

開発者とデータサイエンティストが連携するための5つの方法

データサイエンティストを味方につける

効果的な人工知能 (AI) ベースのアプリケーションを構築するためには、データサイエンティストの仕事を理解し、彼らとどう協働すべきかを学ぶ必要があります。つまり、AI の基本を知り、同僚であるデータサイエンティストとのコラボレーション方法を学ぶことが重要になります。データサイエンティストと協働し、AI を活用したインテリジェントなアプリケーションを構築する際に知っておくべき5つの秘訣をご紹介します。良好な協力関係を築き、優れたアプリケーション開発のコラボレーションを実現するためのガイドとして、このガイドをご活用ください。

1 データサイエンティストの仕事を理解する

データサイエンティストは通常、アプリケーションの開発やソフトウェアへのモデルの統合よりも、モデルの構築と改良に関心があります。継続的インテグレーション/継続的デリバリー (CI/CD) パイプラインの構築や、アプリケーションコードの記述に関わりたいと思うことはほとんどなく、Python、R、Jupyter Notebooks など、開発者には馴染みのないツールを使用していることもあります。

同じ目標に向かっていてもかかわらず、彼らの方からオープンコラボレーションを提案してくれることはまずないでしょう。しかし、優れたアプリケーションを構築するためには、開発者とデータサイエンスチームとの強固で一貫したチームワークが不可欠です。積極的なコラボレーションは、以下のような結果をもたらします。

- ▶ AI を効果的に活用した、インテリジェントなデータ駆動型アプリケーションのデプロイ
- ▶ データサイエンティストのモデルワークを、貴社と顧客に付加価値をもたらすデプロイ可能なソリューションに役立てるチャンス

最初にアプローチをかけ、データサイエンスの同僚との共同作業を促進するのは、おそらくあなたの役目となるでしょう。このチェックリストにあるガイダンスを採用し、データサイエンスチームと有益な方法でつながる方法を見つけましょう。

2 共通点を見つける

開発手法を説明し、それがデータサイエンティストの努力をどのように補完するかを確認することは、摩擦のないコラボレーションを行い、全員にとって有益なエクスペリエンスを生み出すために重要なプロセスです。そのためには、以下のことが不可欠です。

- ▶ **頻繁なタッチポイントを促す**：共に取り組むプロジェクトを確実に軌道に乗せるためには、頻繁かつ定期的なタッチポイントの設定が重要になります。
- ▶ **境界を尊重する**：データサイエンティストは、アプリケーションを本番環境にデプロイする方法を知る気も必要もないかもしれません。MLOps は一般的な概念ですが、データサイエンティストの中には Jupyter Notebook をメールで送ることを好む人もいます。彼らの興味や働き方を尊重すれば、相手もそれに応えてくれるはずです。
- ▶ **互いのプロセスを共有する**：データサイエンティストの思考を学ぶだけでなく、あなたのプロセスや、Git、Tekton、Kubernetes など、本番環境で使っているツールを共有しましょう。オープンソースの精神に則り、あなたのプロセスを公開しましょう。
- ▶ **コラボレーションのために共通のプラットフォームを使用する**：Red Hat® OpenShift® Data Science のような共通のクラウドネイティブ AI 開発プラットフォームは、あなたとデータサイエンスチーム間のコラボレーションをサポートし、促進します。このプラットフォームは、AI ツールの使用を民主化し、チームがインテリジェントなアプリケーション開発を実施し、加速できるよう促します。

3 モデル学習ツールの使い方を学ぶ

データサイエンティストが日常的に使用しているモデル学習ツールの基礎知識を、最低限身につけておきましょう。モデル学習ツールの実用的な知識を身につけることができれば、モデルがどのように構築されるかを理解するのに役立ちます。以下は、代表的なモデル学習ツールやライブラリです。

- ▶ Jupyter と PyCharm の開発環境
- ▶ PyTorch
- ▶ scikit-learn
- ▶ TensorFlow

これらのツールやその他のツールと慣れ親しんでおくことで、デプロイされたモデルアプリケーションを成功させる確率が高まります。また、モデル作成にかかる作業をより深く理解することができ、モデルがインテリジェントなアプリケーションにスムーズに統合されない場合の問題を解決するのに役立ちます。

4 使い慣れたツールやプロセスを使い続ける

データサイエンティストや AI と仕事をするようになると、さまざまな新しいプロセスやツールを学ぶ必要が出てきます。しかし、自分のロジックを行う際に、使い慣れたツールを使い続けることは可能です。アプリケーションのコードやモデリングでは、好きな言語やフレームワークで行うことができます。

たとえば、Quarkus の開発者は、Quarkus でアプリケーションロジックを作成して、Representational State Transfer (REST) エンドポイントへのアプリケーション・プログラミング・インタフェース (API) 呼び出しを行い、データサイエンティストは Python や R などのツールを使用して実際のデータ処理と予測を処理することができます。

5 モデルはアプリケーションの一部であることを忘れない

モデルも重要ですが、その背後にある MLOps も重要です。本番環境に移行してからもモデルの性能を維持するために、必要な 4 つのことをご紹介します。

- ▶ 開発するアプリケーションに適した、モデル向けインフラストラクチャを構築する
- ▶ 既存の CI/CD パイプラインを拡張するか、新規を作成して、モデルのトレーニングとサービスの両方を処理する
- ▶ モデル向けアプリケーションをスケーリングする
- ▶ Apache Kafka などのストリーミング・データ・サービスや、データ収集コンポーネントを統合する

アプリケーションのデプロイは始まりに過ぎません。モデルは常に変化し続けるので、監視する必要があります。データサイエンティストと密接に連携し、モデルドリフトを防ぐために、あなたや運用担当者が監視する必要のあるメトリクスを定義しましょう。問題や変更が発生した場合は、データサイエンティストと協力してモデルを改良し、改善すると良いでしょう。

詳細はこちら

Red Hat OpenShift Data Science で Red Hat の ebook を読む。

試してみる

Red Hat OpenShift Data Science を試してデモを見る。



Red Hat について

Red Hat は、受賞歴のあるサポート、トレーニング、コンサルティングサービスを通じて、組織の環境全体の標準化、クラウドネイティブ・アプリケーションの開発、複雑な環境の統合、自動化、保護、および管理を支援します。

アジア太平洋

+65 6490 4200
apac@redhat.com

オーストラリア

1 800 733 428

インド

+91 22 3987 8888

インドネシア

001 803 440 224

日本

0120 266 086
03 5798 8510

韓国

080 708 0880

マレーシア

1 800 812 678

ニュージーランド

0800 450 503

シンガポール

800 448 1430

中国

800 810 2100

香港

800 901 222

台湾

0800 666
052

f fb.com/RedHatJapan
t twitter.com/RedHatJapan
in linkedin.com/company/red-hat

jp.redhat.com
#F31386_0422

Copyright © 2022 Red Hat, Inc. Red Hat, Red Hat ロゴ、および OpenShift は、米国およびその他の国における Red Hat, Inc. またはその子会社の商標または登録商標です。Linux® は、米国およびその他の国における Linus Torvalds 氏の登録商標です。その他のすべての商標は、それぞれの所有者に帰属します。