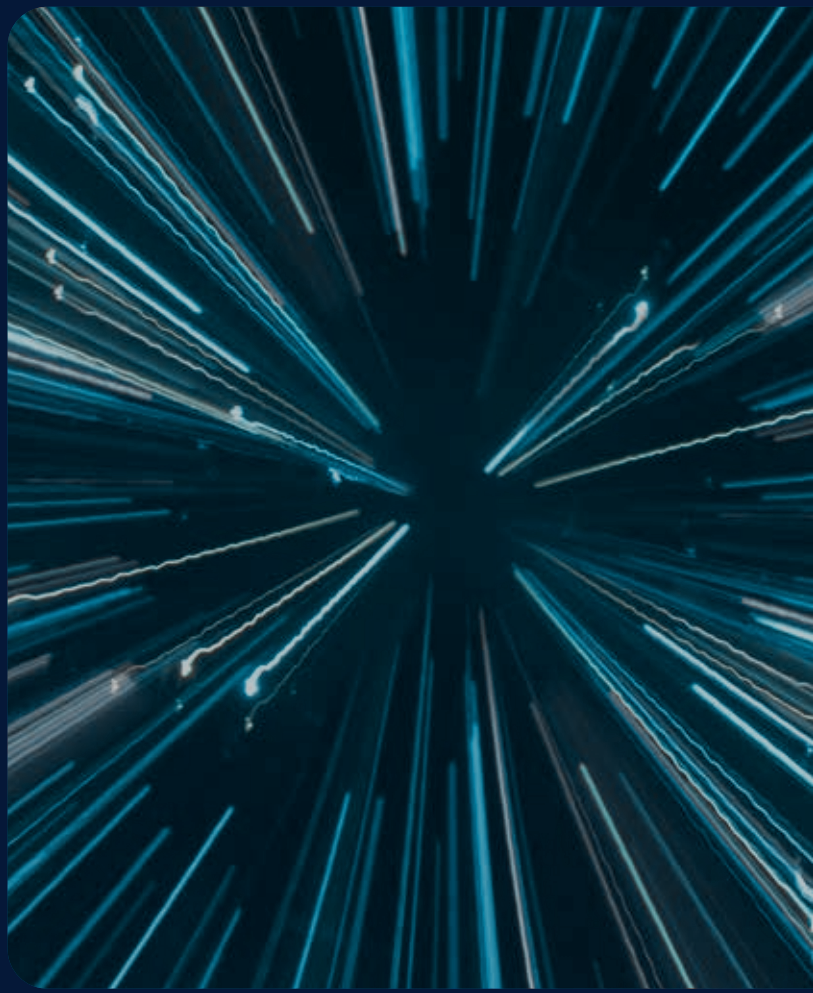
実践ガイド

リアルタイムの顧客データで AIレコメンデーションを強化

TEALIUM®

目次



2024年、レコメンデーションエンジンはeコマースの全注文の19%に影響を与え、ホリデーシーズンだけで世界のオンライン売上高を2,290億ドル押し上げました。この金額は今後も伸び続けるでしょう。ですが、ほとんどの企業が触れない点があります。注目を集めるのはレコメンデーションモデルである一方、実際にそれが機能するかどうかを左右するのはデータパイプラインだということです。

バッチ処理によるデータで学習したレコメンデーションエンジンは、6時間前の顧客に対して判断を下しているようなものです。今まさに貴社のサイトを閲覧し、商品をカートに追加し、競合サイトのタブで価格を比較している顧客についてはどうでしょうか？モデルはそのことをまだ何も把握していません。モデルがそれを把握する頃には、その顧客の購買意図はすでに失われています。本来ならコンバージョンにつながったはずのオファーも、ただのノイズになってしまいます。

本ガイドは、AIレコメンデーションに投資すべきか？という検討段階を終え、具体的にどのようなアーキテクチャを構築すべきか？という段階に入った企業に向けた内容となっています。行動イベントの収集からレコメンデーションの提示に至るまでのパイプライン全体を網羅し、各ステージにおける具体的な技術パターン、実際のベンチマーク、そして避けては通れないトレードオフについて解説します。

リアルタイムのAIレコメンデーションに真に必要なもの

p99レイテンシーが100ミリ秒未満。これが2026年のレコメンデーション配信の本番環境における基準です。これより遅くなると、顧客は遅延を感じます。さらに大幅に遅くなると、レコメンデーションを提示すべき瞬間を逃してしまいます。

この基準を満たすには、4つの要素が連動して機能する必要があります。それは、行動シグナルを発生と同時に捕捉するストリーミングデータ取り込み、それらのシグナルを即座にクエリ可能にする特徴量ストア、候補を1桁ミリ秒単位でスコアリングできる十分に高速なモデル推論、そしてコンテキストが変化する前に適切なチャネルへレコメンデーションを配信するアクティベーションレイヤーです。

これらは、バッチ処理アーキテクチャに後付けできるオプションの強化機能ではありません。1時間単位の同期を前提に設計されたシステムに、後付けでリアルタイム機能を組み込むことはできません。これは、アーキテクチャ上の決定事項であり、単なる機能のオン／オフ切り替えではありません。バッチ処理パイプラインに段階的にリアルタイム機能を追加しようとする企業は、多くの場合、ストリーミング処理の複雑さとバッチ処理の遅延という、両方の欠点を併せ持つ最悪の状況に陥ってしまいます。これはこの分野で最もよくある失敗の一つであり、なぜそれが機能しないのかを完全に理解するには、通常6か月を要し、ホリデーシーズンの商機を1回は逃すことになります。

以降では、各レイヤーについて、実際の運用で直面する具体的な技術選定、ベンチマーク、トレードオフを順を追って解説します。

リアルタイムデータパイプライン：シグナルからレコメンデーションまで

リアルタイムのレコメンデーションパイプラインは、6つのステージで構成されます。各ステージで遅延が発生するため、全体のレイテンシー予算は100ミリ秒未満です。

ステージ　1：イベント捕捉。あらゆる顧客接点は、ページビュー、商品クリック、カート追加、購入、スクロール深度、検索クエリ、さらにはセッションのタイミングパターンといったシグナルを生成します。これらのイベントは発生源で捕捉され、一貫した構造に整えられた上でストリームとして送信される必要があります。TealiumのEventStreamは、Web、モバイル、サーバーサイドの各ソースからイベントをリアルタイムで取り込み、どこで発生したインタラクションであっても一貫した構造のレコードを生成することで、これを処理します。

ステージ 2：ストリーミング取り込み。イベントはApache Kafka、Amazon Kinesis、Google Cloud Pub/Subなどのストリーミングプラットフォームへ流入します。ここで重要なのは、高負荷時のスループットです。ブラックフライデー、新商品発売、フラッシュセールのようなトラフィックのピーク時でも、取り込みレイヤーはデータ欠損やバックプレッシャーを起こすことなく、毎秒数百万件のイベントを処理できなければなりません。

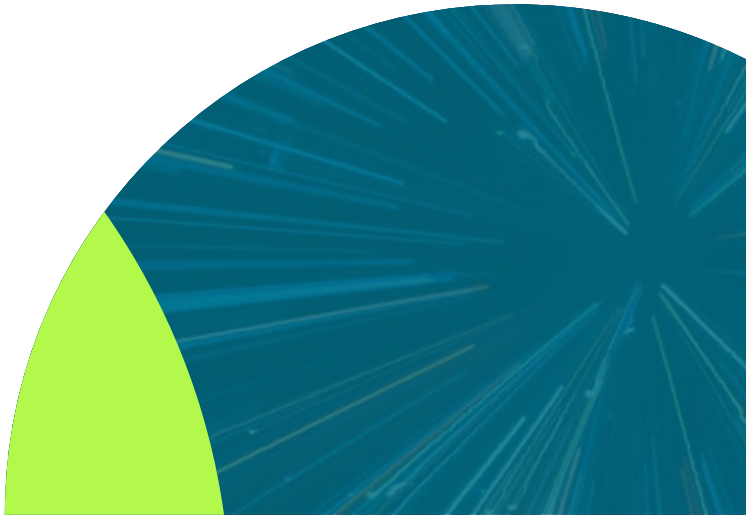
ステージ　3：リアルタイムエンリッチメント。生のイベントデータは有用ですが、それだけでは不十分です。エンリッチメントは、デバイス間でのID解決、顧客セグメントへの所属情報の付加、類似性マッチングのための埋め込み表現の計算、同意ステータスの判定など、ビジネスコンテキストを処理の途中で付加します。このステップによって、生のクリックストリームデータは、追加変換なしでそのままモデルが利用できる、構造化され、エンリッチされたAI対応データへと変換されます。

ステージ　4：特徴量ストア。エンリッチされた特徴量は、即座に照会可能な低レイテンシーの特徴量ストアに格納されます。特徴量ストアは、データパイプラインとモデルをつなぐ橋渡し役です。そこには、ストリーミングデータから継続的に更新されるリアルタイム特徴量と、履歴データから定期的に計算されるバッチ特徴量の両方が格納されます。これについては後述のセクションで詳しく説明します。

ステージ　5：モデル推論。レコメンデーションのリクエストが発生すると、配信レイヤーは顧客の最新特徴量を取得し、候補アイテムをモデルでスコアリングし、結果を順位付けします。ネットワークのオーバーヘッドや後処理の時間を確保するため、このステップは通常20ミリ秒以内で完了する必要があります。

ステージ　6：アクティベーション。順位付けされたレコメンデーションは、顧客が利用しているチャネルを通して届けられます。Webサイトのパーソナライゼーション、モバイルプッシュ通知、電子メール、店舗スタッフ向け画面、コールセンターのオペレータ用ダッシュボードなどです。アクティベーションレイヤーは、同一の顧客が複数のチャネルをまたいで同時に接点を持つ可能性があるため、それらのチャネルを同時にサポートする必要があります。

適切に設計されたシステムであれば、イベントの捕捉からレコメンデーションの提示に至る一連の処理は、100ミリ秒未満で完了します。Netflixは、これを2億3,000万人の同時接続ユーザー向けに実現しています。こうしたアーキテクチャパターンはすでに実証済みです。問題は、自社のデータ基盤がそれらを支えられるかどうかです。



あらゆるチャネルにわたる行動データの収集

Webでは、ページビュー、商品インタラクション（閲覧、クリック、カート追加、ウィッシュリスト登録）、検索クエリやその絞り込み操作、スクロール深度、セッション時間、参照元、離脱時の行動などを収集します。モバイルでは、これらに加えて、プッシュ通知の開封、アプリ内メッセージへの反応、そして同意取得済みの場合、位置情報シグナルといったアプリ固有のイベントも収集されます。IoTやコネクテッドデバイスからは、操作イベント、利用パターン、状態変化などのデータが得られます。オフラインチャネルからは、POS取引、コールセンターでのやり取り、店舗内キオスク端末の利用状況、ロイヤルティプログラムへの参加状況などの情報が収集されます。

難しいのは、これらの一つひとつを捉えることではありません。重要なのは、チャネルを横断してIDを紐付け、一貫したスキーマでこれらすべてをリアルタイムに収集することです。例えば、スマホで靴を閲覧し、デスクトップPCでレビューを調べ、実店舗に足を運ぶ顧客は、単一の意図を持った、同一人物です。もしデータパイプラインがこれらを3つの匿名セッションとして扱ってしまうと、レコメンデーションモデルは本来あるべきシグナルのごく一部しか利用できなくなります。

ここで基盤となるのが、ID解決となります。TealiumのAudienceStream CDPは、チャネルを横断するIDをリアルタイムに解決し、デバイスID、クッキー、ログインイベント、CRMレコードなどを単一の統合プロファイルにまとめます。レコメンデーションモデルが顧客の特徴を照会する際、デバイスごとの断片的な情報ではなく、全体像を把握することが可能になります。

データのエンリッチメントも重要です。取り込み時に、顧客生涯価値、所属セグメント、同意ステータス、リアルタイムの意図スコアなどのビジネスコンテキストを付加することで、特徴量ストアには、モデルでの利用に適した形式で構造化されたデータが格納されることになります。クエリ実行時にエンリッチメントを行うよりも、データ処理の過程でエンリッチメントを行う方がはるかに効率的です。推論のたびに繰り返し計算コストを支払うのではなく、上流工程で一度だけ計算コストを負担すれば済むからです。

低レイテンシな特徴量ストアの構築

特徴量ストアは、多くの企業が最も過小評価しがちなコンポーネントです。これはデータパイプラインとモデルの間に配置されており、その性能がレイテンシ目標を達成できるかどうかを直接左右します。

特徴量ストアは2種類の特徴量を管理します。**リアルタイム特徴量**は、ストリーミングデータから継続的に更新されます。これには、顧客の現在のセッション行動、アクティブなカート内の商品、直近5分間に閲覧されたページなどが該当します。**バッチ特徴量**は、履歴データから定期的に計算されます。これには、生涯購買履歴、長期的な嗜好パターン、人口統計学的属性などが含まれます。どちらのタイプも、推論時には1桁ミリ秒以内で取得できる必要があります。

採用する技術は、規模や低遅延の要件に応じて多岐にわたります。

Redisは、リアルタイムの特徴量提供において最も一般的な選択肢です。10億個の768次元ベクトルを用いたRedis 8のベクトル検索ベンチマークでは、50件の同時クエリに対し90%の精度で、中央値で約200ミリ秒の低遅延を記録し、毎秒66,000件のベクトル挿入を維持できることが示されています。一方、ベクトル検索ではなく特徴量ルックアップの場合、Redisはサブミリ秒の読み取りを実現します。Relevance AIは、Redisへの移行後、ベクトル検索の遅延を2秒から10ミリ秒へ短縮したと報告されています。

ベクトルデータベース(Pinecone、Weaviate、Qdrantなど)は、埋め込みベースのレコメンデーションにおける類似検索に特化しています。これらは、「このユーザーの嗜好ベクトルに最も近いアイテムを見つける」というクエリパターンのために設計されています。トレードオフとして、ベクトル演算には最適化されていますが、汎用的な特徴量提供においては柔軟性に欠けます。

キャッシュレイヤーを備えたクラウドデータウェアハウス(BigQueryとBigtableの併用、SnowflakeとRedisキャッシュの併用など)は、既存のデータウェアハウスを記録システムとして維持しつつ、高速な配信レイヤーを追加したい企業に適しています。アーキテクチャはより複雑になりますが、特徴量パイプライン全体を新しいシステムへ移行せずに済みます。

セマンティックキャッシュは、特に注目する価値があります。レコメンデーションクエリに意味的な類似性が認められる場合(業界全体では約31%のクエリがこれに該当)、セマンティックキャッシュは推論を再実行する代わりに、過去の結果を返すことができます。その効果は絶大で、AWSの実験ではセマンティックキャッシュによりコストは86%削減され、遅延が88%改善されたと報告されています。あるヘルスケア企業では70%のキャッシュヒット率を達成し、LLM推論コストを70%削減できました。Redis LangCacheは、フルのモデル推論を実行する場合と比較して、キャッシュヒット時の応答速度を最大で15倍高速化します。

ただし、積極的なキャッシュ運用には鮮度という、見過ごされがちなトレードオフがあります。30秒前にキャッシュされたレコメンデーションであれば、ほぼ問題はありません。ですが、30分前にキャッシュされたレコメンデーションは、すでに古くなった顧客の意図を反映している可能性があります。問題は、キャッシュの鮮度低下がコンバージョン率を密かに悪化させ続けていても、集計データ上ではその変化に気づきにくいという点です。キャッシュのTTL戦略は、コスト削減とレコメンデーションの関連性のバランスを取る必要があります。その最適解は、顧客の意図がどの程度の早さで変化するかによって異なります。まずは保守的に短いTTLから始め、影響を測定しながら徐々に延ばしていくのがよいでしょう。

ハイブリッドモデル：レコメンデーション精度の最適化

あらゆるレコメンデーションのシナリオにおいて、万能な単一のモデリング手法は存在しません。本番環境で優れたパフォーマンスを発揮するシステムは、複数の手法を組み合わせています。

協調フィルタリングは、例えば、Xという商品を買った顧客はYという商品も購入している、といったユーザー間のパターンを分析します。これはインタラクションデータが豊富にある場合には有効ですが、新規ユーザーにおけるコールドスタート問題や、まだ十分なインタラクション履歴が蓄積されていない新規アイテムには対応が難しい場合があります。

コンテンツベースフィルタリングは、その逆のアプローチを取ります。ユーザーの行動パターンを調べるのではなく、ブランド、価格帯、カテゴリ、商品説明のキーワードといったアイテム属性を分析します。「この商品は、あなたが購入した商品と共通した特徴を持っています」という考え方です。新規アイテムであっても最初から属性情報は備わっているため、ここではコールドスタート問題は生じません。ただし欠点は視野の狭さです。純粋なコンテンツベース型システムは、ユーザーがすでに好んでいるものをさらに推奨する傾向があります。一見役に立ちそうですが、やがてパーソナライズされた体験において、鏡の迷宮のように同じようなものばかりが目につくようになってしまいます。

ハイブリッドアプローチは、この両方を組み合わせたものです。実際、多くの本番システムがこの方式を採用しています。NetflixやAmazonのレコメンデーションエンジンもハイブリッドシステムです。近年はツータワーアーキテクチャが一般的なパターンになっています。これは、片方のニューラルネットワークタワーがユーザーの特徴量をエンコードし、もう片方がアイテムの特徴量をエンコードし、2つの埋め込みベクトルを比較することで関連性を予測するようにモデルを学習させる仕組みです。このアーキテクチャは、アイテムの埋め込みはオフラインで事前計算し、ユーザーの埋め込みのみをリアルタイムで計算できるため、大規模なカタログに対しても優れたスケーラビリティを発揮します。

手法と共に重要となるのがコンテキストです。時間帯、デバイスの種類、位置情報、セッションの深さ、直近のインタラクションなど、あらゆる要素が、どのレコメンデーションが適切かに影響を与えます。例えば、仕事の合間の午前9時にデスクトップPCで閲覧している顧客と、その顧客が午後9時にスマートフォンで閲覧している場合とでは、意図が異なります。モデルは、このようなシナリオを区別するために、特徴量ストアからのリアルタイムなコンテキスト特徴量を必要とします。

モデルアーキテクチャに関する実務上の注意点を一つ挙げると、ディープラーニングの手法、特にニューラル協調フィルタリングは、オフラインでの評価指標において優れた性能を発揮します。しかし、Google　Researchは、大規模な本番環境でのサービス提供においては、計算コストが高くなりすぎる可能性があることを指摘しています。そのため、多くの本番システムでは、ディープラーニングは、数百万点のアイテムを数百点に絞り込む工程の候補生成に使用し、最終的なランキングにはより単純で高速なモデルを使用しています。このような2段階アプローチにより、応答速度の許容範囲を超えずに、ディープラーニングによる精度の高さを活かすことが可能になります。

大規模な環境でのレコメンデーション配信

モデルの学習が完了し、特徴量ストアも高速に動作しています。次に、レイテンシSLAを満たしながら、数百万人の同時利用ユーザーに対してレコメンデーションを配信する必要があります。

配信アーキテクチャは、一般的に次のような構成になります。ユーザーの行動がレコメンデーションの要求をトリガーします。配信レイヤーが特徴量ストアからユーザーの最新特徴量を1〜5ミリ秒で取得し、モデルが候補アイテムを5〜20ミリ秒でスコアリングし、順位付けを行います。その後、多様性制約、在庫状況、表示頻度制御、販促ブーストなどのビジネスルールによる後処理が2〜5ミリ秒で適用されます。最終的に応答がクライアントへ返されます。

推論の合計所要時間は10〜30ミリ秒であり、ネットワークのオーバーヘッドを考慮してもエンドツーエンドで100ミリ秒という許容範囲に対し十分な余裕があります。

これをスケールさせるには、いくつかの特定の手法が必要となります。

連続バッチ処理は、一定数のリクエストが揃うのを待たずに、リクエストキューを動的に管理します。従来のバッチ処理はスループットを最適化する一方で、急激な遅延を招くことがあります。連続バッチ処理は、スループットを維持しながら、テールレイテンシを予測可能な範囲に保ちます。

モデルの量子化は、精度をわずかに低下させるのと引き換えに、大幅な処理の高速化を実現する手法です。FP32からFP16へ変換すると、推論速度は2倍になります。NVIDIA H100 GPUでサポートされているFP8では4倍の高速化が実現します。エッジ環境での展開に有効なINT4の量子化では、精度の劣化を適切に管理することで、10〜15倍の高速化も可能です。

分離推論は、候補生成とランキングのステージを、それぞれの負荷に最適化された異なる計算リソースへ分離します。NetflixやSpotifyがグローバル規模で100ミリ秒未満のレイテンシを維持できているのは、この手法を採用しているためです。

本番環境への展開にあたっては、初日から全てのトラフィックに対しデプロイするといった衝動は抑えてください。まずは小規模なセグメントから始めるべきです。既存のシステム、あるいはレコメンデーションを一切提示しない状態で、コンバージョン率、クリック率、セッションあたり売上といった明確に定義された成功指標を用いて、A／Bテストを実施します。また、ユーザー行動が学習時のデータ分布から変化し、時間の経過と共に性能が低下するモデルドリフトを監視してください。また、モデル自身のレコメンデーションが将来の学習データに影響を与え、レコメンデーションの多様性が失われるようなフィードバックループが発生していないかどうかにも注意を払う必要があります。

SHAPやLIMEといった説明可能性ツールは、モデルがなぜ特定のレコメンデーションを行ったのかを理解するのに役立ちます。これはデバッグはもちろん、事業に関わる関係者との信頼構築や、ますます重要になっている規制遵守の観点からも重要です。顧客から、なぜ特定の商品が表示されたのですか？と尋ねられた際に、モデルがそう推論したからです、という回答では受け入れてもらえません。

真に重要な指標の測定

必要な指標は、大きく2つに分かれます。ビジネス成果とシステムの健全性です。

ビジネス指標は、レコメンデーションが機能しているかどうかを示します。推奨商品と非推奨商品のコンバージョン率（CVR）の比較は、最も直接的なシグナルとなります。レコメンデーションあり・なしのセッションにおける平均注文額（AOV）を比較することで、エンジンが買い物カゴの総額を拡大しているのか、それとも既存の支出を単に振り替えているだけなのかが分かります。レコメンデーションウィジェットのクリック率（CTR）は、顧客が提示された商品を認識しているかどうかを測る指標となります。また、購入経路のどこかでレコメンデーションが関与した総売上の割合である売上寄与率を見ることで、この仕組みがどれほど売上を支える存在になっているかが分かります。高度な実装では、商品レコメンデーションがECサイトの売上の最大31%を生み出しています。もしその割合が1桁台にとどまっているなら、まだ成長の余地があります。

システム指標は、インフラが健全かどうかを示します。レコメンデーション配信におけるp50およびp99のレイテンシを追跡してください。キャッシュヒット率も追跡します。セマンティックキャッシュを運用している場合、70%以上のヒット率は効率性の強力な指標です。特徴量の鮮度、つまり推論実行時点で最新の特徴量更新がどれだけ古いかも確認する必要があります。「リアルタイム」の特徴量が常に30秒遅れている場合は、パイプラインのどこかにボトルネックが存在します。時間経過に伴うモデルの精度も監視し、ドリフトを早期に検知することが重要です。

これら両方のカテゴリに対して、リアルタイムのダッシュボードを構築してください。レイテンシの急上昇、精度の低下、またはCTRの急激な変化に対する自動アラートがあれば、顧客に影響を与えるような障害になる前に問題を検知するのに役立ちます。レコメンデーションエンジンは収益を生み出すシステムです。その監視は、決済処理パイプラインに適用するのと同等の厳格さで行ってください。

プライバシー、同意、ガバナンス

この規模でのリアルタイムの行動データを収集することには、現実的なプライバシー上の義務が伴います。これらは、コンプライアンス上の後付け事項ではなく、アーキテクチャ上の要件として扱う必要があります。

EU一般データ保護規則（GDPR）、カリフォルニア州消費者プライバシー法（CCPA）、および医療保険の携行性と責任に関する法律（HIPAA）のような業界固有の規制は、データ収集に対する明示的な同意、データの使用方法の透明性のある開示、個人データへのアクセスおよび削除の権利、ならびにデータ保持期間の制限といった明確な要件を定めています。これらは例外的なケースではありません。これらは、レコメンデーションパイプラインを流れるすべての行動イベントに適用されます。

アーキテクチャ上の意味は明確です。同意ステータスは、事後的に確認されるものではなく、データとともに伝播される必要があります。イベントがストリーミングパイプラインに入ってきた時点で、取り込み時に同意の処理が完了している必要があります。レコメンデーション関連のデータ利用に同意していないユーザーのイベントは、決して特徴量ストアやモデル学習パイプラインに流入させてはなりません。Tealiumの同意管理アーキテクチャは、データレイヤーでこれを強制し、イベントが処理される際にリアルタイムで同意設定を評価します。これは単なるコンプライアンスのチェックボックスではなく、サーキットブレーカーのようなものです。同意を得ていないデータは、システムが通過させないため、下流に伝播することはありません。

規制コンプライアンスに加えて、説明可能性とオプトアウトの仕組みは顧客の信頼を築きます。顧客がなぜそのレコメンデーションを受け取ったのかを理解でき、かつレコメンデーションに基づく体験からオプトアウトできるのであれば、パーソナライゼーションを損なう「気味の悪さ」を和らげることができます。レコメンデーションは、監視されていると感じさせるものではなく、役立つと感じられるものであるべきです。

データガバナンスは、モデルへの入力も対象とします。レコメンデーションモデルがどの特徴量を利用しているか、それらの特徴量にどのデータソースが供給されているか、そしてそれぞれにどの同意根拠が適用されるかを文書化してください。規制当局や内部監査担当者から、AIがどのように意思決定を行っているかを問われた場合、生のイベントから配信されたレコメンデーションに至るまでの完全なデータ系譜を追跡できる必要があります。このトレーサビリティを最初から構築するには数週間かかります。本番環境で1年間運用した後で後付けで導入しようとすると、数ヶ月を要し、どのデータがどこへ流れたのかを巡る気まずい議論を交わさなければならなくなります。

スタック全体でレコメンデーションをアクティベート

自社ウェブサイトでしか機能しないレコメンデーションでは、価値を十分に活かしきれていないことになります。ウェブでのパーソナライゼーションを支えるのと同じ顧客プロファイルが、電子メールコンテンツ、モバイルプッシュ通知のタイミング、コールセンターのオペレーターへの指示、そして有料広告の配信抑制にも活用されるべきです。

これを実現するには2つの要素が必要です。チャネルを横断して利用可能な統合された顧客プロファイルと、顧客が企業やブランドと接するあらゆるシステムでレコメンデーションをアクティベートできる統合レイヤーです。

実際のアクティベーションフローは以下の通りです。AudienceStreamが、リアルタイムの行動データで継続的に更新される統合プロファイルを維持します。レコメンデーションモデルが結果を生成すると、その結果、あるいはそれに基づいて定義されるオーディエンスセグメントは、電子メールプラットフォーム、広告ネットワーク、CMS、モバイルエンゲージメントツール、カスタマーサービスプラットフォームへ届けられる必要があります。Tealiumは1,300以上の下流プラットフォームと連携できるため、アクティベーションレイヤーをチャネルごとに個別に構築する必要はありません。同じレコメンデーションシグナルで、パーソナライズされた電子メールの配信をトリガーしたり、すでにコンバージョン済みの顧客へのリターゲティング広告の表示を抑制したり、次回のウェブサイト訪問時に関連商品を表示したりすることができます。

運用面では、オーディエンス管理を一元化することにより、レコメンデーションの一貫性を損なう断片化を防ぐことができます。電子メールチーム、ウェブチーム、有料広告チームがそれぞれ独自にレコメンデーションロジックを構築していると、顧客は矛盾した体験をすることになります。共通の統合レイヤーを通してチャネル横断的にアクティベーションされる単一のレコメンデーションシグナルにより、顧客体験の一貫性を保てます。

早期に実装する価値のあるパターンの一つが、チャネルのコンテキストを考慮した次善のアクションロジックです。例えば、午前2時にモバイルプッシュ通知で届く商品レコメンデーションは、パーソナライゼーションではなく、単なる邪魔に過ぎません。一方、同じレコメンデーションでも、顧客が翌朝にアプリを開いた際に表示されれば、有益な提案になります。リアルタイムの行動シグナルに基づく、チャネル認識型のアクティベーションルールが、この差を生み出します。

今後の展望：エージェンティックAIとレコメンデーションの未来

本ガイドでご説明してきたとおり、レコメンデーションのアーキテクチャは、すでに進化を始めています。特に注目すべき変化が3つあります。

レコメンデーションワークフローにおけるエージェンティックAI。顧客プロファイルを自律的に照会し、コンテキストを評価し、レコメンデーション戦略を選択し、チャネル横断でアクティベーションを実行できるAIエージェントが、研究段階のプロトタイプから初期の本番運用段階へ移行しつつあります。これらのエージェントには、1秒未満のレイテンシで統合された顧客データにリアルタイムにアクセスできることが必要です。データインフラ要件は、従来のレコメンデーション配信と同等か、それ以上の厳しさが求められます。TealiumのMCP統合は、まさにこのパターンを想定して設計されており、AIエージェントに対して、同意取得済みの顧客コンテキストへのリアルタイムアクセスを提供します。

レコメンデーションコンテンツのための生成AI。生成モデルは、あらかじめ用意されたレコメンデーションテンプレートから選択するのではなく、顧客一人ひとりのコンテキストに合わせてパーソナライズされた説明文、商品説明、オファー文章を生成できます。これには、前述のリアルタイムの特徴量パイプラインが必要になりますが、今度は従来のランキングモデルと並行して、あるいはそれに代わって生成モデルにもデータを供給します。

マルチモーダルなレコメンデーションシグナル。音声によるインタラクション、動画の視聴傾向、画像検索は、レコメンデーションモデルが取り込める新たな行動シグナルを生み出しています。パイプラインアーキテクチャは、これらのシグナルを追加のイベントタイプとして取り込みますが、エンリッチメントレイヤーおよび特徴量エンジニアリングレイヤーは、非構造化データを処理できなければなりません。一般的には、マルチモーダル入力をベクトル表現に変換する埋め込みモデルによって実現されます。

これら3つの変化すべてに共通しているのは、鮮度の高い、同意取得済みの統合された顧客データをリアルタイムで配信する必要があるということです。モデルやチャネルは変わり続けていくでしょう。しかし、データ基盤の重要性は低下するどころか、ますます高まっていくでしょう。

モデルの前に基盤を構築

AIレコメンデーションの議論は、どのアーキテクチャを採用するのか、どのアルゴリズム、フレームワークを使うのかなど、最終的にはモデルへと焦点が移ります。

もちろん、これらの選択は重要です。ですが、それ以上に重要なのは、モデルへ供給されるデータです。古く、断片的で、同意を得ていないデータ上で動作する洗練されたハイブリッドモデルであっても、鮮度の高い、統合され適切に統制されたデータ上で動作するシンプルな協調フィルタリングよりも必ず性能が劣ります。

レコメンデーションシステムを適切に実装できる企業は、モデルの構築から始めたりしません。まずはパイプラインの構築から始めます。リアルタイムのイベント収集、取り込み時のID解決、処理途中でのデータエンリッチメント、そして変化する顧客の意図に追従できるほど高速な特徴量ストアに投資します。モデルは最後に構築します。なぜなら、モデルは最も容易に変更できる部分だからです。データ基盤こそが、時間とともに価値を積み上げていく部分です。

レコメンデーションモデルは今後も進化し続けるでしょう。本ガイドでご説明したアーキテクチャは、いかなる個々のモデル世代よりも長く使われ続けるでしょう。難しい課題は昔から変わりません。それは、大切な瞬間が過ぎてしまう前に、適切なデータを適切な場所に届けることができるか、ということです。この課題を解決した企業は、タイヤ交換をするようにモデルを変更できるようになります。解決できなかったチームは、これからもアルゴリズムのせいにし続けるでしょう。

参考文献

1. Redis. “Real-Time AI Recommendation Systems.” redis.io/blog/real-time-ai-recommendation-systems.
2. Tinybird. “Real-Time Recommendation System.” tinybird.co/blog/real-time-recommendation-system.
3. HumAI. “Real-Time AI Inference 2026: Complete Guide to Sub-100ms Models.” humai.blog/real-time-ai-inference-2026.
4. BrainForge. “How Netflix Uses Machine Learning to Create Perfect Recommendations.” brainforge.ai/blog/how-netflix-uses-machine-learning.
5. Catchpoint. “Semantic Caching: What We Measured, Why It Matters.” catchpoint.com/blog/semantic-caching-what-we-measured-why-it-matters.
6. AWS. “Lower Cost and Latency for AI Using Amazon ElastiCache as a Semantic Cache with Amazon Bedrock.” aws.amazon.com/blogs/database/lower-cost-and-latency-for-ai-using-amazon-elasticache.
7. Envive. “63 AI Personalization in eCommerce Lift Statistics.” envive. ai/post/ai-personalization-in-ecommerce-lift-statistics.

Tealiumについて

Tealium社は、業界をリードするカスタマーデータオーケストレーションプラットフォームを通して、エンタープライズ規模でAIに活用できる信頼性の高いデータを提供します。基盤となるデータレイヤーとして、インテリジェントなデータストリーミング、コンテキストエンジン、エンタープライズタグマネジメント、堅牢なAPIハブを備え、コンポーザブルアーキテクチャとリアルタイムアクティベーション向けに設計されたハイブリッド型カスタマーデータプラットフォーム（CDP）を提供しています。Tealiumの統合エコシステムは、1,300以上のすぐに使える連携機能や拡大を続けるAIパートナーエコシステムを含め、主要なデータクラウドやテクノロジープロバイダーと連携します。リアルタイムでコンテキストを捉えた、エンリッチ化された同意取得済みデータを提供することで、企業がAIのパフォーマンスを加速させ、業務効率を向上し、重要な顧客体験を強化できるよう支援します。

詳しくは **tealium.com/ja** をご覧ください。