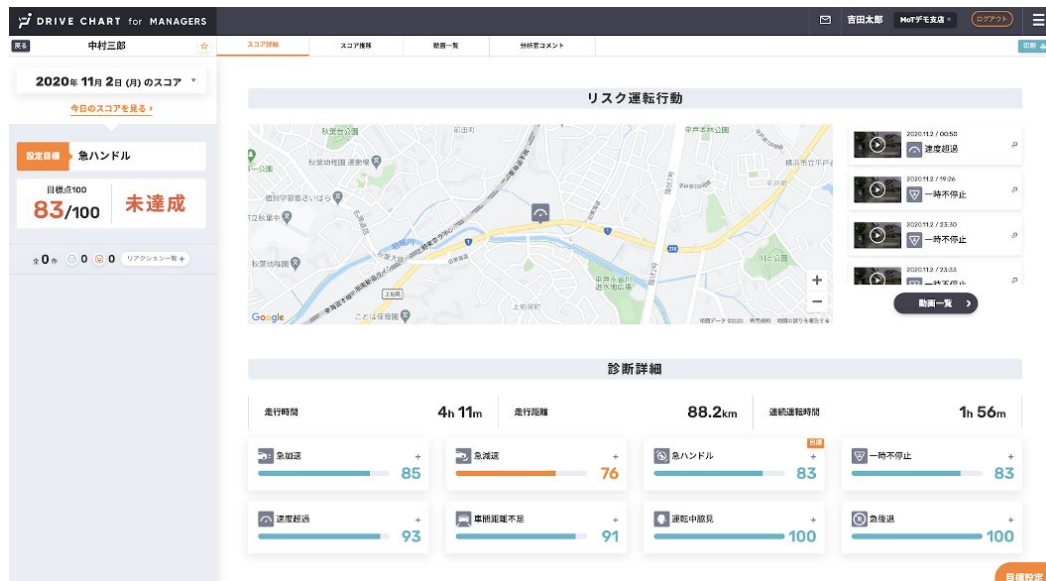
 DRIVE CHART

次世代AIドラレコサービス



DRIVE CHARTは危険シーンを抽出しWebでレポート確認

- AIにより危険シーンを抽出
- Webで簡単に該当シーンの動画を閲覧
- 運転行動をレポート表示



なぜDRIVE CHARTが必要なのか

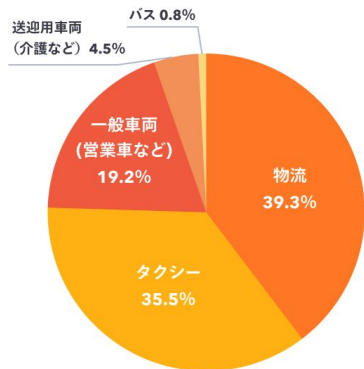
- 社員が社有車で事故を起こせば会社の責任
 - 企業は事故を起こさないための仕組みづくりが必要
- やみくもに事故削減活動を行ってもキリがない
 - すべての運転記録をチェックするのは現実的ではない
 - 社員それぞれの事故のリスクを把握し効率・効果的な指導、改善が求められる
 - 運転の癖を可視化し、社員ごとに指導内容をカスタマイズできる



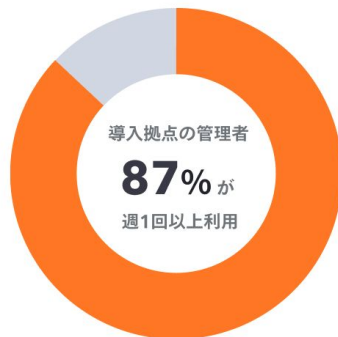
DRIVE CHART 契約台数が大幅に伸長

- 2019年6月のリリース以降、日々導入事業者が増加しています
- 契約車両6万台突破
- 運転が生業のタクシー・物流事業者をはじめ、営業車や介護送迎車両など幅広く導入
- DRIVE CHART搭載車両が日本中を走行、1日で高速道路の約9割・一般国道の約7割を網羅

車両利用別導入割合



週1回以上サービス利用の管理者比率

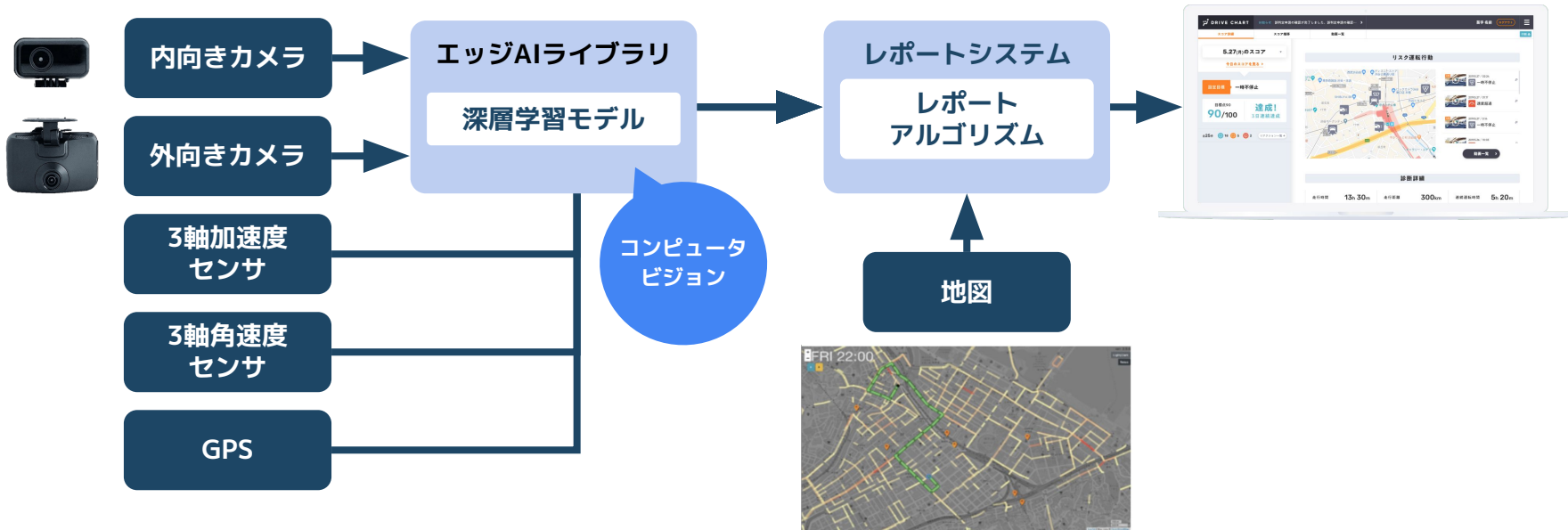


高速道路の網羅率

96%以上!



AIドラレコ DRIVE CHARTの仕組み



参考：DRIVE CHARTを支える技術（内田、2020）
<https://www.slideshare.net/ren4yu/drive-chartai-239311682>

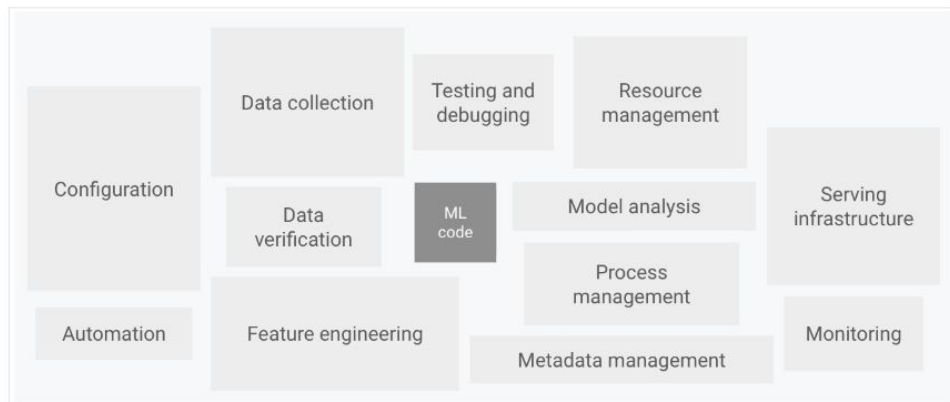
02

MLOps

成熟度モデルの
復習

MLOpsの成熟度

- Google
 - Level 0: 手動プロセス
 - Level 1: ML パイプラインの自動化
 - Level 2: CI / CD パイプラインの自動化
- Microsoft
 - Level 0: MLOps なし
 - Level 1: DevOps あり、MLOps なし
 - Level 2: トレーニングの自動化
 - Level 3: モデル デプロイの自動化
 - Level 4: MLOps の再トレーニングの完全自動化
- 学習/デプロイ プロセスが自動化されているかどうか

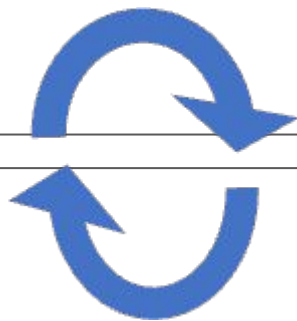


<https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning?hl=ja>
<https://learn.microsoft.com/ja-jp/azure/architecture/example-scenario/mlops/mlops-maturity-model>

機械学習モデルのデプロイサイクルを自動化する

- 実験による成果物を本番環境へデプロイする
- 本番環境で得られた結果を次の実験に活かす

実験環境



検証環境・本番環境

03

MIL現場課題と 解決方法

現場のよくあるMLOps課題

- 実験管理
 - 再現できない、GPUコストが高い、学習に時間がかかる
- データセット
 - データフローが複雑、ラベルが疑わしい
- インフラ
 - 構成が複雑になり、問題発生時の原因切り分けが困難
 - コストが最適化できていない
- プロダクションコード
 - サービス・プロダクトが発展しソースコードが膨大になり、コードの変更が怖い、可読性が低い
- テスト
 - 変更のたびに不具合が混入する、単体テストコードがない、テストコードの一貫性がない(オレオレテストコード)
- ドキュメント
 - ドキュメントがない、ドキュメントの種類が考慮されていない、更新されていない
- ロギング
 - 必要なロギングがない、無駄なログが多くコスト最適化できていない
- OSSの進化についていけない
 - deprecatedなバージョンを使用している
- 実験環境から本番環境へのシームレスな適用
 - A/Bテスト、シャドウテスト、大規模テスト
- バージョン管理
 - モデルバージョン、データバージョン、どこに記録するか
- MLOpsは何でも屋になりがち

課題をグループ化してみた

実験環境

機械学習

- 実験管理
- データセット作成

検証環境・本番環境

ソフトウェア
エンジニアリ
ング

- テスト
- ドキュメント
- ログイン
- バージョン追従

インフラ
構築運用

- 安定運用
- 監視・通知
- コスト最適化

機械学習視点の課題と対応方法

- 正しいプロセスで実験を素早く多数行いたい
 - 課題: 正しいプロセスでないと実験の知見が積み上がらない
 - あの実験は再現できないからもう一度やる
 - 目的不明の実験が発生する
 - 解決策1: 実験管理ツールの導入
 - 管理を自動化することで再現性を担保し結果の共有や可視化が容易になる
 - 解決策2: 実験記録のテンプレート化
 - 目的、背景、手法、データセット、結果、考察、ネクストアクションをテンプレにする
 - GitHub issueなどでまとめた過去の実験の積み上がりが可視化できるように
 - 課題: 素早く多数行うことでビジネス価値のある結論に競合優位でたどり着ける
 - 必要なコンピューティングリソースを確保できない
 - 実験開始までの導入作業が多い
 - 解決策3:
 - スケールする実験環境を整備する

ソフトウェアエンジニアリング視点の課題と対応方法

- サービス・プロダクトが発展しソースコードが複雑化& ソースコードの可読性が低下
 - プロダクションコードをミステリー小説にしてはならない
 - ファイル名、クラス名、関数名、(スコープの長い)変数名がわかりづらい
 - 共通系ファイル(common.py, util.py)に汎用でないコードが散逸している
 - 変数名の不用意な省略はしない
 - 複数の意味を持つ名前が使用されていて文脈を考慮しないといけない
 - conf
 - ML => confidence
 - SWE => configuration
 - 解決策1: 納得性、実用性の高いコーディング規約を用意する(コスト高)
 - 解決策2: コーディングレビューのたびに指摘(コスト低、なんども同じことを指摘する可能性)
 - 解決策3: 定期的にミーティングを開いて方向性やあるべき姿の共通認識を持つ(コスト中)

04

MLOps 成熟度を 高める道

MLOps成熟度を高める道

- 評価指標の定義
 - ML評価指標: precision, recall, ...
 - ビジネス評価指標: 売上, ユーザー満足度, カスタマーサクセス対応工数, ...
- データフローの確立
 - 本番環境で推論した結果を次の実験に活用する
 - 4w2v (when, who, where, what, model version, data version)
 - 推論結果をDWHに蓄積する
 - アノテーションフロー (Human-in-the-Loop)
- 監視・通知
 - 本番環境でのサーバリソースの監視・通知
 - データ肥大化時の運用
 - OOM発生
 - MLの評価の監視・通知
 - データドリフト、コンセプトドリフト
- 継続的学習
 - 学習トリガーの決定 (定期実行 / 精度劣化)
 - ワークフローの運用: データ取得, 前処理, 学習, 後処理
- デプロイ
 - A/Bテスト, シャドウテスト, 多腕バンディット

サイクルの自動化

実験環境

機械学習

検証環境・本番環境

ソフトウェア
エンジニアリ
ング

インフラ
構築運用

まとめ

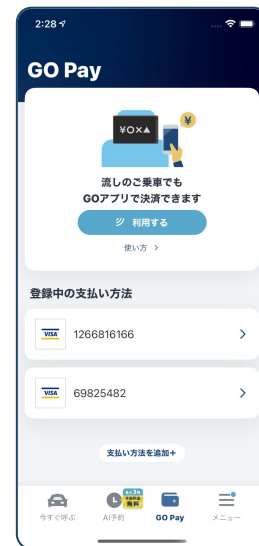
- MLOps成熟度モデルの理解する
- あるべき姿と現時点のギャップを認識する
 - あるべき姿は自分たちで定義する
- 現場の課題に向き合い続ける
- 属人化を減らし自動化する、そして新しい取り組みを増やす
- 解決策は1つではない
 - 体系的に案をだし、最適解を選択する
 - あるべき姿から離れた局所解に陥らない

GO株式会社と一緒に開発しませんか



GO
TAXI GOEs Next.

<https://hrmos.co/pages/goinc/jobs>





移動で人を幸せに。

文章・画像等の内容の無断転載及び複製等の行為はご遠慮ください。