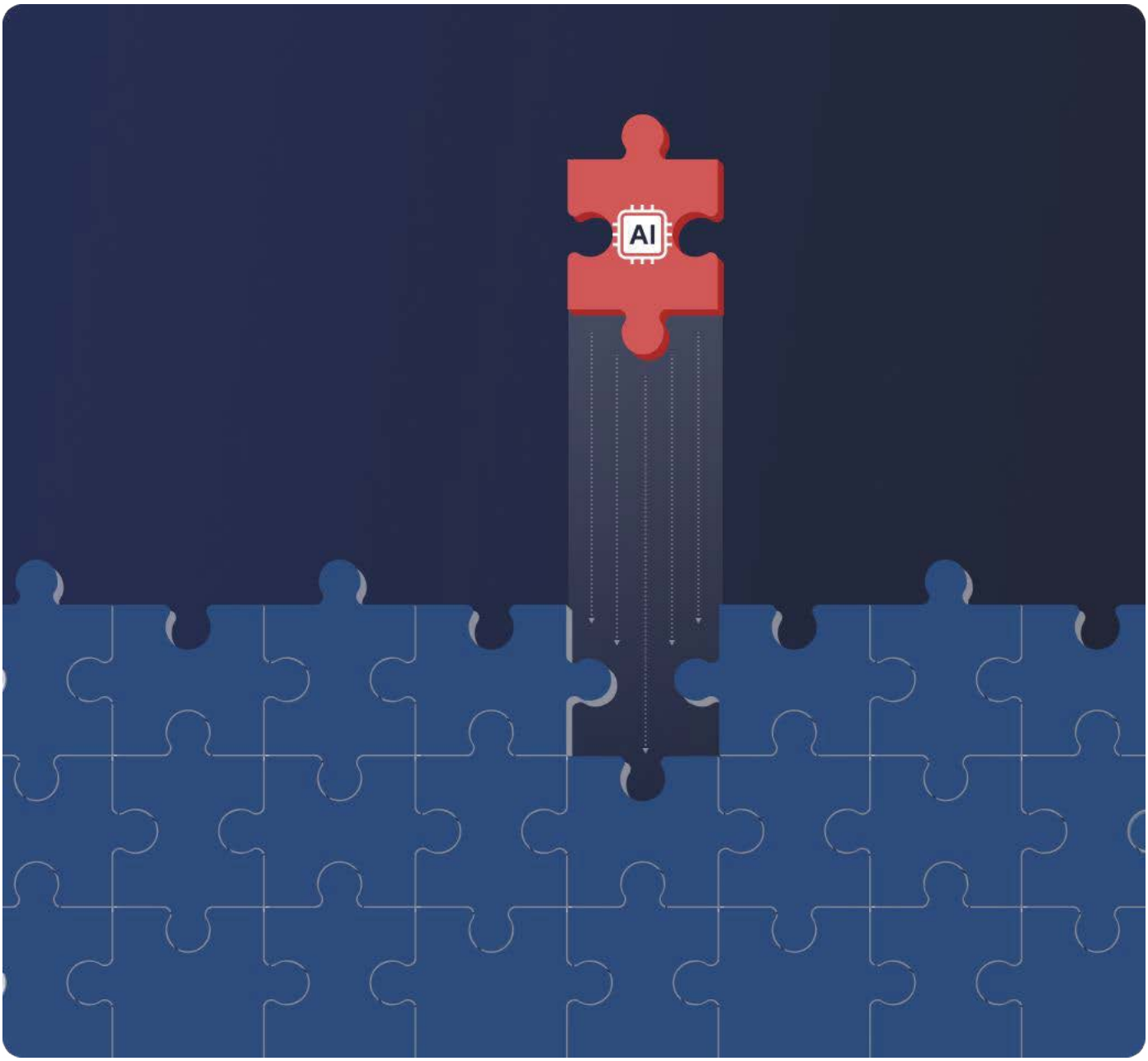
テクノロジーリーダーのための AI/MLOpsガイド

目 次

急速な技術革新の中で、AI/MLを活用したアプリケーション開発は、もはや競争優位ではなく不可欠な取り組みとなっています。高度な機能を求める顧客の期待に応えるため、企業はサービスの進化を迫られています。

しかし多くの企業は、AI/MLを既存アプリケーションに効果的に統合したり、新たなAI主導のソリューションを構築したりすることに課題を抱えています。急速な変化への対応に加え、プロセス整備や標準化の必要性にも直面しているためです。

本ホワイトペーパーでは、AI/MLを既存の開発フレームワークへ統合するための背景、課題、ベストプラクティスに加え、無計画な実験のリスクやMLOps導入の進め方について解説します。



概要

テクノロジーリーダーがMLOpsを推進するうえでの重要ポイントは次のとおりです。

■ 競争・イノベーション・コスト削減・スケールが導入を後押し

顧客期待の高まりと競争の激化により、AI/MLへの投資は不可欠になっています。加えて、MLOpsはコスト削減とスケーラビリティ向上にも大きく貢献します。

■ 標準化・ガバナンス・セキュリティが課題

AI/MLは比較的新しい領域であり、従来の開発プロセスとの統合やセキュリティ確保が十分に確立されていないことが、多くの組織にとっての課題となっています。

■ AI/MLプラットフォームの活用が重要

社内で一から試行錯誤するのではなく、既存ツールやプロセスと統合されたMLプラットフォームを活用することで、開発効率を大幅に高められます。

■ 経営層のリーダーシップが不可欠

AI/ML開発には複数チームの連携が必要です。組織横断の協力体制を構築するには、経営層の関与が重要になります。

■ AI/ML特有のセキュリティリスクへの対応

AI/MLの導入では、従来のソフトウェア開発と同様のセキュリティ原則を適用する必要があります。

■ 成功の鍵は ML(Sec)Ops

AI/MLをSDLC全体に組み込み、開発・運用・セキュリティを統合することで、データ保護を強化し、組織全体でスケーラブルなAI活用を実現できます。

現在のAI/MLランドスケープ

MLおよびAIへの投資を促す主要なビジネス要因は、顧客からの期待の高まりです。ユーザーは効率的に動作する製品を求めるだけでなく、インテリジェントなアルゴリズムによる高度な機能を期待しています。

その他の主な要因は以下の通りです。

- **競争:** 急速に変化する市場において、AI/MLでイノベーションを起こせない企業は、取り残されるリスクがあります。
- **コスト削減**: ML開発ワークフロー全体を通じたタスクの自動化は、運用効率の向上をもたらし、時間とリソースを節約します。
- **イノベーション:** AIとMLは、成長とサービス向上のための新しい道を模索する能力を提供します。
- **スケーラビリティ :** 組織は、特に金融や保険などの分野において、既存のモデルをより効果的に活用しようとしています。

しかし、実装の現実は想定よりもはるかに困難です。AIとMLがすでにどこにでも存在し、一夜にしてすべての分野に革命を起こしていると信じる人は多いですが、実態はもっと複雑です。メディアやマーケティングが「すべての企業がAIを使いこなしている」ような世界を描き出している一方で、ほとんどの大企業は、これらのテクノロジーを構築・展開する最も効果的な方法を模索している段階です。イノベーションへの強い意欲はあるものの、多くのビジネスにはAIとMLをうまく運用化（オペレーショナライズ）するために必要な知識とインフラが不足しています。

用語の整理：AI vs ML vs 生成AI

AIと生成AI、あるいはAIとMLはしばしば混同されがちですが、それぞれ明確な役割があります。

AI (人工知能)：知的なシステムやアプリケーションという「最終結果」を指す包括的な分野です。
ML (機械学習)：その結果を達成するための「手法」です。
生成AI (GenAI)：AIのサブセットであり、既存データの収集と合成を通じて、テキスト、画像、動画、音声などの新しいコンテンツを生成することに焦点を当てています。

AIを包括的な分野、MLをその中の重要な技術、生成AIをその技術の特化した応用例と考えると分かりやすいでしょう。

過去から予測するAI/MLの未来

数年ごとに、組織の規制や確立されたベストプラクティスを追い越すような、技術の突然の飛躍が起こります。例えば、オープンソース革命やクラウド革命の際も、熱心な個人が組織の対応よりも早く新技術を使い始めるという傾向が見られました。この動きは当初、自由と創造性をもたらしましたが、多くの企業が経験したように、構造的なアプローチがなければすぐに混乱に陥る可能性があります。

AI/ML開発に関して「The State of AI & LLMs Report」によれば、これらの開発イニシアチブの多くはベータ段階（43%）にありますが、かなりの割合（38%）がすでに本番稼働しています。これは、組織がアプリケーションにAIを取り入れることをいかに熱望しているかを示しています。今後1年間でより多くのプロジェクトが本番段階に進むにつれ、組織は品質、セキュリティ、コンプライアンスを維持するために、多様なツールやプロセスを統合する必要に迫られます。しかし、これらすべての新しいプロセスを構築するという膨大なタスクが、本番移行の遅れを招く可能性があります。

これに対処するため、組織はMLモデルやAIサービスを提供するための内製ツールやオープンソースツールから、商用ソリューションへと移行し始めています。このシフトは、企業がMLモデル開発の内製能力を維持するオーバーヘッドを避けるために、外部ソリューションを求めるようになっていることを示しています。市場の多様性が高まることは本質的に悪ではありませんが、この急速な探索はしばしば標準化やガバナンスの欠如を招きます。これは、企業が自社の慣行を評価し始める際に注意が必要となる課題です。

実例

英国のある小規模銀行は、数年前からリスク評価にAI/MLを活用していました。しかし、モデルの使用要求が急増し、モデルを外部アプリケーションに提供する必要が生じた際、既存のシステムでは増加した負荷を処理できないことに気づきました。これが、JFrog MLのような統合ソリューションを探すきっかけとなりました。

AI/MLモデル開発における役割

成功するAI/MLプロジェクトには、データサイエンティスト、ソフトウェアエンジニア、DevOpsプロフェッショナル、プロダクトマネージャー、データエンジニアなど、複数のステークホルダーが関わります。しかし、これらの役割は背景や専門知識が異なるため、部門横断的なコラボレーションにおいて課題が生じることがよくあります。

責任とステークホルダーの関与

AI/MLモデル開発において、各役割は重要かつ補完的な機能を果たします。以下に主な責任をまとめます。

データサイエンス

- **データの探索と分析：**大規模なデータセットを精査し、パターンや異常を発見する。
- **特徴量エンジニアリング：**MLモデルの性能を向上させるために、新しい変数（特徴量）を作成する。
- **モデル開発と評価：**アルゴリズムを構築し、モデルを設計する。ハイパーパラメータの調整を行い、精度、適合率、再現率、F1スコアなどの指標で性能を評価する。
- **統計分析：**モデルの堅牢性を確保するために、統計的テストを用いて発見事項を検証する。

データエンジニアリング

- **データパイプライン開発：**多様なソースからデータを収集、操作、保存するためのデータパイプラインを作成する。
- **データ品質管理：**分析やモデルトレーニングのためにデータを準備する際、検証チェックやクリーニングプロセスを実施する。
- **データベース管理：**大容量のデータを効果的に扱うために、データウェアハウスやデータベースを最適化する。

MLエンジニアリング

- **モデルの展開と運用化：**モデルを開発から本番環境へ移行させ、本番システムに統合された際のスケール、効率、信頼性を確保する。
- **モデルの監視と保守：**本番環境でのモデルのパフォーマンスを継続的に監視し、予測のドリフト（乖離）や異常を検知するためのツールを実装して、必要に応じてモデルを再学習させる。

MLエンジニアリング

- **APIとサービスの構築：**他のシステムがMLモデルと対話できるように、APIを構築する。
- **データサイエンスとの連携：**データサイエンス部門と緊密に連携してモデルの要件を理解し、コードのパフォーマンスを最適化し、ソフトウェア開発のベストプラクティスを実装する。
- **アルゴリズムの最適化：**MLワークフローのボトルネックを特定し、計算効率を向上させるための最適化を行う。
- **ツール提供と自動化：**反復的なタスクを自動化し、MLパイプラインを合理化するために必要なツールを開発・維持する。
- **ステークホルダーとのコミュニケーション：**ビジネスユニットやIT部門を含む多様なステークホルダーと連絡を取り、モデルの出力を組織の目標に合わせ、既存システムとの強固な統合を確実にする。

DevOps

- **インフラ管理：**クラウドまたはオンプレミスを問わず、ビッグデータやAI推論をサポートするインフラを含め、MLモデルをホストするために必要なインフラを管理し、環境の安全性とスケーラビリティを確保する。
- **継続的インテグレーション/継続的デリバリー (CI/CD)：**開発と展開のプロセスを自動化し、定期的な更新を可能にするとともに、MLアプリケーションの信頼性を維持する。
- **監視と保守：**本番環境でモデルのパフォーマンスを追跡する監視ツールを実装し、異常やパフォーマンスの問題が発生した際にチームにアラートを通知する。

SecOps

- **AIモデルのセキュリティ監視：**AI/MLモデルを標的とした異常や悪意のある活動を検知する監視システムを実装し、その完全性とパフォーマンスを保護する。
- **AI脅威へのインシデント対応：**モデル反転やデータポイズニングなどのリスクに対処するため、AI/ML環境に特化したインシデント対応プロトコルを開発する。
- **アクセス制御とデータガバナンス：**機密性の高いデータセットやアルゴリズムを不正アクセスから保護するため、厳格なアクセス制御とデータガバナンスを強制する。
- **脆弱性アセスメントとコンプライアンス：**MLパイプラインの定期的な脆弱性評価を実施し、AI関連の規制やデータ保護法への準拠を確実にする。

プロダクトマネジメント

- **要件定義：**エンドユーザーやステークホルダーの多様なニーズを特定し、MLイニシアチブの範囲と目的を定義する。
- **優先順位付けとロードマップ策定：**ユーザーのフィードバックと組織の戦略に基づいて機能を優先順位付けし、双方に合致するプロダクトロードマップを作成する。
- **部門横断的なリーダーシップ：**データサイエンス、エンジニアリング、マーケティング、営業などの各チーム間の架け橋となり、明確な製品ビジョンを組織全体に浸透させる。
- **ユーザー体験とフィードバック：**製品リリース後、ユーザーからのフィードバックを収集し、パフォーマンス指標を追跡して将来のアップデートに向けた反復作業を行う。

成功するMLモデル開発には、これらの役割間のシームレスな連携が必要です。残念ながら、DevOpsとMLOpsパイプラインの統合が欠如しているため、エンジニアリング、データサイエンス、運用の各チーム間にサイロが生じています。これらのサイロは、コミュニケーション不足、目標の不一致、展開の遅延を招きます。データサイエンティストは、ソフトウェアエンジニアやDevOpsとの一貫した協力体制がないために、モデルの本番化に苦労したり、インフラ関連のタスクに不必要な時間を費やしたりすることになります。

リーダーシップの認識の重要性

これらのステークホルダーがより良く連携し、AI/MLの取り組みを効果的に統合できるよう支援する責任は、主に開発効率を担うCIO、CTO、または開発リーダーにあります。しかし、AIの機能や特性に対する期待に直接関わるプロダクトオーナーの関与も同様に重要です。これらのリーダーは、ステークホルダーが自律的に、かつ必要に応じて協力して働けるようにするための、効果的なツールやリソースの必要性を伝えなければなりません。

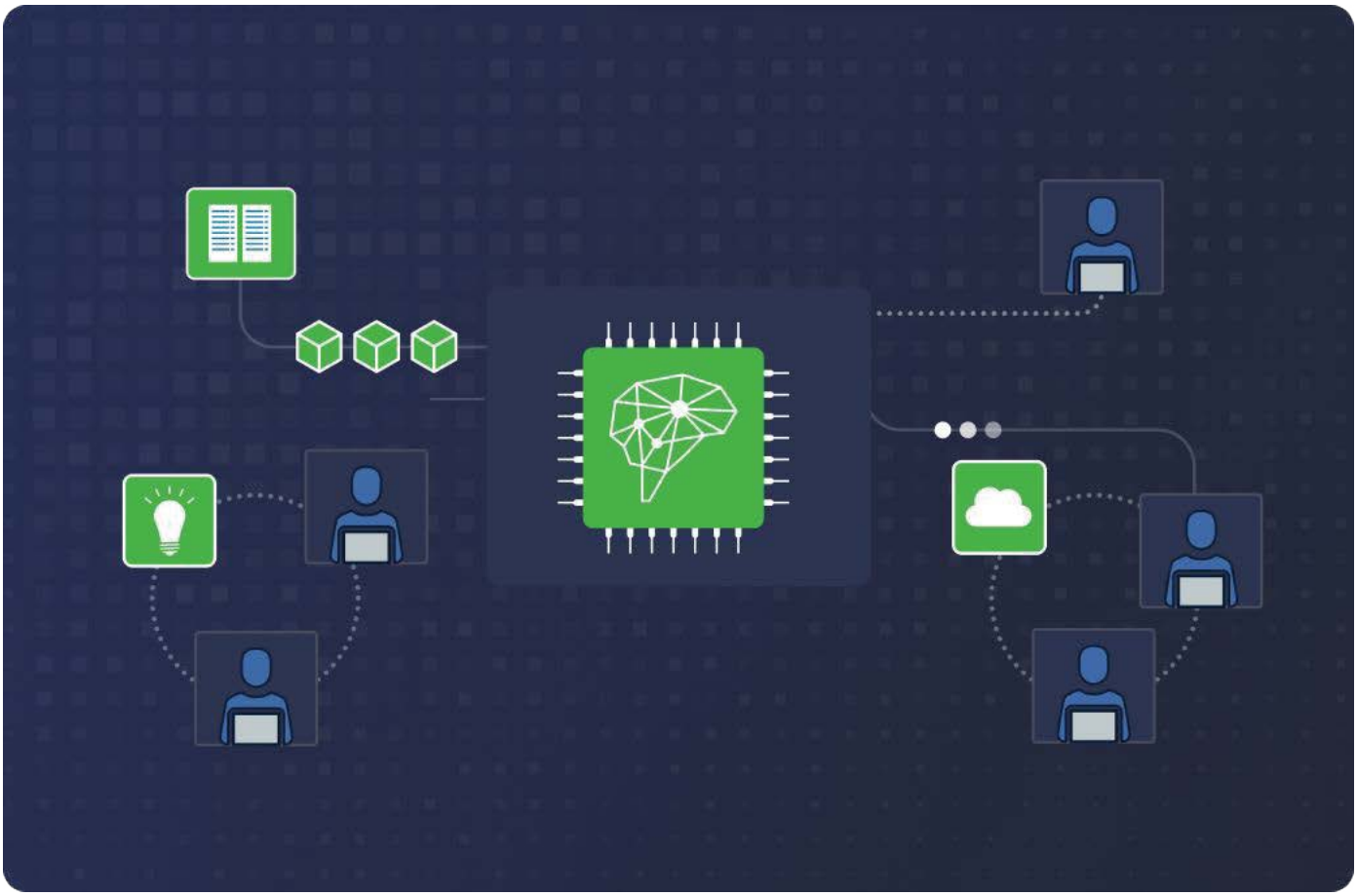
ビジネスリーダーは、AIとMLがもたらす課題を遠い未来の問題だと考えるかもしれませんが、実際には、コンプライアンス、コスト効率、運用の完全性に関連する問題にすぐ直面することになります。言い換えれば、「今こそ行動すべき時」なのです。市場のリーダーたちはすでに早期導入のメリットを享受しており、遅れを取る企業は競争力を失うリスクがあります。

統合されたAI/MLチームの台頭

多くの組織がかなり以前からデータサイエンスを実践したりモデルを公開したりしてきましたが、これらの取り組みは歴史的に、隔離された場所で働く「マッドラボ（風変わりな研究所）」チームに限定されてきました。しかし、アプリケーションがモデルに依存する度合いが強まるにつれ、AIは今やラボから出て、ソフトウェアデリバリーのクリティカルパス（最重要経路）へと入ってきています。

AI/MLの専門知識への需要が高まるにつれ、より多くの組織がAI/MLに特化した専用のチームとインフラを構築しています。すでに運用モデルを本番環境に持っている企業は、AIおよびML専用の役割を設ける傾向が特に強いです。ただし、チームの規模やAI/MLジャーニーにおける成熟度、その他の要因によって、各組織に見られる役割の構成は異なります。

この変化は、効果的なAI/MLの統合には専門知識とリソース、そして継続的な学習と適応へのコミットメントが必要であるという理解が広がっていることを示しています。組織はサイロ化したチームを維持するのではなく、より統合された構造を育む必要があります。適切なドキュメント作成、確立されたプロセス、および可視性の向上が不可欠です。それにより、問題が発生した際、その原因を特定し修正することが容易になります。



統合における課題

組織がMLモデルを業務に統合しようとする際、いくつかの重大な課題に直面します。

1 文化的シフト

歴史的に、データサイエンスチームはサイロ化して運営されており、特定の分析のみに集中する孤立したユニットと見なされがちでした。しかし、モデルを本番化するために開発や運用のプロフェッショナルと緊密に連携する必要が生じ、この状況は劇的に変化しています。

2 取り組みのスケールアップ

多くの組織がAI/MLプロジェクトを開始していますが、それらをスケールさせることは困難です。社内チームには、これらのスケーリングの取り組みを効果的に管理するための適切なツールやプロセスが不足していることがよくあります。

3 モデルの理解と管理

モデルを独立した存在として扱うことはできないという認識が広まっています。MLモデルは既存のソフトウェアに組み込まれることが多いため、既存のソフトウェア開発プロセス内に統合される必要があります。この変化には、組織がモデルのライフサイクルをどう捉え、扱うかという根本的な変革が求められます。

4 標準化の欠如

多くのチームには、MLモデルに関連するバージョニングやメタデータの管理に関する標準的な慣行がありません。この統一感の欠如が、コラボレーションや実行におけるさらなる障壁となります。

5 セキュリティの実装

オープンソースモデルの使用は、OSSパッケージで直面するものと同様の課題（セキュリティ、可用性、バージョニングなど）をもたらします。特にオープンソースモデルのエコシステムは比較的新しく、脅威の状況も不透明です。

非標準的なアプローチのリスク

AIとMLの採用を急ぐ中で、標準化や既存のソフトウェアサプライチェーンプロセスとの統合を欠いたまま進めることには、大きなリスクが伴います。明確なビジネスフレームワークなしにAI/MLモデルの実験を行っている企業は、以下のような数多くの落とし穴に遭遇する可能性が高いです。

1. **データの悪用とガバナンスの問題:**
   モデルは本質的に、既存のデータに基づいて将来の結果を予測する統計方程式です。特に金融のような規制の厳しい業界では、データのコンプライアンスとバイアスの複雑さを乗り越えなければなりません。適切に統治されないモデルの使用は、深刻な評判の低下や法的処罰を招く可能性があります。

2. **インフラの課題**
   高度なモデルのためのインフラを、特に大規模に構築・維持することは困難な課題です。企業はAIアプリケーションの高い需要をサポートするために、適切なリソース、ツール、プロセスを備える必要があります。

3. **運用の冗長性**
   組織は本番環境での避けられない失敗に備えなければなりません。モデルが故障したときに何が起きるでしょうか。シームレスな運用を維持するためには、冗長システムやバックアップモデルの構築が不可欠です。

4. **コスト管理**
   特に企業がAI/MLの取り組みをスケールし始める際、コストを厳格に管理することが重要です。支出を管理するための明確な戦略がない企業は、多方面で予算を急速に使い果たすことになるでしょう。

5. **市場投入までの時間 (Time to Market)**
   スピードは常に重要です。効果的なモデルを本番環境に投入するのが早いほど、企業の競争力は高まります。不明確なプロセスによる躊躇は、動きの速い市場での後退につながります。

6. **セキュリティの懸念**
   従来のソフトウェアアプリケーションのセキュリティ確保における標準化の欠如が非効率性やデータ漏洩を招くのと同様に、機械学習モデルについても同じことが言えます。チームはAI/MLのワークフローを妨げることなく、セキュリティを注入する必要があります。

AI/MLの攻撃対象領域（アタックサーフェス）

セキュリティとコンプライアンスは、AI/MLランドスケープにおける重大な懸念事項です。実際に、JFrogのセキュリティリサーチチームは**Hugging Faceにアップロードされたすべての新しいモデルを分析**しており、これまでに約100個の悪意のあるモデルを特定しています。企業はAI/MLの技術的な側面だけでなく、導入するシステムとプロセスが安全であることを保証しなければなりません。リスクを回避するためには、AI/MLの攻撃対象領域を徹底的に理解することが重要です。

AI/MLの攻撃対象領域は、データ、アルゴリズム、モデル、展開環境など、さまざまな要素で構成されており、それぞれが独自の脆弱性を持っています。

データの完全性：収集、保存、前処理の段階でリスクにさらされます。攻撃者はデータセットを操作してバイアスを導入したり、敵対的サンプル（アドバーサリアルイグザンプル）を混入させたりする可能性があります。
アルゴリズムとモデル：モデル反転（インバージョン）や回避（エベイジョン）などの手法で標的とされる可能性があります。悪意のあるアクターは、モデルのアーキテクチャの弱点を突いて機密情報を抽出したり、予測を誤らせたりします。
展開環境：モデルはしばしばAPIや外部システムと対話するため、攻撃の潜在的な侵入口となります。さらに、AI/MLワークフローでサードパーティのライブラリやツールに依存していると、それらの依存関係に含まれる脆弱性にさらされることになります。

脅威を軽減するために、組織は継続的な監視、データ入力の厳格な検証、強固なアクセス制御を含む包括的なセキュリティフレームワークを採用すべきです。

生成AIとその独自の要件

生成AI（GenAI）は、ニューラルネットワークとディープラーニングを使用して既存データのパターンを特定し、新しいコンテンツを生成する機械学習の一種です。生成AIモデルは、テキスト、音声、画像、アニメーション、さらには3Dモデルまで、あらゆる種類のデータを処理・生成できます。生成AIツールの量と種類が増え、出力の品質が向上するにつれ、ユーザーが使用する大多数のソフトウェアにこれらの機能を期待するようになるのは避けられません。

生成AIアプリケーション開発の課題

生成AIを搭載した製品の作成には、従来のMLモデルトレーニングと同様の段階がありますが、さらに複雑な層が加わります。生成モデルは、連続的なフィードバックと適応を必要とするため、従来の機械学習モデルとは動作が異なります。これらは純粋に数学的なものというより、主観的でユーザー主導になることが多いのです。

生成AIソリューションの開発には、従来の機械学習モデルと比較して、以下のような独自の要件と課題があります。

- **データの品質と多様性：**生成モデルが効果的で正確な出力を出すには、大規模で多様なデータセットが必要です。バイアスがかかっていたり品質が悪かったりするデータは、欠陥のある結果を招き、ツールを事実上使い物にならなくします。
- **モード崩壊と学習の安定性：**敵対的生成ネットワーク（GAN）におけるモード崩壊は出力の多様性を著しく制限し、学習中の不安定さは収束を妨げます。慎重なハイパーパラメータの調整が必要です。
- **評価指標：**生成された出力の品質評価は複雑で、精度のよう単純な指標ではなく、ユーザーのフィードバックやドメイン固有の基準といった主観的な尺度が必要になります。
- **計算リソース：**特に大規模な生成AIモデルは、膨大なハードウェアリソースと長い学習時間を必要とします。これにより、開発コストが高騰し、時間がかかる可能性があります。
- **倫理的考慮事項と法的問題：**フェイクコンテンツ作成への悪用の懸念といった倫理的問題があります。また、学習データの知的財産（IP）や著作権に関連する法的考慮事項も存在します。
- **解釈可能性と制御：**望ましい結果を得るために出力属性を制御する能力は、生成モデルの意思決定プロセスを理解する能力と密接に関連していますが、これは往々にして不透明です。
- **汎用化と過学習：**効果的な汎用化には、多様で高品質な出力を生成することと、学習データへの過学習を避けることの適切なバランスが必要です。
- **スケーラビリティと展開：**生成モデルのサイズとリソース要件に加え、不適切または不要なコンテンツ生成を防ぐための継続的な監視が必要であり、展開において独自の課題を提示します。

実用的なツールとフレームワーク

前述のような生成AI特有の課題に対処するには、AI研究の成果と倫理的考慮事項の両方の手法を取り入れた多角的なアプローチ、およびデータサイエンティスト、ドメインエキスパート、ステークホルダー間の協力が必要です。また、それらを支援するために設計された、目的特化型のツールやフレームワークも増えています。

テストを簡素化するツール

いくつかのツールは、大規模なフロントエンド開発の必要性を減らすことで、データサイエンティストの業務を簡素化します。例えば、Google ColabやHugging FaceのGradio、そしてその他の台頭しつつあるローコード/ノーコードプラットフォームは、AIモデルのためのシンプルなユーザーインターフェースやテスト環境を作成するのに非常に有用です。

埋め込み（エンベディング）

埋め込みは、AI分野におけるもう一つの成長領域です。埋め込みとは、テキスト、画像、音声などの非構造化データを、「ベクトル」として知られる構造化された数値形式で表現する手法です。これらのベクトルはデータの「本質」や「意味論的な意味」を捉えるため、機械学習モデルによる処理や分析が容易になります。これはAIモデルのトレーニングや微調整（ファインチューニング）など、さまざまなアプリケーションで使用できます。

生成AIとMLOpsの統合

MLOpsは、機械学習モデルのライフサイクル管理において極めて重要な役割を果たします。データの準備、モデルのトレーニングから、展開、監視に至るまで、MLOpsはAIモデルの堅牢な機能を保証します。このプロセスには、データの準備、モデルのトレーニングと微調整、展開、および展開後の継続的な監視が含まれます。特に生成モデルにおいては、標準的なパフォーマンス監視だけでなく、出力を改善するためのユーザーフィードバックの収集も必要とされます。

MLOpsの自動化

MLOpsを自動化することは、チームが生成AI開発の課題を克服する上で大きな助けとなり、以下のような無数のメリットをもたらします。

効率の向上、モデル精度の改善、スケーラビリティ、再現性、コラボレーションの強化、市場投入の迅速化（Time to Marketの短縮）、コスト削減、コンプライアンスとセキュリティの向上

データソース、ツール、統合機能、および環境を統合することで、チームはAI/MLワークフローをスケールさせることができ、生成AIモデルが堅牢で信頼性が高く、高品質な出力を提供できることを保証できます。

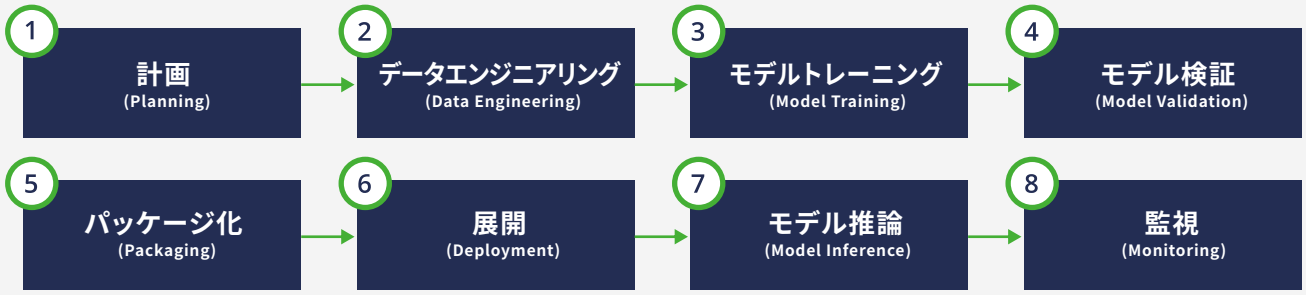
将来のAI、特に生成AIの未来が可能性に満ちていることは明らかです。生成AIは、私たちがテクノロジーについて考え、対話する方法を効果的に再形成しています。この分野における継続的な進歩は、さらなる革新的なツールやアプリケーションをもたらすことを約束しています。

AI/MLのための統合ソフトウェアサプライチェーン

AI/ML開発固有の課題とリスクに直面している組織には、主に2つの選択肢があります。複雑さを管理するために専門の人員を雇うか、さまざまなニーズに対応する多様なツールを組み立てることに投資するかです。しかし、人か技術のどちらか一方だけに投資しても、十分なインパクトは得られません。DevOpsと同様に、人、技術、プロセスの適切な組み合わせが必要です。これが「MLOps」の実践につながります。

実践的なMLOps

MLOps（Machine Learning Operations）とは、データサイエンスと運用のギャップを埋め、MLモデルをソフトウェア開発ライフサイクル（SDLC）の各段階を通じて移動させるために設計された慣行とツールを指します。DevOpsムーブメントによって最初に導入された手法に触発されたMLOpsは、本番環境でのAIアプリケーションとMLモデルの展開、管理、監視のプロセスをエンドツーエンドで自動化し、簡素化することを目指しています。



ML開発ライフサイクル

MLOpsを採用することで、組織はMLライフサイクル全体のプロセスを標準化および合理化し、ワークフローと展開の効率と一貫性を高めることができます。このアプローチは、運用を最適化するだけでなく、本番環境内でのMLモデルのスケール可能で持続可能な統合をサポートします。

MLOpsの主要構成要素

コラボレーションとコミュニケーション
データサイエンティスト、エンジニア、運用チーム間のシームレスな情報共有を促進。

モデルの監視とオブザーバビリティ
モデル性能の監視、ドリフト検知、バージョン管理のプロセスを構築。

バージョン管理
コード、データ、モデル構成の変更を追跡し、再現性と追跡可能性を確保。

セキュリティとコンプライアンス
セキュリティ懸念を管理し、規制への準拠を確実に。

継続的インテグレーション/継続的デリバリー (CI/CD)
自動テスト、構築、展開のためのパイプラインを開発。

フィードバックループと反復
本番環境からのフィードバックを収集し、モデルを反復的に改善する仕組みを構築。

組織はアプローチを標準化するのを待つべきではありません。独自の状況に適応できる即時のソリューションが必要です。この積極的なアプローチを取ることで、市場で差別化を図ることができます。

ML、運用、セキュリティのギャップを埋める

通常、従来のソフトウェア開発とAI/ML開発は、別々のワークフロー、ツール、目的で運営されています。残念ながら、DevOpsとMLOpsのパイプラインを分離したままにする傾向は、ソフトウェアデリバリーに悪影響を及ぼす無数の非効率と冗長性を招きます。

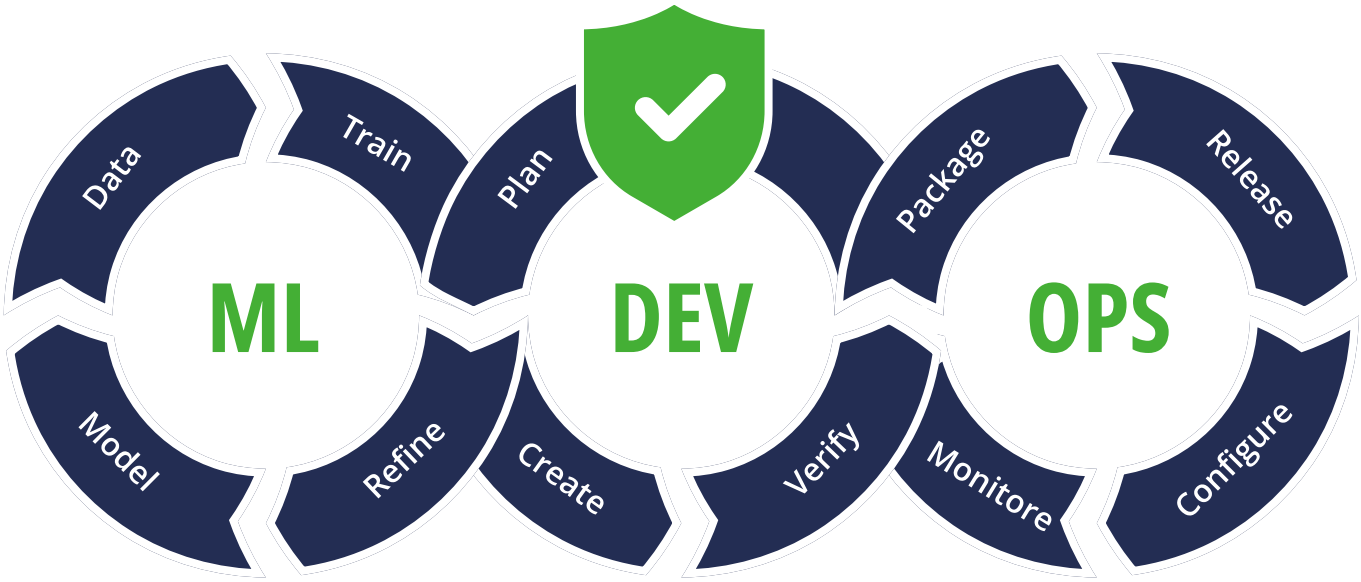
MLモデルをパッケージとして扱う

すべてのAI駆動型アプリケーションの核心には、それを動かす「モデル」があります。このモデルは、本質的にはソフトウェアバイナリであり、高品質なソフトウェアアプリケーションと同様に、セキュリティを確保し、管理し、追跡し、展開する必要があります。

多くのデータサイエンティストは、自分たちの仕事を簡素化されることに抵抗を感じることがあります。なぜなら、彼らは複雑なモデルを中心にキャリアを築いてきたからです。しかし、モデルがソフトウェアコンポーネントと同じように扱えると認識すれば、その使用に関する不安は解消されます。開発者がコンパイラの複雑な詳細をすべて理解する必要がないのと同様に、データサイエンティストもアルゴリズムに固執するのではなく、実用的な成果に集中すべきです。開発や運用のプロフェッショナルも、これらの新しい技術やその背後にある科学に臆する必要はありません。モデルは「単なるひとつのパッケージ」であり、そのように扱われるべきなのです。

MLOpsとDevOpsパイプラインを統合するJFrog ML

JFrogは、既存の開発、セキュリティ、運用の原則の下で、MLソフトウェアとアーティファクトの管理を「正常化（ノーマライズ）」することを目指しています。この包括的な管理により、ソフトウェア開発で採用されているものと同じベストプラクティスをAI/MLにも適用できるようになります。AI/MLをSDLC全体の一部として扱うことで、組織はサイロを減らし、部門間のコラボレーションを強化できます。



JFrogは、ソフトウェアとMLアーティファクトの両方を統合されたフレームワークの下で管理するソリューションを提供し、従来のソフトウェア開発と最新のMLプラクティスの間のギャップを埋める独自のポジションにあります。

モデルレジストリ – 高度なモデルレジストリとしてのJFrog Artifactory

- 単一の集中レジストリでモデルのライフサイクル全体を管理
- トレーニングパラメータ、ハイパーパラメータ調整、モデルメタデータの可視化
- モデルと従来のソフトウェアアーティファクトを同一システムで管理
- DevOpsとMLOpsを単一の信頼できる情報源（Source of Truth）に統合

モデルセキュリティ – 信頼できるAI/MLコンポーネントの提供

- 悪意のあるモデルの検知とブロック
- モデルの脆弱性およびライセンスのスキャン
- セキュリティとコンプライアンスポリシーの自動強制

MLOps – あらゆるモデルのトレーニングと展開

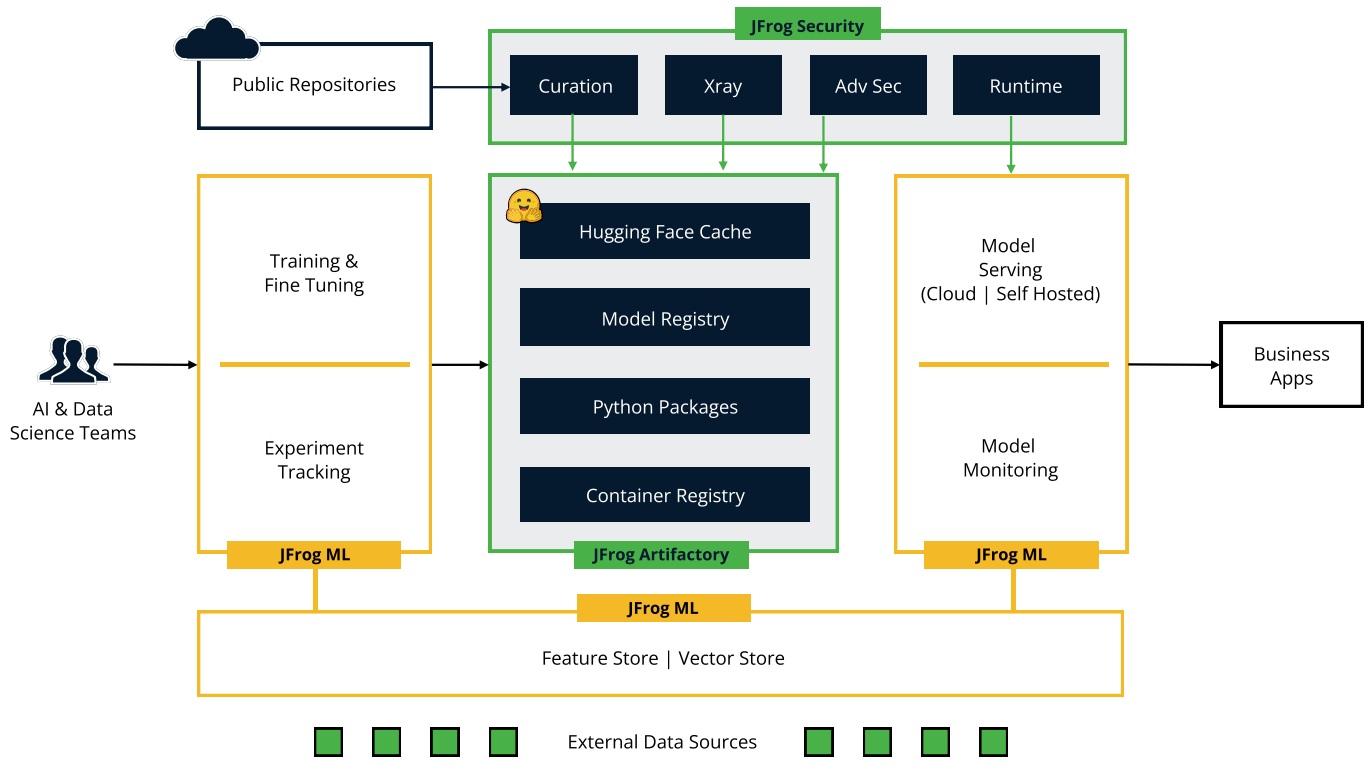
- 研究から本番まで、すべてのモデルを管理
- ワンクリックでモデルのトレーニングと微調整を実施
- ライブAPIエンドポイントからKafkaストリームまで、あらゆる規模で展開
- リアルタイムの監視とアラート

LLMOps – LLM（大規模言語モデル）アプリケーションの開発

- プロンプト管理とバージョン追跡
- JFrog最適化済みのオープンソースLLMの展開
- 複雑なLLMワークフローの作成と可視化
- デバッグを容易にするための全プロセスの追跡
- 本番環境のLLMを監視し、最適なパフォーマンスを維持
- 埋め込みベクトル（embedding vectors）の大規模な保存と管理

Data Ops – データの変換

- トレーニングと推論のためのすべての特徴（feature）を管理
- あらゆるソースからのデータの取り込みと処理
- 堅牢な特徴量エンジニアリング・パイプラインの構築



今後の展望

組織のフレームワークへのMLおよびAIの統合は、機会と課題の両方をもたらします。AI/MLと従来のソフトウェア管理の実践を一致させつつ、独自のセキュリティおよび運用の要件に対処する、統合されたアプローチを採用することが不可欠です。標準化、ガバナンス、戦略的計画に焦点を当ててこの変革の旅に取り組む組織は、成功に向けたより良いポジションを確保できるでしょう。

チーム間のコラボレーションを促進し、セキュリティを優先し、堅牢なデータガバナンスを確保することで、企業はAIとMLの複雑さを乗り越えることができます。JFrogは、組織が関連するリスクを軽減しながら、これらのイニシアチブを立ち上げ、スケールさせるための準備を整えています。AIとMLを組織のソフトウェア開発戦略に統合する方法や、MLモデル管理のためのJFrogプラットフォームの詳細についてご興味がある場合は、今すぐ**デモをご予約**ください。